

1. Introducción

Este informe detalla la implementación de infraestructura en Azure mediante Terraform, abordando los siguientes problemas:

- Despliegue de infraestructura en Azure para Odoo.
- Optimización de código Terraform.
- Despliegue de un balanceador de carga con máquinas virtuales.

Cada sección incluye una descripción del problema, la solución implementada y evidencia del despliegue realizado.

2. Desarrollo

Problema 1: Despliegue de infraestructura en Azure (Odoo)

Descripción del problema

Se requiere desplegar la infraestructura necesaria en Azure para realizar pruebas de penetración en la última versión de Odoo. La infraestructura debe ser optimizada en términos de costo y esfuerzo.

Solución propuesta

Criterios de costo y complejidad

Para optimizar costos, se ha priorizado el uso de recursos eficientes, evitando configuraciones sobredimensionadas. En lugar de una arquitectura distribuida costosa, se implementa una única máquina virtual con almacenamiento persistente y una distribución de azure Bastión para proteger el puerto SSH, Además, la adopción de redes privadas y políticas de seguridad adecuadas ha permitido minimizar la exposición innecesaria y reducir los costos asociados a la transferencia de datos.

En cuanto a la complejidad, se ha buscado un balance entre seguridad y facilidad de administración. La herramienta para automatización se realiza con Terraform para agilizar el despliegue.

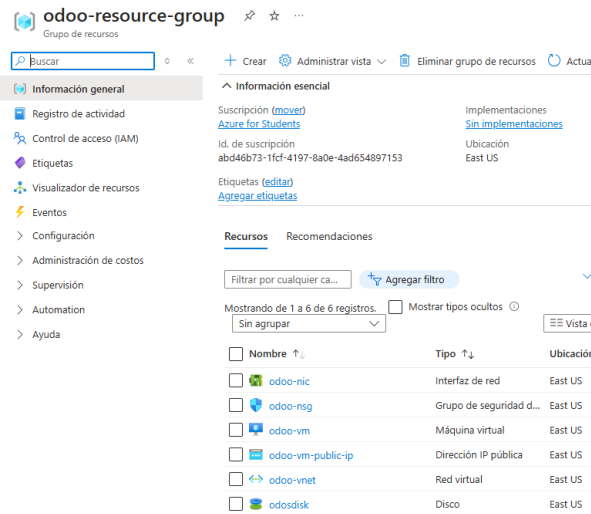
Para este despliegue se utilizaron los siguientes servicios en Azure:

- Máquina virtual (VM) basada en Ubuntu.
- Red virtual (VNet) con una subred privada.
- Grupo de seguridad de red (NSG) para controlar accesos.
- Almacenamiento persistente para la base de datos.

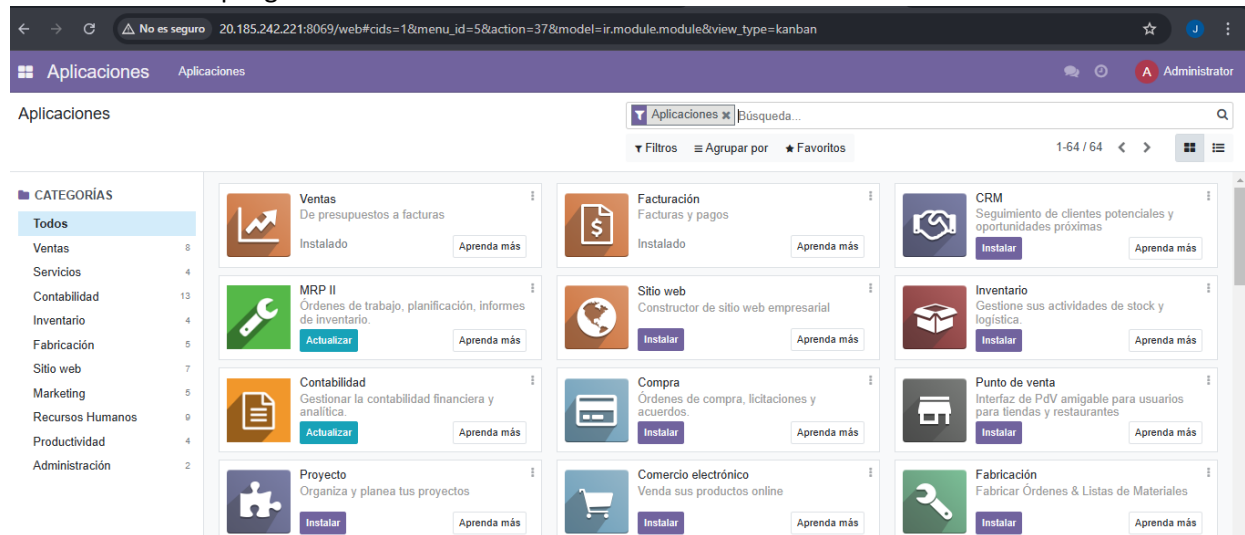
Evidencia del despliegue

Se implementó correctamente la infraestructura en la región East US mediante Terraform. Como se aprecia en la imagen

Infra estructura desplegada En Azure



Resultado del despliegue.



LINK: de production

http://20.185.242.221:8069/web#cids=1&menu_id=5&action=37&model=ir.module.module&view_type=kanban

U:pruebas@gmail.com

C:odoo

Problema 2: Optimización de código Terraform

Código original

El código original contenía redundancia en la declaración de grupos de recursos:

```
resource "azurerm_resource_group" "resource_g" {
  name     = "projectbittest"
  location = "West Europe"
}

resource "azurerm_resource_group" "resource_g" {
  name     = "projectbittest11"
  location = "West Europe"
}
```

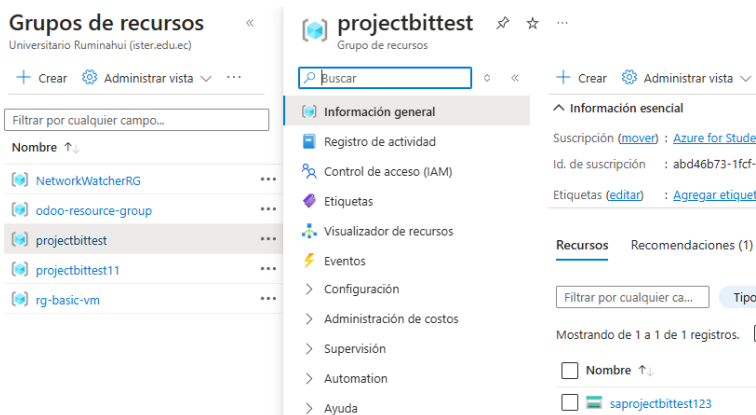
```
}
```

Optimización realizada

Se eliminó la duplicación y se estructuró mejor el código para mejorar su mantenimiento:

```
resource "azurerm_resource_group" "resource_g" {  
  name = "projectbittest"  
  location = "West Europe"  
}
```

Se aplicaron mejores prácticas en la nomenclatura de variables y reutilización de recursos.



Problema 3: Despliegue de un Balanceador de Carga

Descripción del problema

Se solicita desplegar la siguiente infraestructura en Azure mediante Terraform:

Grupo de recursos: rg-basic-vm.

Red virtual y subred: vnet-basic y subnet-basic.

Máquina Virtual (VM) con tamaño Standard_B1s, conectada a la subred y con IP pública.

Grupo de Seguridad de Red (NSG) permitiendo tráfico SSH (22) y HTTP/HTTPS.

Balanceador de carga (LB) configurado con un backend pool y health probe.

Solución propuesta

- Se implementaron los siguientes recursos en Terraform:
- Máquina Virtual: Ubuntu 22.04 con autenticación por usuario/contraseña.
- Balanceador de carga: Configurado con una IP pública estática.
- Reglas de seguridad: Acceso SSH y tráfico web permitido.

Estructura del código

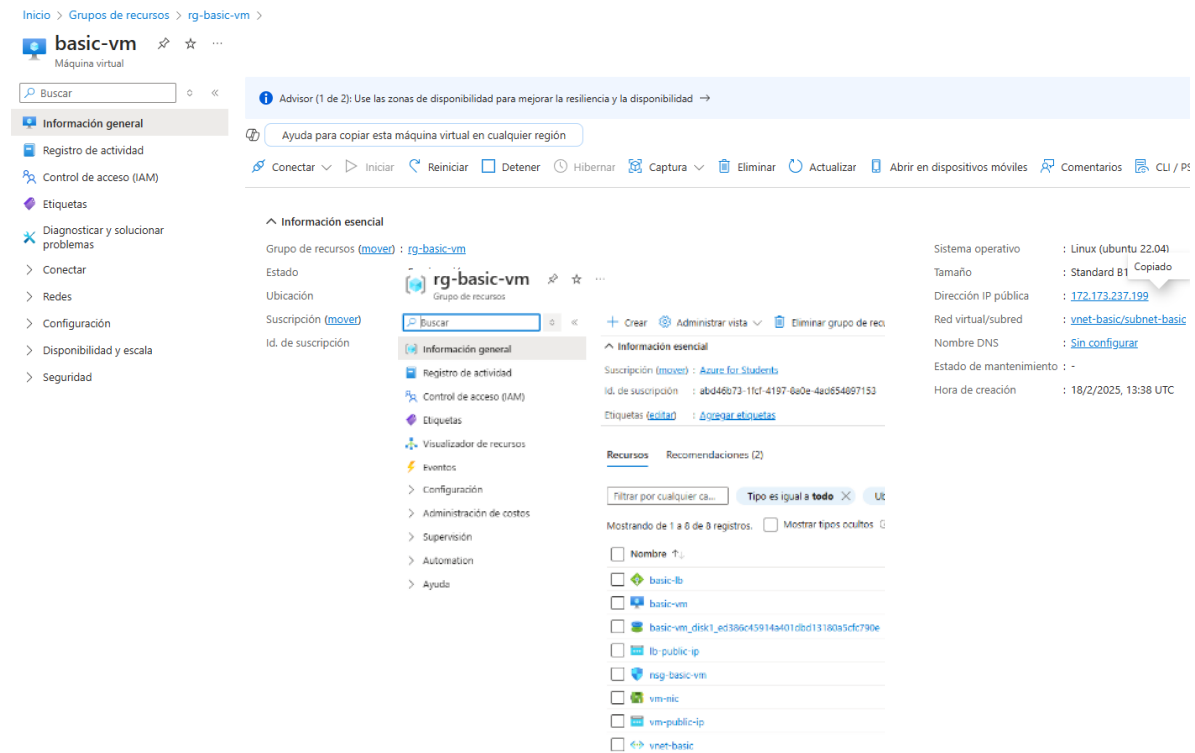
El código ha sido organizado en archivos:

main.tf → Define los recursos principales.

load_balancer.tf → Configuración del balanceador de carga.
deploy.sh → Script para instalar Docker y desplegar un contenedor Nginx.
El código completo se adjunta en un archivo comprimido.

Evidencia del despliegue

Se realizó el despliegue exitosamente en East US, como se muestra en las capturas de pantalla adjuntas.



20.185.70.186 (lb-public-ip)
ssh administrador@20.185.242.221
Contraseña: Pruebas10203040

3. Conclusión

El despliegue se realizó exitosamente utilizando Terraform en Azure, asegurando buenas prácticas en optimización de código e infraestructura. Se adjunta el código completo y evidencia del despliegue comprimido para una mejor apreciación.