

# Caso\_practico\_2

---

## Preliminares

---

### Estructura de la practica

La práctica se encuentra repositada en el siguiente repositorio de github: [Actividad2](#)

La estructura de la misma se muestra a continuación:

- En la carpeta IaC encontraremos el código *Terraform* y *Ansible* de la solución.
- En la carpeta dos encontraremos la documentación de la solución.
- En la carpeta k8s hay pruebas y tests adicionales que se han hecho aprovechando la coyuntura.

```
├── docs
│   ├── Caso_Practico_2.md
│   ├── md.sh
│   └── _resources
├── IaC
│   ├── ansible
│   │   ├── aks.yml
│   │   ├── a.txt
│   │   ├── inventories
│   │   │   ├── azure
│   │   │   │   ├── group_vars
│   │   │   │   │   └── azure
│   │   │   │   └── main.yml
│   │   │   └── hosts.yml
│   │   └── local
│   │       ├── group_vars
│   │       │   └── local
│   │       ├── k8s.yml
│   │       ├── main.yml
│   │       └── hosts.yml
│   ├── registry.yml
│   ├── roles
│   │   ├── acr
│   │   │   ├── files
│   │   │   │   ├── a.html
│   │   │   │   └── Dockerfile
│   │   │   └── tasks
│   │   │       └── main.yml
```

```
| | | | | └─ aks
| | | | | | └─ files
| | | | | | | └─ argocd
| | | | | | | | └─ loadbalancer.yml
| | | | | | | | └─ namespace.yml
| | | | | | | └─ config.yml
| | | | | | | └─ tekton
| | | | | | | | └─ namespace.yml
| | | | | | └─ wp
| | | | | | | └─ mysql-deployment.yml
| | | | | | | └─ mysql-pvc.yml
| | | | | | | └─ mysql-service.yml
| | | | | | | └─ namespace.yml
| | | | | | | └─ secret.yml
| | | | | └─ tasks
| | | | | | └─ argocd.yml
| | | | | | └─ main.yml
| | | | | | └─ tekton.yml
| | | | | | └─ wp.yml
| | | | | └─ templates
| | | | | | └─ mysql-deployment.yaml.j2
| | | | | | └─ registry-credentials.yaml.j2
| | | | | | └─ wordpress-deployment.yaml.j2
| | | | └─ podman
| | | | | └─ tasks
| | | | | | └─ main.yml
| | └─ tfnode.yml
| └─ terraform
| | └─ acr.tf
| | └─ aks.tf
| | └─ dns.tf
| | └─ graph.svg
| | └─ installer.tf
| | └─ locals.tf
| | └─ provider.tf
| | └─ resources
| | | └─ ips.yaml
| | | └─ scripts
| | | | └─ cloud-init.yml
| | | | └─ webserver.sh
| | └─ templates
| | | └─ deployer_vars.tpl
| | | └─ remote_inventory.tpl
```

```

|         |         └─ remote_vars.tpl
|         |─ rg.tf
|         |─ setup.sh
|         |─ sg.tf
|         |─ terraform.tf
|         |─ terraform.tfstate
|         |─ terraform.tfstate.backup
|         |─ terraform.tfvars
|         |─ variables.tf
|         |─ vm.tf
|         └─ vpc.tf
└─ k8s
    |─ argocd
    |   |─ deployment
    |   |   └─ prod
    |   |       └─ deployment.yml
    |   |       └─ kustomization.yml
    |   |       └─ namespace.yml
    |   |       └─ registry-credentials.yml
    |   |       └─ service.yml
    |   |─ Dockerfile
    |   └─ main.py
    └─ helm
        └─ wordpress
            └─ charts
                └─ Chart.yaml
                └─ templates
                    |─ deployment.yaml
                    |─ _helpers.tpl
                    |─ hpa.yaml
                    |─ ingress.yaml
                    |─ NOTES.txt
                    |─ persistenVolumeClaim.yaml
                    |─ serviceaccount.yaml
                    |─ service.yaml
                    └─ tests
                        └─ test-connection.yaml
                └─ values.yaml
└─ README.md

```

## Creación de la infraestructura

---

### Acceso a la cuenta de Azure

## Instalación de Azure Cli

Instalamos en local la herramienta de Azure Cli para poder interactuar con el proveedor de forma programática

```
[dani@draco ~/Documents/asignaturas/unir/devops/actividades/act2/actividad2 ] $ paru azure-cli
1 extra/azure-cli 2.61.0-1 [41.84 MiB 1.08 GiB]
    Command-line tools for Microsoft Azure
2 aur/azure-cli-bin 2.61.0-2 [+4 ~1.43]
    Azure Command-Line Interface (Debian package)
:: Packages to install (eg: 1 2 3, 1-3):
:: 2
:: Resolving dependencies...
:: Calculating conflicts...
:: Calculating inner conflicts...

Aur (1) azure-cli-bin-2.61.0-2

:: Proceed to review? [Y/n]: y
```

Una vez instalada la herramienta probamos el acceso via az login

```
[dani@draco ~/Documents/asignaturas/unir/devops/actividades/act2/actividad2 ] $ az login
A web browser has been opened at https://login.microsoftonline.com/organizations/oauth2/v2.0/authorize?client_id=00000000-0000-0000-0000-000000000000&redirect_uri=https://login.microsoftonline.com/organizations/oauth2/v2.0/authorize?client_id=00000000-0000-0000-0000-000000000000&response_type=code&scope=openid%20profile%20email%20https://microsoft.com/devicelogin
.

Retrieving tenants and subscriptions for the selection...

[Tenant and subscription selection]

No      Subscription name      Subscription ID      Tenant
-----
[1] *   Azure for Students   c190437f-864c-4fab-a46d-94b7dfc565d0   UNIR

The default is marked with an *; the default tenant is 'UNIR' and subscription is 'Azure for S

Select a subscription and tenant (Type a number or Enter for no changes): 1

Tenant: UNIR
Subscription: Azure for Students (c190437f-864c-4fab-a46d-94b7dfc565d0)

[Announcements]
With the new Azure CLI login experience, you can select the subscription you want to use more

If you encounter any problem, please open an issue at https://aka.ms/azclibug

[Warning] The login output has been updated. Please be aware that it no longer displays the fu

[dani@draco ~/Documents/asignaturas/unir/devops/actividades/act2/actividad2 ] $
```

## Y verificamos los datos

```
[dani@draco ~] $ az account list
[
  {
    "cloudName": "AzureCloud",
    "homeTenantId": "899789dc-202f-44b4-8472-a6d40f9eb440",
    "id": "c190437f-864c-4fab-a46d-94b7dfc565d0",
    "isDefault": true,
    "managedByTenants": [],
    "name": "Azure for Students",
    "state": "Enabled",
    "tenantDefaultDomain": "comunidadunir.net",
    "tenantDisplayName": "UNIR",
    "tenantId": "899789dc-202f-44b4-8472-a6d40f9eb440",
    "user": {
      "name": "daniel.argamenteria132@comunidadunir.net",
      "type": "user"
    }
  }
]
```

## Aprovisionamiento de Infra

Para la provisión de la infraestructura vamos a utilizar OpenTofu. Antes de realizar cualquier acción se va a crear una *storage account* para almacenar ahí el fichero de estado *tfstate*.

## Creación de una *storage account* para el almacenamiento del *tfstate*

Con el siguiente *script* creamos una *storage account* para el almacenamiento del *tfstate*. Definimos una serie de variables para personalizar la creación de la cuenta y lo creamos en *northeurope* esta localización será la que usaremos a lo largo del proyecto por defecto.

```
#!/bin/bash
```

```
RESOURCE_GROUP_NAME=tfstates  
STORAGE_ACCOUNT_NAME=tfstateunir  
CONTAINER_NAME=unir
```

```
az group create --name $RESOURCE_GROUP_NAME --location northeurope  
az storage account create --resource-group $RESOURCE_GROUP_NAME --name  
$STORAGE_ACCOUNT_NAME --sku Standard_LRS --encryption-services blob  
az storage container create --name $CONTAINER_NAME --account-name  
$STORAGE_ACCOUNT_NAME
```

```
+ [dani@draco  
~/Documents/asignaturas/unir/devops/actividades/act2/actividad2/IaC/terrafor  
m ] (TF:default) $ bash -x setup.sh  
+ RESOURCE_GROUP_NAME=tfstates  
+ STORAGE_ACCOUNT_NAME=tfstateunir  
+ CONTAINER_NAME=unir  
+ az group create --name tfstates --location westeurope  
{  
  "id": "/subscriptions/c190437f-864c-4fab-a46d-  
94b7dfc565d0/resourceGroups/tfstates",  
  "location": "westeurope",  
  "managedBy": null,  
  "name": "tfstates",  
  "properties": {  
    "provisioningState": "Succeeded"  
  },  
  "tags": null,  
  "type": "Microsoft.Resources/resourceGroups"  
}  
+ az storage account create --resource-group tfstates --name tfstateunir --  
sku Standard_LRS --encryption-services blob  
{
```

```
"accessTier": "Hot",
"accountMigrationInProgress": null,
"allowBlobPublicAccess": false,
"allowCrossTenantReplication": false,
"allowSharedKeyAccess": null,
"allowedCopyScope": null,
"azureFilesIdentityBasedAuthentication": null,
"blobRestoreStatus": null,
"creationTime": "2024-07-08T16:07:47.349594+00:00",
"customDomain": null,
"defaultToOAuthAuthentication": null,
"dnsEndpointType": null,
"enableHttpsTrafficOnly": true,
"enableNfsV3": null,
"encryption": {
  "encryptionIdentity": null,
  "keySource": "Microsoft.Storage",
  "keyVaultProperties": null,
  "requireInfrastructureEncryption": null,
  "services": {
    "blob": {
      "enabled": true,
      "keyType": "Account",
      "lastEnabledTime": "2024-07-08T16:07:47.630849+00:00"
    },
    "file": {
      "enabled": true,
      "keyType": "Account",
      "lastEnabledTime": "2024-07-08T16:07:47.630849+00:00"
    },
    "queue": null,
    "table": null
  }
},
"extendedLocation": null,
"failoverInProgress": null,
"geoReplicationStats": null,
"id": "/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/tfstates/providers/Microsoft.Storage/storageAccounts/tfstateunir",
"identity": null,
"immutableStorageWithVersioning": null,
"isHnsEnabled": null,
```

```
"isLocalUserEnabled": null,
"isSftpEnabled": null,
"isSkuConversionBlocked": null,
"keyCreationTime": {
  "key1": "2024-07-08T16:07:47.490221+00:00",
  "key2": "2024-07-08T16:07:47.490221+00:00"
},
"keyPolicy": null,
"kind": "StorageV2",
"largeFileSharesState": null,
"lastGeoFailoverTime": null,
"location": "westeurope",
"minimumTlsVersion": "TLS1_0",
"name": "tfstateunir",
"networkRuleSet": {
  "bypass": "AzureServices",
  "defaultAction": "Allow",
  "ipRules": [],
  "ipv6Rules": [],
  "resourceAccessRules": null,
  "virtualNetworkRules": []
},
"primaryEndpoints": {
  "blob": "https://tfstateunir.blob.core.windows.net/",
  "dfs": "https://tfstateunir.dfs.core.windows.net/",
  "file": "https://tfstateunir.file.core.windows.net/",
  "internetEndpoints": null,
  "microsoftEndpoints": null,
  "queue": "https://tfstateunir.queue.core.windows.net/",
  "table": "https://tfstateunir.table.core.windows.net/",
  "web": "https://tfstateunir.z6.web.core.windows.net/"
},
"primaryLocation": "westeurope",
"privateEndpointConnections": [],
"provisioningState": "Succeeded",
"publicNetworkAccess": null,
"resourceGroup": "tfstates",
"routingPreference": null,
"sasPolicy": null,
"secondaryEndpoints": null,
"secondaryLocation": null,
"sku": {
  "name": "Standard_LRS",
```

```

    "tier": "Standard"
  },
  "statusOfPrimary": "available",
  "statusOfSecondary": null,
  "storageAccountSkuConversionStatus": null,
  "tags": {},
  "type": "Microsoft.Storage/storageAccounts"
}
+ az storage container create --name unir --account-name tfstateunir

```

There are no credentials provided **in** your command **and** environment, we will query **for** account key **for** your storage account.

It **is** recommended to provide `--connection-string`, `--account-key` **or** `--sas-token` **in** your command **as** credentials.

You also can add `--auth-mode login` **in** your command to use Azure Active Directory (Azure AD) **for** authorization **if** your login account **is** assigned required RBAC roles.

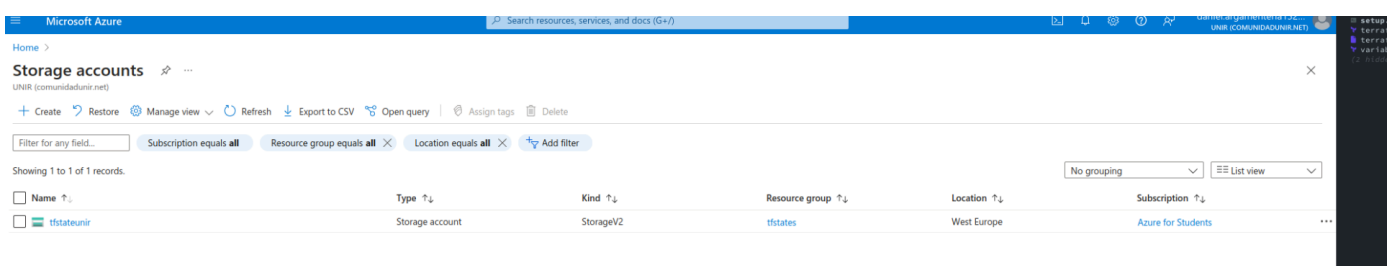
For more information about RBAC roles **in** storage, visit <https://docs.microsoft.com/azure/storage/common/storage-auth-aad-rbac-cli>.

In addition, setting the corresponding environment variables can avoid inputting credentials **in** your command. Please use `--help` to get more information about environment variable usage.

```

{
  "created": true
}

```



## Terraformación

### Preliminares

El proceso de terraformación en Azure no es tan facil como en AWS. Hay que tener paciencia y tener en cuenta la naturaleza asíncrona de la creación de elementos, ya que estos no se crean/destruyen tan rápido como en AWS.

También se ha evitado el uso de módulos de terceras partes. Al contrario que con AWS que utilizamos los modulos de la comunidad.



## Código terraform

Para mayor legibilidad se ha separado el código en un fichero por recurso. Del mismo modo se han separado la definición de las variables y los valores de las variables en ficheros separados. Esto nos permite utilizar el mismo código para distintos entornos.

También se ha puesto énfasis en separar la configuración del código de tal forma que **no se tenga que tocar el código para cambiar la configuración de un recurso**.

```
.
├─ acr.tf
├─ aks.tf
├─ installer.tf
├─ locals.tf
├─ provider.tf
├─ resources
│   ├─ ips.yaml
│   ├─ scripts
│   │   ├─ cloud-init.yml
│   │   └─ webserver.sh
│   └─ templates
│       ├─ deployer_vars.tpl
│       ├─ remote_inventory.tpl
│       └─ remote_vars.tpl
├─ rg.tf
├─ setup.sh
├─ sg.tf
├─ terraform.tf
├─ terraform.tfstate
├─ terraform.tfstate.backup
├─ terraform.tfvars
├─ variables.tf
├─ vm.tf
└─ vpc.tf

4 directories, 22 files
```

Casi todos los recursos utilizan el siguiente patron:

```
for_each = { for k, v in var.acr : k => v if v.enabled == true } #only
enabled ones
```

Esto se realiza a propósito para poder habilitar o deshabilitar recursos desde configuración, así como permitir una configuración más dinámica ya que los bloques de tipo `count` se indexan y son

dependientes del orden de creación.

No se han hecho arcos de iglesia para gestionar la infraestructura, a excepción de las reglas de seguridad que se ha utilizado un código algo exótico para evitar colisiones en los nombres de las reglas y en la asignación de los prefijos de red de origen y destino.

```
resource "azurerm_network_security_rule" "nsgr" {  
  # nsrg is a list  
  # convert it to k => V  
  # The key is combination of rg, nsrg, and sha1 of several values  
  for_each = { for rule in var.nsgr : join("-", [rule.rg, rule.nsrg,  
sha1("${rule.rg}${rule.sr}${rule.dr}${rule.sp}${rule.dp}")] )  
=> rule }  
  
  name = each.key  
  
  resource_group_name      =  
azurerm_resource_group.rgs[each.value.rg].name  
  network_security_group_name = each.value.nsrg  
  
  priority          = each.value.priority  
  direction         = each.value.direction  
  access            = each.value.access  
  protocol           = each.value.protocol  
  source_port_range = each.value.sr  
  destination_port_range = each.value.dr  
  #Need a list fof sap an dap  
  # if != *  
  #   if have dots  
  #     mean we have a string with ips so convert the string onto list of  
ips  
  #   else  
  #     go to the local ip_map and get its contents  
  # else  
  #   return *  
  source_address_prefixes = [each.value.sr] != "*" ?  
strcontains(each.value.sp, ".") ? tolist(split(", ", each.value.sp)) :  
["${local.ip_map[each.value.sp]}/32"] : ["*"]  
  destination_address_prefixes = [each.value.dr] != "*" ?  
strcontains(each.value.dp, ".") ? tolist(split(", ", each.value.dp)) :  
["${local.ip_map[each.value.dp]}/32"] : ["*"]  
}
```

**Recusos creados**

## A continuación se muestra nuestro plan

```
azurerm_resource_group.rgs["unir-arga2"]: Refreshing state...
[id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2]
azurerm_subnet.snets["pub_snet_a"]: Refreshing state...
[id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/virtualNetworks/unir_vpc/subnets/pub_snet_a]
azurerm_nat_gateway.ngw["unir_ngw"]: Refreshing state...
[id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/natGateways/unir_ngw]
azurerm_public_ip.ngw-ip["unir_ngw"]: Refreshing state...
[id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/publicIPAddresses/unir_ngw-ip]
azurerm_network_security_rule.nsgr["unir-arga2-public-3d1e0387f8539b6b9fc67803909b5e17a64c247b"]: Refreshing state...
[id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/networkSecurityGroups/public/securityRules/unir-arga2-public-3d1e0387f8539b6b9fc67803909b5e17a64c247b]
azurerm_network_security_rule.nsgr["unir-arga2-public-a7d53631b0ac6515341eb4d9fc2318aafe8af511"]: Refreshing state...
[id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/networkSecurityGroups/public/securityRules/unir-arga2-public-a7d53631b0ac6515341eb4d9fc2318aafe8af511]
azurerm_virtual_network.vpc["unir_vpc"]: Refreshing state...
[id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/virtualNetworks/unir_vpc]
azurerm_network_security_rule.nsgr["unir-arga2-public-a72b86ad9b512aeb5540e38f789edc72eb705168"]: Refreshing state...
[id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/networkSecurityGroups/public/securityRules/unir-arga2-public-a72b86ad9b512aeb5540e38f789edc72eb705168]
azurerm_container_registry.acr["acr"]: Refreshing state...
[id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.ContainerRegistry/registries/acrunir]
azurerm_network_security_group.nsg["public"]: Refreshing state...
[id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/networkSecurityGroups/public]
azurerm_kubernetes_cluster.aks["aks"]: Refreshing state...
[id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.ContainerService/managedClusters/aksunir]
azurerm_public_ip.pub_ip["webserver"]: Refreshing state...
```

```
[id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-
arga2/providers/Microsoft.Network/publicIPAddresses/webserver-public-ip]
azurerm_subnet_network_security_group_association.sb_nsg["snet_pub_snet_a_ns
g_public"]: Refreshing state... [id=/subscriptions/c190437f-864c-4fab-a46d-
94b7dfc565d0/resourceGroups/unir-
arga2/providers/Microsoft.Network/virtualNetworks/unir_vpc/subnets/pub_snet_
a]
data.template_file.remote_inventory: Reading...
data.template_file.remote_inventory: Read complete after 0s
[id=3193e090bd09119fc743dd3425c959bf746f7a4c754dedce8afb38a97bb43748]
azurerm_network_interface.wms_ani["webserver"]: Refreshing state...
[id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-
arga2/providers/Microsoft.Network/networkInterfaces/webserver-ani]
data.template_file.deployer_vars: Reading...
data.template_file.remote_vars: Reading...
data.template_file.deployer_vars: Read complete after 0s
[id=496bca6b34c223d249e4a496c93b35ad713364e5a7d956007228eb8f989f8048]
data.template_file.remote_vars: Read complete after 0s
[id=7c1028c0d326691c5cb415a763c9fd05168f58a99d24531437c8c5d463815e16]
local_file.deployer_vars: Refreshing state...
[id=de9686b0a87ca980661ac502ff2e6e9a36elf974]
azurerm_linux_virtual_machine.vms["webserver"]: Refreshing state...
[id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-
arga2/providers/Microsoft.Compute/virtualMachines/webserver]
local_file.remote_inventory: Refreshing state...
[id=e4bbbeb1c88df65ca55f1e72dc3481dd1e385fa28]
local_file.remote_vars: Refreshing state...
[id=7f184eeef05f491289681e4893638831c9e02a6a]

No changes. Your infrastructure matches the configuration.

OpenTofu has compared your real infrastructure against your configuration and
found no differences, so no changes are needed.
```

```
[dan@draco ~]$ cat /dev/null > /dev/null && terraform [main] >>> [!default] $ tofu plan
azurerm_resource_group.rg["unir-arga2"]: Refreshing state... [id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2]
azurerm_subnet.snet["pub_snet_a"] Refreshing state... [id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/virtualNetworks/unir_vpc/subnets/pub_snet_a]
azurerm_nat_gateway.ngw["unir-ngw"] Refreshing state... [id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/natGateways/unir-ngw]
azurerm_public_ip.ngw["unir-ngw"] Refreshing state... [id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/publicIPAddresses/unir-ngw-ip]
azurerm_virtual_network.vpc["unir-vpc"] Refreshing state... [id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/virtualNetworks/unir_vpc]
azurerm_container_registry.acr["acr"] Refreshing state... [id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.ContainerRegistry/registries/acr-unir]
azurerm_network_security_group.nsg["public"] Refreshing state... [id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/networkSecurityGroups/public]
azurerm_network_security_rule.nsg["allow_http_public"] Refreshing state... [id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/networkSecurityGroups/public/securityRules/allow_http_public]
azurerm_network_security_rule.nsg["allow_ssh_public"] Refreshing state... [id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/networkSecurityGroups/public/securityRules/allow_ssh_public]
azurerm_public_ip.pub["webserver"] Refreshing state... [id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/publicIPAddresses/webserver-public-ip]
azurerm_kubernetes_cluster.example["aks"] Refreshing state... [id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.ContainerService/managedClusters/aks-unir]
data.template_file.remote_inventory: Reading...
data.template_file.remote_inventory: Read complete after 0s [id=3193e090bd09119fc743dd3425c959bf746f7a4c754dedce8afb38a97bb43748]
azurerm_network_interface.wms_ani["webserver"] Refreshing state... [id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/networkInterfaces/webserver-ani]
azurerm_subnet_network_security_group_association.sb_nsg["snet_pub_snet_a_nsg_public"] Refreshing state... [id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/virtualNetworks/unir_vpc/subnets/pub_snet_a]
azurerm_linux_virtual_machine.vms["webserver"] Refreshing state... [id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Compute/virtualMachines/webserver]
data.template_file.deployer_vars: Reading...
data.template_file.deployer_vars: Read complete after 0s [id=496bca6b34c223d249e4a496c93b35ad713364e5a7d956007228eb8f989f8048]
data.template_file.remote_vars: Read complete after 0s [id=7c1028c0d326691c5cb415a763c9fd05168f58a99d24531437c8c5d463815e16]
local_file.deployer_vars: Refreshing state... [id=de9686b0a87ca980661ac502ff2e6e9a36elf974]
local_file.remote_vars: Refreshing state... [id=7f184eeef05f491289681e4893638831c9e02a6a]
local_file.remote_inventory: Refreshing state... [id=e4bbbeb1c88df65ca55f1e72dc3481dd1e385fa28]
```

El siguiente bloque de código muestra los recursos terraformados

```
# data.template_file.deployer_vars:
data "template_file" "deployer_vars" {
```

```

    id          =
"496bca6b34c223d249e4a496c93b35ad713364e5a7d956007228eb8f989f8048"
    rendered = <<-EOT
        registry:
            admin_password: "YYYY"
            admin_username: "acrunir"
            source_repo: "registry.hub.docker.com/nginx:latest"
            source_img: "nginx:latest"
            dest_repo: "acrunir.azurecr.io"
            dest_img: "nginx:casopractico2"
    EOT
    template = <<-EOT
        registry:
            admin_password: "${admin_password}"
            admin_username: "${admin_username}"
            source_repo: "registry.hub.docker.com/nginx:latest"
            source_img: "nginx:latest"
            dest_repo: "${dest_repo}"
            dest_img: "nginx:casopractico2"
    EOT
    vars      = {
        "admin_password" = (sensitive value)
        "admin_username" = "acrunir"
        "dest_repo"      = "acrunir.azurecr.io"
    }
}

# data.template_file.remote_inventory:
data "template_file" "remote_inventory" {
    id          =
"3193e090bd09119fc743dd3425c959bf746f7a4c754dedce8afb38a97bb43748"
    rendered = <<-EOT
        azure:
            hosts:
                tfnode:
                    ansible_host: 40.69.204.208
                    ansible_port: 22
    EOT
    template = <<-EOT
        azure:
            hosts:
                tfnode:
                    ansible_host: ${tfhost}

```

```

        ansible_port: 22

EOT
vars      = {
    "tfhost" = "40.69.204.208"
}
}

# data.template_file.remote_vars:
data "template_file" "remote_vars" {
    id      =
    "7c1028c0d326691c5cb415a763c9fd05168f58a99d24531437c8c5d463815e16"
    rendered = <<-EOT
        packages:
            - podman
            - tree

        container:
            localPath: /data
            containerPath: /usr/share/nginx/html

        registry:
            admin_password: "YYYY"
            admin_username: "acrunir"
            source_repo: "registry.hub.docker.com/nginx:latest"
            source_img: "nginx:latest"
            dest_repo: "acrunir.azurecr.io"
            dest_img: "nginx:casopractico2"
    EOT
    template = <<-EOT
        packages:
            - podman
            - tree

        container:
            localPath: /data
            containerPath: /usr/share/nginx/html

        registry:
            admin_password: "${admin_password}"
            admin_username: "${admin_username}"
            source_repo: "registry.hub.docker.com/nginx:latest"
            source_img: "nginx:latest"
            dest_repo: "${dest_repo}"

```

```

        dest_img: "nginx:casopractico2"

EOT
vars      = {
    "admin_password" = (sensitive value)
    "admin_username" = "acrunir"
    "dest_repo"      = "acrunir.azurecr.io"
}
}

# azurerm_container_registry.acr["acr"]:
resource "azurerm_container_registry" "acr" {
  admin_enabled          = true
  admin_password         = (sensitive value)
  admin_username         = "acrunir"
  anonymous_pull_enabled = false
  data_endpoint_enabled = false
  encryption             = [
    {
      enabled          = false
      identity_client_id = ""
      key_vault_key_id  = ""
    },
  ]
  export_policy_enabled = true
  id                    = "/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.ContainerRegistry/registries/acrunir"
  location              = "northeurope"
  login_server          = "acrunir.azurecr.io"
  name                  = "acrunir"
  network_rule_bypass_option = "AzureServices"
  network_rule_set      = []
  public_network_access_enabled = true
  quarantine_policy_enabled = false
  resource_group_name      = "unir-arga2"
  retention_policy         = [
    {
      days      = 7
      enabled   = false
    },
  ]
  sku = "Basic"
  tags = {

```

```

        "environment" = "unir"
        "terraform"   = "true"
    }
    trust_policy          = [
        {
            enabled = false
        },
    ]
    zone_redundancy_enabled = false
}

# azurerm_kubernetes_cluster.aks["aks"]:
resource "azurerm_kubernetes_cluster" "aks" {
    api_server_authorized_ip_ranges = []
    cost_analysis_enabled           = false
    current_kubernetes_version      = "1.28.10"
    custom_ca_trust_certificates_base64 = []
    dns_prefix                      = "unir"
    enable_pod_security_policy       = false
    fqdn                           = "unir-
wliuru70.hcp.northeurope.azmk8s.io"
    id                             = "/subscriptions/c190437f-864c-
4fab-a46d-94b7dfc565d0/resourceGroups/unir-
arga2/providers/Microsoft.ContainerService/managedClusters/aksunir"
    image_cleaner_enabled          = false
    image_cleaner_interval_hours   = 48
    kube_admin_config              = (sensitive value)
    kube_config                    = (sensitive value)
    kube_config_raw                = (sensitive value)
    kubernetes_version             = "1.28"
    local_account_disabled         = false
    location                      = "northeurope"
    name                          = "aksunir"
    node_resource_group            = "MC_unir-
arga2_aksunir_northeurope"
    node_resource_group_id         = "/subscriptions/c190437f-864c-
4fab-a46d-94b7dfc565d0/resourceGroups/MC_unir-arga2_aksunir_northeurope"
    oidc_issuer_enabled           = false
    portal_fqdn                   = "unir-
wliuru70.portal.hcp.northeurope.azmk8s.io"
    private_cluster_enabled        = false
    private_cluster_public_fqdn_enabled = false
    public_network_access_enabled  = true

```



```

resource_group_name          = "unir-arga2"
role_based_access_control_enabled = true
run_command_enabled         = true
sku_tier                     = "Free"
support_plan                 = "KubernetesOfficial"
tags                         = {
    "environment" = "unir"
    "terraform"   = "true"
}
workload_identity_enabled    = false

default_node_pool {
    custom_ca_trust_enabled = false
    enable_auto_scaling     = false
    enable_host_encryption  = false
    enable_node_public_ip   = false
    fips_enabled            = false
    kubelet_disk_type       = "OS"
    max_count               = 0
    max_pods                = 110
    min_count               = 0
    name                    = "default"
    node_count              = 1
    node_labels             = {}
    node_taints            = []
    only_critical_addons_enabled = false
    orchestrator_version    = "1.28"
    os_disk_size_gb        = 128
    os_disk_type            = "Managed"
    os_sku                  = "Ubuntu"
    scale_down_mode        = "Delete"
    tags                   = {
        "environment" = "unir"
        "terraform"   = "true"
    }
    type = "VirtualMachineScaleSets"
    ultra_ssd_enabled = false
    vm_size            = "Standard_D2s_v3"
    zones              = []

    upgrade_settings {
        drain_timeout_in_minutes = 0
        max_surge                = "10%"
    }
}

```

```

        node_soak_duration_in_minutes = 0
    }
}

identity {
    identity_ids = []
    principal_id = "f92587da-50a7-4a18-acaf-1bc01728602b"
    tenant_id     = "899789dc-202f-44b4-8472-a6d40f9eb440"
    type          = "SystemAssigned"
}

kubelet_identity {
    client_id                = "11069553-f2fa-485d-bf28-b8ab6e380be8"
    object_id                = "f5a2ffb1-1168-468b-9580-2bb27d379678"
    user_assigned_identity_id = "/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/MC_unir-arga2_aksunir_northeurope/providers/Microsoft.ManagedIdentity/userAssignedIdentities/aksunir-agentpool"
}

linux_profile {
    admin_username = "ubuntu"

    ssh_key {
        key_data = "ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQDj/qvzCFBoF7piKZzY7faURI4IeZowQGWhIzIkruxqmYz2
CQOxjrr02dNM68ndJb/0nHv0aVApUzSsVPCjpbq9FcVhysjtmKnPedDLpsQL2gCKoJJm1GAVNt/xL
sV57dxma1/5Vf3oLjgKavQUG/PDho2z62/hg0U+MUoegcjG7STKVuidOWGE3mNsKIksWslwI6y20
ON04ueO1pKWBBSSzbCxBK/lRo+gf6jiEVqmwXvOSv453H4ta4PN7iRpInwDQU1Dxz+tCewPLID8d5E
wgao4a9oL04H0io8ESSSnnxyVaNbbG/pEOhN1MER81e2IS2MVXu7bodPIAPIjOMUrN8/
dani@draco"
    }
}

network_profile {
    dns_service_ip      = "10.0.0.10"
    ip_versions         = [
        "IPv4",
    ]
    load_balancer_sku   = "standard"
    network_data_plane  = "azure"
    network_plugin      = "kubenet"
    outbound_ip_address_ids = []
}

```

```

    outbound_ip_prefix_ids = []
    outbound_type           = "loadBalancer"
    pod_cidr                = "10.244.0.0/16"
    pod_cidrs               = [
        "10.244.0.0/16",
    ]
    service_cidr            = "10.0.0.0/16"
    service_cidrs           = [
        "10.0.0.0/16",
    ]

    load_balancer_profile {
        effective_outbound_ips = [
            "/subscriptions/c190437f-864c-4fab-a46d-
94b7dfc565d0/resourceGroups/MC_unir-
arga2_aksunir_northeurope/providers/Microsoft.Network/publicIPAddresses/9cf8
b64a-3153-4731-ab33-aebc6c9ae1b1",
        ]
        idle_timeout_in_minutes = 0
        managed_outbound_ip_count = 1
        managed_outbound_ipv6_count = 0
        outbound_ip_address_ids = []
        outbound_ip_prefix_ids = []
        outbound_ports_allocated = 0
    }
}

# azurerm_linux_virtual_machine.vms["webserver"]:
resource "azurerm_linux_virtual_machine" "vms" {
    admin_username = "ubuntu"
    allow_extension_operations = true
    bypass_platform_safety_checks_on_user_schedule_enabled = false
    computer_name = "webserver"
    custom_data = (sensitive
value)
    disable_password_authentication = true
    disk_controller_type = "SCSI"
    encryption_at_host_enabled = false
    extensions_time_budget = "PT1H30M"
    id =
"/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-
arga2/providers/Microsoft.Compute/virtualMachines/webserver"

```

```

location = "northeurope"
max_bid_price = -1
name = "webserver"
network_interface_ids = [
    "/subscriptions/c190437f-864c-4fab-a46d-
94b7dfc565d0/resourceGroups/unir-
arga2/providers/Microsoft.Network/networkInterfaces/webserver-ani",
]
patch_assessment_mode = "ImageDefault"
patch_mode = "ImageDefault"
platform_fault_domain = -1
priority = "Regular"
private_ip_address = "10.0.0.4"
private_ip_addresses = [
    "10.0.0.4",
]
provision_vm_agent = true
public_ip_address = "40.69.204.208"
public_ip_addresses = [
    "40.69.204.208",
]
resource_group_name = "unir-arga2"
secure_boot_enabled = false
size =
"Standard_D2s_v3"
tags = {
    "environment" = "unir"
    "terraform" = "true"
}
virtual_machine_id = "78dc13d0-8820-
4ae6-a02f-c4e40f9c7caf"
vm_agent_platform_updates_enabled = false
vtpm_enabled = false

admin_ssh_key {
    public_key = "ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQDj/qvzCFBoF7piKZzY7faURI4IeZowQGWhIzIkruXqmYz2
CQOxjrr02dNM68ndJb/0nHv0aVApUzSsVPCjPq9FcVhysjtmKnPedDLpsQL2gCKoJJm1GAVNt/xL
sV57dxma1/5Vf3oLjgKavQUG/PDho2z62/hg0U+MUoegcjG7STKVuidOWGE3mNsKIksWslwI6y20
ON04ue01pKWBBsZbCxK/lRo+gf6jiEVqmwXvOSv453H4ta4PN7iRpInwDQU1Dxz+tCewPLID8d5E
wgao4a9oL04H0io8ESSSnnxyVaNbbG/pEOhN1MER81e2IS2MVXu7bodPIAPIjOMUrN8/
dani@draco"
    username = "ubuntu"

```

```

}

os_disk {
  caching          = "ReadWrite"
  disk_size_gb     = 30
  name             = "webserver-osdisk"
  storage_account_type = "Standard_LRS"
  write_accelerator_enabled = false
}

source_image_reference {
  offer      = "0001-com-ubuntu-server-jammy"
  publisher = "canonical"
  sku       = "22_04-lts-gen2"
  version   = "latest"
}
}

# azurerm_nat_gateway.ngw["unir_ngw"]:
resource "azurerm_nat_gateway" "ngw" {
  id = "/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/natGateways/unir_ngw"
  idle_timeout_in_minutes = 4
  location                = "northeurope"
  name                    = "unir_ngw"
  resource_group_name     = "unir-arga2"
  resource_guid           = "c1c593e6-7e00-4359-9188-ae7c525acbe0"
  sku_name                = "Standard"
  tags                    = {
    "environment" = "unir"
    "terraform"   = "true"
  }
  zones = []
}

# azurerm_network_interface.wms_ani["webserver"]:
resource "azurerm_network_interface" "wms_ani" {
  accelerated_networking_enabled = false
  applied_dns_servers            = []
  dns_servers                    = []
  enable_accelerated_networking = false
  enable_ip_forwarding          = false
}

```

```

    id = "/subscriptions/c190437f-864c-4fab-
a46d-94b7dfc565d0/resourceGroups/unir-
arga2/providers/Microsoft.Network/networkInterfaces/webserver-ani"
    internal_domain_name_suffix =
"3qjwd2hsluwvdtrci0225atl.cfx.internal.cloudapp.net"
    ip_forwarding_enabled = false
    location = "northeurope"
    mac_address = "00-0D-3A-B4-6D-4C"
    name = "webserver-ani"
    private_ip_address = "10.0.0.4"
    private_ip_addresses = [
        "10.0.0.4",
    ]
    resource_group_name = "unir-arga2"
    tags = {
        "environment" = "unir"
        "terraform" = "true"
    }
    virtual_machine_id = "/subscriptions/c190437f-864c-4fab-
a46d-94b7dfc565d0/resourceGroups/unir-
arga2/providers/Microsoft.Compute/virtualMachines/webserver"

    ip_configuration {
        name = "webserver-ani-config"
        primary = true
        private_ip_address = "10.0.0.4"
        private_ip_address_allocation = "Dynamic"
        private_ip_address_version = "IPv4"
        public_ip_address_id = "/subscriptions/c190437f-864c-4fab-
a46d-94b7dfc565d0/resourceGroups/unir-
arga2/providers/Microsoft.Network/publicIPAddresses/webserver-public-ip"
        subnet_id = "/subscriptions/c190437f-864c-4fab-
a46d-94b7dfc565d0/resourceGroups/unir-
arga2/providers/Microsoft.Network/virtualNetworks/unir_vpc/subnets/pub_snet_
a"
    }
}

# azurerm_network_security_group.nsg["public"]:
resource "azurerm_network_security_group" "nsg" {
    id = "/subscriptions/c190437f-864c-4fab-a46d-
94b7dfc565d0/resourceGroups/unir-
arga2/providers/Microsoft.Network/networkSecurityGroups/public"

```

```

location          = "northeurope"
name              = "public"
resource_group_name = "unir-arga2"
security_rule     = [
    {
        access                = "Allow"
        description           = ""
        destination_address_prefix = ""
        destination_address_prefixes = [
            "0.0.0.0/0",
        ]
        destination_application_security_group_ids = []
        destination_port_range = "22"
        destination_port_ranges = []
        direction              = "Inbound"
        name                   = "unir-arga2-public-
a72b86ad9b512aeb5540e38f789edc72eb705168"
        priority               = 100
        protocol                = "Tcp"
        source_address_prefix   = ""
        source_address_prefixes = [
            "213.195.126.222/32",
        ]
        source_application_security_group_ids = []
        source_port_range        = "*"
        source_port_ranges       = []
    },
    {
        access                = "Allow"
        description           = ""
        destination_address_prefix = ""
        destination_address_prefixes = [
            "0.0.0.0/0",
        ]
        destination_application_security_group_ids = []
        destination_port_range = "80"
        destination_port_ranges = []
        direction              = "Inbound"
        name                   = "unir-arga2-public-
a7d53631b0ac6515341eb4d9fc2318aafe8af511"
        priority               = 101
        protocol                = "Tcp"
        source_address_prefix   = ""

```

```

        source_address_prefixes = [
            "213.195.126.222/32",
        ]
        source_application_security_group_ids = []
        source_port_range = "*"
        source_port_ranges = []
    },
    {
        access = "Allow"
        description = ""
        destination_address_prefix = ""
        destination_address_prefixes = [
            "0.0.0.0/0",
        ]
        destination_application_security_group_ids = []
        destination_port_range = "8080"
        destination_port_ranges = []
        direction = "Inbound"
        name = "unir-arga2-public-3d1e0387f8539b6b9fc67803909b5e17a64c247b"
        priority = 102
        protocol = "Tcp"
        source_address_prefix = ""
        source_address_prefixes = [
            "213.195.126.222/32",
        ]
        source_application_security_group_ids = []
        source_port_range = "*"
        source_port_ranges = []
    },
]
tags = {
    "environment" = "unir"
    "terraform" = "true"
}
}

# azurerm_network_security_rule.nsg["unir-arga2-public-3d1e0387f8539b6b9fc67803909b5e17a64c247b"]:
resource "azurerm_network_security_rule" "nsg" {
    access = "Allow"
    destination_address_prefixes = [
        "0.0.0.0/0",
    ]

```



```

    ]
    destination_application_security_group_ids = []
    destination_port_range                    = "8080"
    destination_port_ranges                   = []
    direction                                = "Inbound"
    id                                         = "/subscriptions/c190437f-
864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-
arga2/providers/Microsoft.Network/networkSecurityGroups/public/securityRules
/unir-arga2-public-3d1e0387f8539b6b9fc67803909b5e17a64c247b"
    name                                       = "unir-arga2-public-
3d1e0387f8539b6b9fc67803909b5e17a64c247b"
    network_security_group_name               = "public"
    priority                                  = 102
    protocol                                   = "Tcp"
    resource_group_name                       = "unir-arga2"
    source_address_prefixes                   = [
        "213.195.126.222/32",
    ]
    source_application_security_group_ids      = []
    source_port_range                         = "*"
    source_port_ranges                        = []
}

# azurerm_network_security_rule.nsgr["unir-arga2-public-
a72b86ad9b512aeb5540e38f789edc72eb705168"]:
resource "azurerm_network_security_rule" "nsgr" {
    access                                     = "Allow"
    destination_address_prefixes              = [
        "0.0.0.0/0",
    ]
    destination_application_security_group_ids = []
    destination_port_range                    = "22"
    destination_port_ranges                   = []
    direction                                = "Inbound"
    id                                         = "/subscriptions/c190437f-
864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-
arga2/providers/Microsoft.Network/networkSecurityGroups/public/securityRules
/unir-arga2-public-a72b86ad9b512aeb5540e38f789edc72eb705168"
    name                                       = "unir-arga2-public-
a72b86ad9b512aeb5540e38f789edc72eb705168"
    network_security_group_name               = "public"
    priority                                  = 100
    protocol                                   = "Tcp"

```

```

    resource_group_name                = "unir-arga2"
    source_address_prefixes            = [
        "213.195.126.222/32",
    ]
    source_application_security_group_ids = []
    source_port_range                  = "*"
    source_port_ranges                  = []
}

# azurerm_network_security_rule.nsg["unir-arga2-public-
a7d53631b0ac6515341eb4d9fc2318aafe8af511"]:
resource "azurerm_network_security_rule" "nsg" {
    access                        = "Allow"
    destination_address_prefixes = [
        "0.0.0.0/0",
    ]
    destination_application_security_group_ids = []
    destination_port_range                = "80"
    destination_port_ranges                = []
    direction                             = "Inbound"
    id                                     = "/subscriptions/c190437f-
864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-
arga2/providers/Microsoft.Network/networkSecurityGroups/public/securityRules
/unir-arga2-public-a7d53631b0ac6515341eb4d9fc2318aafe8af511"
    name                                = "unir-arga2-public-
a7d53631b0ac6515341eb4d9fc2318aafe8af511"
    network_security_group_name          = "public"
    priority                             = 101
    protocol                             = "Tcp"
    resource_group_name                  = "unir-arga2"
    source_address_prefixes              = [
        "213.195.126.222/32",
    ]
    source_application_security_group_ids = []
    source_port_range                    = "*"
    source_port_ranges                    = []
}

# azurerm_public_ip.ngw-ip["unir_ngw"]:
resource "azurerm_public_ip" "ngw-ip" {
    allocation_method      = "Static"
    ddos_protection_mode   = "VirtualNetworkInherited"
    id                     = "/subscriptions/c190437f-864c-4fab-a46d-

```

```

94b7dfc565d0/resourceGroups/unir-
arga2/providers/Microsoft.Network/publicIPAddresses/unir_ngw-ip"
    idle_timeout_in_minutes = 4
    ip_address              = "52.169.156.147"
    ip_tags                 = {}
    ip_version              = "IPv4"
    location                = "northeurope"
    name                    = "unir_ngw-ip"
    resource_group_name     = "unir-arga2"
    sku                     = "Standard"
    sku_tier                = "Regional"
    tags                    = {
        "environment" = "unir"
        "terraform"   = "true"
    }
    zones                  = []
}

# azurerm_public_ip.pub_ip["webserver"]:
resource "azurerm_public_ip" "pub_ip" {
    allocation_method      = "Dynamic"
    ddos_protection_mode   = "VirtualNetworkInherited"
    id                     = "/subscriptions/c190437f-864c-4fab-a46d-
94b7dfc565d0/resourceGroups/unir-
arga2/providers/Microsoft.Network/publicIPAddresses/webserver-public-ip"
    idle_timeout_in_minutes = 4
    ip_address              = "40.69.204.208"
    ip_tags                 = {}
    ip_version              = "IPv4"
    location                = "northeurope"
    name                    = "webserver-public-ip"
    resource_group_name     = "unir-arga2"
    sku                     = "Basic"
    sku_tier                = "Regional"
    tags                    = {
        "environment" = "unir"
        "terraform"   = "true"
    }
    zones                  = []
}

# azurerm_resource_group.rgs["unir-arga2"]:
resource "azurerm_resource_group" "rgs" {

```

```

    id      = "/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2"
    location = "northeurope"
    name     = "unir-arga2"
    tags     = {
        "environment" = "unir"
        "terraform"   = "true"
    }
}

# azurerm_subnet.snets["pub_snet_a"]:
resource "azurerm_subnet" "snets" {
    address_prefixes      = [
        "10.0.0.0/24",
    ]
    default_outbound_access_enabled = true
    enforce_private_link_endpoint_network_policies = false
    enforce_private_link_service_network_policies = false
    id =
"/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/virtualNetworks/unir_vpc/subnets/pub_snet_a"
    name = "pub_snet_a"
    private_endpoint_network_policies = "Enabled"
    private_endpoint_network_policies_enabled = true
    private_link_service_network_policies_enabled = true
    resource_group_name = "unir-arga2"
    service_endpoint_policy_ids = []
    service_endpoints = []
    virtual_network_name = "unir_vpc"
}

#
azurerm_subnet_network_security_group_association.sb_nsg["snet_pub_snet_a_ns_g_public"]:
resource "azurerm_subnet_network_security_group_association" "sb_nsg" {
    id = "/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/virtualNetworks/unir_vpc/subnets/pub_snet_a"
    network_security_group_id = "/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/networkSecurityGroups/public"
}

```

```

        subnet_id                = "/subscriptions/c190437f-864c-4fab-a46d-
94b7dfc565d0/resourceGroups/unir-
arga2/providers/Microsoft.Network/virtualNetworks/unir_vpc/subnets/pub_snet_
a"
    }

# azurerm_virtual_network.vpc["unir_vpc"]:
resource "azurerm_virtual_network" "vpc" {
    address_space                = [
        "10.0.0.0/16",
    ]
    dns_servers                  = []
    flow_timeout_in_minutes     = 0
    guid                         = "f06113ec-5df2-4a2d-8e71-1235ce7c135a"
    id                           = "/subscriptions/c190437f-864c-4fab-a46d-
94b7dfc565d0/resourceGroups/unir-
arga2/providers/Microsoft.Network/virtualNetworks/unir_vpc"
    location                     = "northeurope"
    name                         = "unir_vpc"
    resource_group_name         = "unir-arga2"
    subnet                       = [
        {
            address_prefix = "10.0.0.0/24"
            id               = "/subscriptions/c190437f-864c-4fab-a46d-
94b7dfc565d0/resourceGroups/unir-
arga2/providers/Microsoft.Network/virtualNetworks/unir_vpc/subnets/pub_snet_
a"
            name            = "pub_snet_a"
            security_group = "/subscriptions/c190437f-864c-4fab-a46d-
94b7dfc565d0/resourceGroups/unir-
arga2/providers/Microsoft.Network/networkSecurityGroups/public"
        },
    ]
    tags                         = {
        "environment" = "unir"
        "terraform"   = "true"
    }
}

# local_file.deployer_vars:
resource "local_file" "deployer_vars" {
    content                = <<-EOT
        registry:

```

```

    admin_password: "YYYY"
    admin_username: "acrunir"
    source_repo: "registry.hub.docker.com/nginx:latest"
    source_img: "nginx:latest"
    dest_repo: "acrunir.azurecr.io"
    dest_img: "nginx:casopractico2"
EOT
content_base64sha256 = "SWvKazTCI9JJ5KSWyTslrXEzZOWn2VYAcijrj5ifgEg="
content_base64sha512 =
"FD3p0J0YvcOluUBcNpUdnX3C4yca6Q89hIIwak5yAkFYaSufkQJJ5FRu+c1Exz3Lw/UgzB9YxS+
Or9EzSBiLDQ=="
content_md5 = "de665837e79a5eb6d12214d0ff99979e"
content_sha1 = "de9686b0a87ca980661ac502ff2e6e9a36e1f974"
content_sha256 =
"496bca6b34c223d249e4a496c93b35ad713364e5a7d956007228eb8f989f8048"
content_sha512 =
"143de9389d18bdc3a5b9405c36951d9d7dc2e3271ae90f3d8482306a4e72024158692b9f910
249e4546ef9c944c73dcbc3f520cc1f58c52fb4afd13348188b0d"
directory_permission = "0777"
file_permission = "0777"
filename =
"../ansible/inventories/local/group_vars/local/main.yml"
  id = "de9686b0a87ca980661ac502ff2e6e9a36e1f974"
}

# local_file.remote_inventory:
resource "local_file" "remote_inventory" {
  content = <<-EOT
    azure:
      hosts:
        tfnode:
          ansible_host: 40.69.204.208
          ansible_port: 22
EOT
  content_base64sha256 = "MZPgkL0JEZ/HQ900JclZv3Rvekx1Te3Oivs4qXu0N0g="
  content_base64sha512 =
"I5ypjsn1H6eSaG5XRBduy03y/KoZb1882o18JoY4YX2AIDDDWAKXhBhYsc4NyRHODHfODEYu+Pf
TMiUrMg99LA=="
  content_md5 = "d4a24d3bdb3fe9c55e1ee65813813983"
  content_sha1 = "e4bbeb1c88df65ca55f1e72dc3481dd1e385fa28"
  content_sha256 =
"3193e090bd09119fc743dd3425c959bf746f7a4c754dedce8afb38a97bb43748"
  content_sha512 =

```

```

"239ca98ec9f51fa792686e5744176ecb4df2fcaa196f5f3cda8d7c268638617d802030dd580
297841858b1ce0dc911e80c77ce0c462ef8f7d332252b320f7d2c"

    directory_permission = "0777"
    file_permission      = "0777"
    filename              = "../ansible/inventories/azure/hosts.yml"
    id                    = "e4bbeb1c88df65ca55f1e72dc3481dd1e385fa28"
}

# local_file.remote_vars:
resource "local_file" "remote_vars" {
    content = <<-EOT
        packages:
            - podman
            - tree

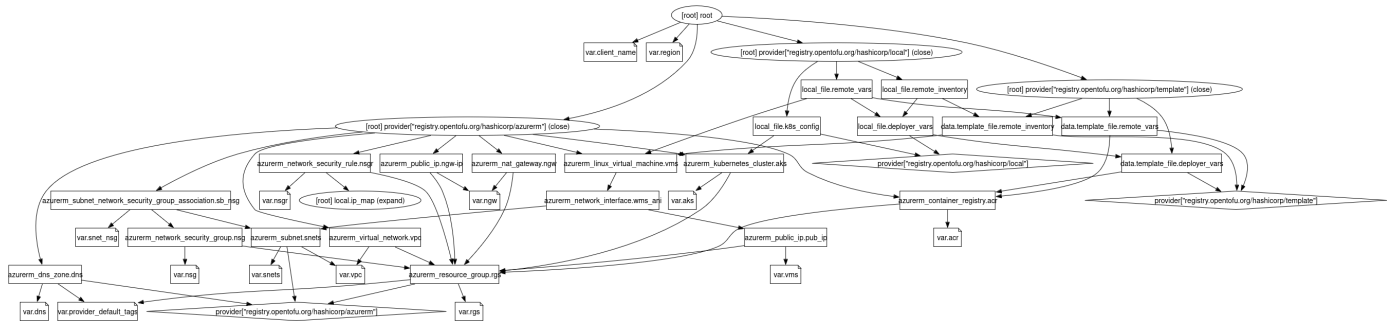
        container:
            localPath: /data
            containerPath: /usr/share/nginx/html

        registry:
            admin_password: "YYYY"
            admin_username: "acrunir"
            source_repo: "registry.hub.docker.com/nginx:latest"
            source_img: "nginx:latest"
            dest_repo: "acrunir.azurecr.io"
            dest_img: "nginx:casopractico2"
    EOT
    content_base64sha256 = "fBAowNMmaRxctBWnY8n9BRaPWKmdJFMUN8jF1GOBXhY="
    content_base64sha512 =
"aJln29ofKCHeiomdpPvkyZGfbVVkdEIJV/V/6pQa/auV0hcT1aanpDnJjFGbf/imct7NZmMKe3n
irKjQaA+v1A=="
    content_md5          = "ca19ecc144025c3d3d123bb7e6e653b5"
    content_sha1         = "7f184eeef05f491289681e4893638831c9e02a6a"
    content_sha256       =
"7c1028c0d326691c5cb415a763c9fd05168f58a99d24531437c8c5d463815e16"
    content_sha512       =
"689967dbda1f2821de8a899da4fbe4c9919f6d556474420957f57fea941afdab95d21713d5a
6a7a439c98c519b7ff8a672dec66630a7b79e2aca8d0680fafd4"
    directory_permission = "0777"
    file_permission      = "0777"
    filename              =
"../ansible/inventories/azure/group_vars/azure/main.yml"
    id                    = "7f184eeef05f491289681e4893638831c9e02a6a"

```

```
}
```

Y un claro diagrama para ilustrarlo todo `tofu graph -draw-cycles | dot -Tsvg > graph.svg`



## Configuración de la infra

### Preliminares

Para la configuración automática de la infra de forma automática se ha decidido que sea el propio gestor de la infraestructura el que la configure. Es decir *Terraform* llamará a *Ansible*. Para ello se han pensado las dos siguientes alternativas

- Usar un inventario dinámico de azure
- Generar un inventario estático dinamicamente

### Código Ansible

La estructura general de la solución de *Ansible* es la estándar en estos casos. Se han creado dos inventarios uno para los recursos que se ejecutan desde el nodo instalador, local, y el remoto que se corresponde con la instancia creada en Azure.

Se utilizan templates para dar un poco de dinamismo a la solución y parte de la configuración.

```
├─ aks.yml
├─ inventories
│   └─ azure
│       └─ group_vars
│           └─ azure
│               └─ main.yml
│           └─ hosts.yml
│       └─ local
│           └─ group_vars
│               └─ local
│                   └─ k8s.yml
│                   └─ main.yml
│                   └─ hosts.yml
├─ registry.yml
```



```

├─ roles
│   ├── acr
│   │   ├── files
│   │   │   ├── a.html
│   │   │   └─ Dockerfile
│   │   └─ tasks
│   │       └─ main.yml
│   ├── aks
│   │   ├── files
│   │   │   ├── argocd
│   │   │   │   ├── loadbalancer.yml
│   │   │   │   └─ namespace.yml
│   │   │   ├── config.yml
│   │   │   └─ wp
│   │   │       ├── mysql-deployment.yml
│   │   │       ├── mysql-pvc.yml
│   │   │       ├── mysql-service.yml
│   │   │       ├── namespace.yml
│   │   │       └─ secret.yml
│   │   └─ tasks
│   │       ├── argocd.yml
│   │       ├── main.yml
│   │       └─ wp.yml
│   └─ templates
│       ├── mysql-deployment.yaml.j2
│       ├── registry-credentials.yaml.j2
│       └─ wordpress-deployment.yaml.j2
└─ podman
    └─ tasks
        └─ main.yml
└─ tfnode.yml

```

## Inventario dinámico

Ansible tiene un plugin para el inventariado dinámico de recursos en azure. La jugada sería una vez se ha creado la infra en *terraform* lanzar la ejecución del *playbook* de *ansible* utilizando este inventario dinámico `ansible-playbook tfnode.yml -i inventories/myazure_rm.yml`

El problema es que no hemos podido hacerlo funcionar experimentando los problemas que se describen en esta [issue](#) en *github*

## Generar un inventario estático dinamicamente

La jugada es un poco más artificiosa y consiste en que *terraform* genere el inventario dinámicamente ya que genera las credenciales y conoce las ip para conectarse

En nuestro caso, se ha decidido no complicarse mucho ya que script que se utiliza daba muchos problemas a la hora de realizar el registro contra *Azure* y generar el inventario.

## Configuración de la infra utilizando *Terraform* y *Ansible*

El script de *terraform installer.tf* será el disparador del proceso de configuración de la infraestructura. Está estará dividida en varias fases

- Configurar *ACR*
- Configurar *WebServer*
- Instalar *AKS*

Y hay que recalcar que provisionamos y configuramos nuestra infraestructura **automáticamente** sin intervención humana.

### Installer

Simplemente utiliza una plantilla para generar el inventario de *Ansible* y ejecuta el *playbook*

```
data "template_file" "deployer_vars" {
  template = file("../resources/templates/deployer_vars.tpl")
  vars = {
    admin_password = azurerm_container_registry.acr["acr"].admin_password
    admin_username = azurerm_container_registry.acr["acr"].admin_username
    dest_repo      = azurerm_container_registry.acr["acr"].login_server
  }
  depends_on = [
    azurerm_container_registry.acr["acr"]
  ]
}

resource "local_file" "deployer_vars" {
  content  = data.template_file.deployer_vars.rendered
  filename = "../ansible/inventories/local/group_vars/local/main.yml"

  provisioner "local-exec" {
    working_dir = "../ansible"
    command     = "ansible-playbook -v -i inventories/local/hosts.yml -u
dani -b registry.yml"
  }

  depends_on = [
    azurerm_container_registry.acr["acr"]
  ]
}
```

```

}

data "template_file" "remote_vars" {
  template = file("./resources/templates/remote_vars.tpl")
  vars = {
    admin_password = azurerm_container_registry.acr["acr"].admin_password
    admin_username = azurerm_container_registry.acr["acr"].admin_username
    dest_repo      = azurerm_container_registry.acr["acr"].login_server
  }
}

resource "local_file" "remote_vars" {
  content  = data.template_file.remote_vars.rendered
  filename = "../ansible/inventories/azure/group_vars/azure/main.yml"

  depends_on = [
    local_file.deployer_vars,
    azurerm_linux_virtual_machine.vms["webserver"],
    azurerm_public_ip.pub_ip["webserver"]
  ]
}

data "template_file" "remote_inventory" {
  #"azurerm_linux_virtual_machine" "vms"

  #azurerm_public_ip.pub_ip["webserver"]
  template = file("./resources/templates/remote_inventory.tpl")
  vars = {
    tfhost = azurerm_public_ip.pub_ip["webserver"].ip_address
  }
  depends_on = [
    azurerm_linux_virtual_machine.vms["webserver"],
    azurerm_public_ip.pub_ip["webserver"]
  ]
}

resource "local_file" "remote_inventory" {

```

```

content = data.template_file.remote_inventory.rendered
filename = "../ansible/inventories/azure/hosts.yml"

provisioner "local-exec" {
    working_dir = "../ansible"
    command     = "ansible-playbook -v -i inventories/azure/hosts.yml -u
ubuntu tfnode.yml"
}
depends_on = [
    local_file.deployer_vars,
    azurerm_linux_virtual_machine.vms["webserver"],
    azurerm_public_ip.pub_ip["webserver"]

]
}

resource "local_file" "k8s_config" {
    content = azurerm_kubernetes_cluster.aks["aks"].kube_config_raw
    filename = "../ansible/roles/aks/files/config.yml"

    provisioner "local-exec" {
        working_dir = "../ansible"
        command     = "ansible-playbook -v -i inventories/local/hosts.yml -u
dani aks.yml"
    }
    depends_on = [
        azurerm_kubernetes_cluster.aks["aks"]
    ]
}

```

## Configuración del ACR

Este es digamos el primer paso de la configuración. En el desplegamos una imagen *Docker* a nuestro repositorio ACR. La imagen *Docker* no es más que un *Nginx* con un fichero *HTML* personalizado.

Nuestro *Dockerfile* se ha reducido a la mínima expresión mostrandose a continuación

```

FROM nginx:latest
COPY a.html /usr/share/nginx/html

```

El orden de ejecución es el siguiente

1. El plan de *Terraform* genera la configuración de *Ansible* para nuestro *playbook*
2. El plan de *Terraform* ejecuta en **local** el *playbook* de *Ansible*.

### 3. El playbook ejecuta las siguientes acciones

1. Registrase en ACR
2. Construir la imagen y subirla al ACR

La plantilla que utilizará *Terrform* para generar la configuración se muestran a continuación.

```
registry:
  admin_password: "${admin_password}"
  admin_username: "${admin_username}"
  source_repo: "registry.hub.docker.com/nginx:latest"
  source_img: "nginx:latest"
  dest_repo: "${dest_repo}"
  dest_img: "nginx:latest"
```

Que luego genera en *Ansible* la siguiente estructura

```
{
  "_meta": {
    "hostvars": {
      "localhost": {
        "ansible_connection": "local",
        "ansible_host": "localhost",
        "ansible_port": 22,
        "registry": {
          "admin_password": "YYYY",
          "admin_username": "XXXX",
          "dest_img": "nginx:latest",
          "dest_repo": "acrunir.azurecr.io",
          "source_img": "nginx:latest",
          "source_repo": "registry.hub.docker.com/nginx:latest"
        }
      }
    }
  },
  "all": {
    "children": [
      "ungrouped",
      "local"
    ]
  },
  "local": {
    "hosts": [
      "localhost"
    ]
  }
}
```

```
}  
  
}
```

Una vez generada la configuración, el siguiente paso es ejecutar el *playbook* de *Ansible* que se muestra a continuación

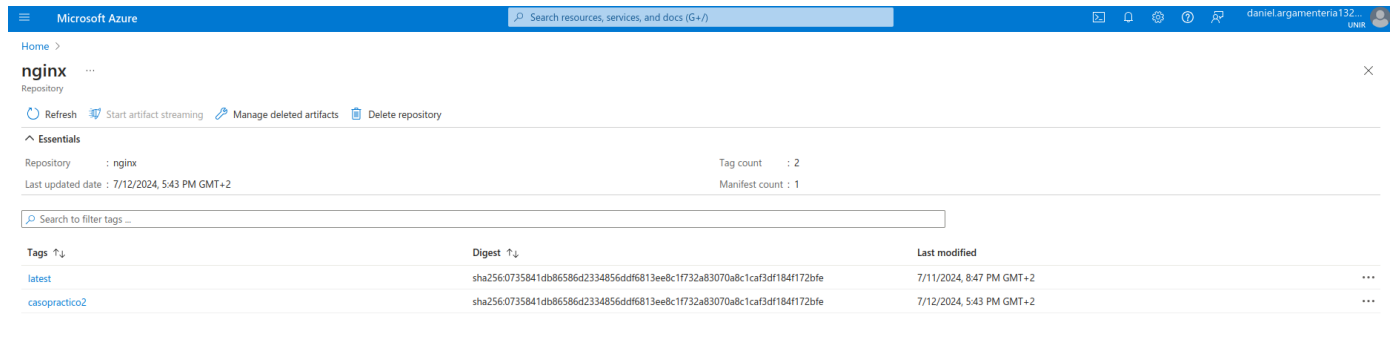
```
- name: Log into private registry and force re-authorization  
  docker_login:  
    registry: "{{ registry.dest_repo }}"  
    username: "{{ registry.admin_username }}"  
    password: "{{ registry.admin_password }}"  
    reauthorize: yes  
  
- name: Build an image and push it to a private repo  
  docker_image:  
    build:  
      path: "{{ role_path }}/files"  
    name: "{{ registry.dest_repo }}/{{ registry.dest_img }}"  
    push: yes  
    source: build
```

Las imágenes siguientes muestran nuestro ACR montado y configurado

The first screenshot shows the 'Overview' page of the 'unir' Container Registry in the Azure portal. It displays the resource group 'unir-arga2', location 'North Europe', and subscription 'Azure for Students'. The 'Login server' is 'acrunir.azurecr.io'. The 'Provisioning state' is 'Succeeded'.

The second screenshot shows the 'Repositories' page for the 'unir' registry. It lists several repositories: 'app', 'mysql', 'nginx', 'unir', and 'wordpress'. The 'unir' repository is selected, showing its 'Tags' and 'Digest' columns. The 'Tags' column lists various image versions, and the 'Digest' column shows the corresponding SHA hashes. The 'Last modified' column indicates the date and time of the last update for each tag.

## La imagen de nginx generada



## Configuración del WebServer

La configuración del servidor web se realiza mediante *Ansible* desde un plan de *Terraform*, que primero genera el inventario y luego ejecuta el comando de *Ansible*.

El orden de ejecución es el siguiente

1. El plan de *Terraform* genera el inventario de *Ansible*
2. El plan de *Terraform* ejecuta el *playbook* de *Ansible*.
3. El *playbook* ejecuta las siguientes acciones
  1. Actualizar los paquetes
  2. Instalar podman
  3. Obtener la imagen del ACR
  4. Arrancar el contenedor
  5. Probar la URL

En nuestro caso la plantilla del inventario es muy sencilla.

```
azure:
  hosts:
    tfnode:
      ansible_host: ${tfhost}
      ansible_port: 22
```

Una vez se ejecuta el plan de *Terraform* se genera automáticamente una inventario en *Ansible* con la siguiente estructura.

```
{
  "_meta": {
    "hostvars": {
      "tfnode": {
        "ansible_host": "13.74.176.255",
        "ansible_port": 22,
        "container": {
          "containerPath": "/usr/share/nginx/html",
```

```

        "localPath": "/data"
    },
    "packages": [
        "podman",
        "tree"
    ],
    "registry": {
        "admin_password": "YYYY",
        "admin_username": "XXXX",
        "dest_img": "nginx:latest",
        "dest_repo": "acrunir.azurecr.io",
        "source_img": "nginx:latest",
        "source_repo": "registry.hub.docker.com/nginx:latest"
    }
}

},
"all": {
    "children": [
        "ungrouped",
        "azure"
    ]
},
"azure": {
    "hosts": [
        "tfnode"
    ]
}
}

```

Siendo nuestro rol de *Ansible* para crear nuestro servidor web el siguiente

```

- name: Update packages
  become: true
  apt:
    update_cache: yes

- name: Install stuff
  become: true
  ansible.builtin.apt:
    name: "{{ item }}"
  with_items:
    - "{{ packages }}"

```



```
- name: Get podman version
  ansible.builtin.command: podman --version
  register: podman

- name: Print podman version
  ansible.builtin.debug:
    msg:
      - "Podman running on version {{ podman.stdout }}"

- name: Create statefull vol
  become: true
  ansible.builtin.file:
    path: /data
    state: directory
    owner: ubuntu
    group: ubuntu
    mode: '0755'

- name: Dummy Content
  ansible.builtin.shell: echo $(date) >> /data/date

- name: Login to ACR
  containers.podman.podman_login:
    authfile: ~/.config/containers/auth.json
    username: "{{ registry.admin_username }}"
    password: "{{ registry.admin_password }}"
    registry: "{{ registry.dest_repo }}"

- name: Run container
  containers.podman.podman_container:
    name: "webserver"
    authfile: ~/.config/containers/auth.json
    image: "{{ registry.dest_repo }}/{{ registry.dest_img }}"
    volume: "{{ container.localPath }}:{{ container.containerPath }}"
    ports:
      - 8080:80
    state: started

- name: Logout to ACR
  containers.podman.podman_logout:
    authfile: ~/.config/containers/auth.json
    all: true
```

```

- name: Get URL
  ansible.builtin.uri:
    url: http://{{ ansible_host }}:8080/date
    return_content: true
    register: content

- name: Show url content
  ansible.builtin.debug:
    msg: |-
      URL: "{{ content.url }}"
      Status: {{ content.status }}
      Content: {{ content.content }}

```

## Salida de la ejecución de la configuración

```

azurerm_resource_group.rgs["unir-arga2"]: Refreshing state...
[id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2]

azurerm_subnet.snets["pub_snet_a"]: Refreshing state...
[id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/virtualNetworks/unir_vpc/subnets/pub_snet_a]

azurerm_network_security_rule.nsgr["allow_ssh_public"]: Refreshing state...
[id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/networkSecurityGroups/public/securityRules/allow_ssh_public]

azurerm_network_security_rule.nsgr["allow_http_public"]: Refreshing state...
[id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/networkSecurityGroups/public/securityRules/allow_http_public]

azurerm_virtual_network.vpc["unir_vpc"]: Refreshing state...
[id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/virtualNetworks/unir_vpc]

azurerm_nat_gateway.ngw["unir_ngw"]: Refreshing state...
[id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/natGateways/unir_ngw]

azurerm_public_ip.ngw-ip["unir_ngw"]: Refreshing state...
[id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/publicIPAddresses/unir_ngw-ip]

azurerm_public_ip.pub_ip["webserver"]: Refreshing state...
[id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-arga2/providers/Microsoft.Network/publicIPAddresses/webserver-public-ip]

```

```
azurerm_container_registry.acr["acr"]: Refreshing state...
[id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-
arga2/providers/Microsoft.ContainerRegistry/registries/acrunir]
azurerm_network_security_group.nsg["public"]: Refreshing state...
[id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-
arga2/providers/Microsoft.Network/networkSecurityGroups/public]
azurerm_subnet_network_security_group_association.sb_nsg["snet_pub_snet_a_ns
g_public"]: Refreshing state... [id=/subscriptions/c190437f-864c-4fab-a46d-
94b7dfc565d0/resourceGroups/unir-
arga2/providers/Microsoft.Network/virtualNetworks/unir_vpc/subnets/pub_snet_
a]
data.template_file.remote_inventory: Reading...
data.template_file.remote_inventory: Read complete after 0s
[id=f991173e7f9bf3e27f015298125e9513a7ebd77879d08951c986b3d01a2f8888]
azurerm_network_interface.wms_anip["webserver"]: Refreshing state...
[id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-
arga2/providers/Microsoft.Network/networkInterfaces/webserver-anip]
local_file.remote_inventory: Refreshing state...
[id=b2d2787a94178b9313eb9a24fcb6f4277ebef3ab]
azurerm_linux_virtual_machine.vms["webserver"]: Refreshing state...
[id=/subscriptions/c190437f-864c-4fab-a46d-94b7dfc565d0/resourceGroups/unir-
arga2/providers/Microsoft.Compute/virtualMachines/webserver]
data.template_file.deployer_vars: Reading...
data.template_file.remote_vars: Reading...
local_file.acrdata: Refreshing state...
[id=154f5b2aebc6090b9bd04997bd83804d57c9137d]
data.template_file.deployer_vars: Read complete after 0s
[id=e8a740fbbe4e527eb8c4f568733d7d66c8b66df76acad85aa5b65d3863cddd63]
data.template_file.remote_vars: Read complete after 0s
[id=cc8044b76030e9f467facf3e2e68e60659cb9a6cbaac36a9cb42c3de79fe35a1]
local_file.deployer_vars: Refreshing state...
[id=71eafdf7794e26ccc0b3a8bbf66358bb113148c2]
local_file.remote_vars: Refreshing state...
[id=4e4e38f0089a759f70e10ddf80513e5d2b3b0eca]
```

OpenTofu used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:

- + create

OpenTofu will perform the following actions:

```
# local_file.deployer_vars will be created
+ resource "local_file" "deployer_vars" {
```

```

+ content                                = <<-EOT
  registry:
    admin_password: "XXXX"
    admin_username: "YYYY"
    source_repo: "registry.hub.docker.com/nginx:latest"
    source_img: "nginx:latest"
    dest_repo: "acrunir.azurecr.io"
    dest_img: "nginx:latest"

  EOT
+ content_base64sha256 = (known after apply)
+ content_base64sha512 = (known after apply)
+ content_md5          = (known after apply)
+ content_sha1         = (known after apply)
+ content_sha256       = (known after apply)
+ content_sha512       = (known after apply)
+ directory_permission = "0777"
+ file_permission      = "0777"
+ filename             =

"../ansible/inventories/local/group_vars/local/main.yml"
+ id                   = (known after apply)
}

# local_file.remote_inventory will be created
+ resource "local_file" "remote_inventory" {
  + content                                = <<-EOT
    azure:
      hosts:
        tfnode:
          ansible_host: 13.74.176.255
          ansible_port: 22

    EOT
  + content_base64sha256 = (known after apply)
  + content_base64sha512 = (known after apply)
  + content_md5          = (known after apply)
  + content_sha1         = (known after apply)
  + content_sha256       = (known after apply)
  + content_sha512       = (known after apply)
  + directory_permission = "0777"
  + file_permission      = "0777"
  + filename             = "../ansible/inventories/azure/hosts.yml"
  + id                   = (known after apply)
}

```

```

# local_file.remote_vars will be created
+ resource "local_file" "remote_vars" {
  + content              = <<-EOT
    packages:
      - podman
      - tree

    container:
      localPath: /data
      containerPath: /usr/share/nginx/html

    registry:
      admin_password: "XXXX"
      admin_username: "YYYY"
      source_repo: "registry.hub.docker.com/nginx:latest"
      source_img: "nginx:latest"
      dest_repo: "acrunir.azurecr.io"
      dest_img: "nginx:latest"

  EOT
  + content_base64sha256 = (known after apply)
  + content_base64sha512 = (known after apply)
  + content_md5          = (known after apply)
  + content_sha1         = (known after apply)
  + content_sha256       = (known after apply)
  + content_sha512       = (known after apply)
  + directory_permission = "0777"
  + file_permission      = "0777"
  + filename             =

"../ansible/inventories/azure/group_vars/azure/main.yml"
  + id                  = (known after apply)
}

```

Plan: 3 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?

OpenTofu will perform the actions described above.

Only 'yes' will be accepted to approve.

Enter a value: yes

local\_file.deployer\_vars: Creating...

local\_file.deployer\_vars: Provisioning with 'local-exec'...

local\_file.deployer\_vars (local-exec): Executing: ["/bin/sh" "-c" "ansible-

```
playbook -v -i inventories/local/hosts.yml -u dani -b registry.yml"]
local_file.deployer_vars (local-exec): No config file found; using defaults

local_file.deployer_vars (local-exec): PLAY [localhost]
*****

local_file.deployer_vars (local-exec): TASK [Gathering Facts]
*****

local_file.deployer_vars (local-exec): [WARNING]: Platform linux on host
localhost is using the discovered Python
local_file.deployer_vars (local-exec): interpreter at /usr/bin/python3.12,
but future installation of another Python
local_file.deployer_vars (local-exec): interpreter could change the meaning
of that path. See
local_file.deployer_vars (local-exec): https://docs.ansible.com/ansible-
local_file.deployer_vars (local-exec):
core/2.17/reference_appendices/interpreter_discovery.html for more
information.
local_file.deployer_vars (local-exec): ok: [localhost]

local_file.deployer_vars (local-exec): TASK [acr : Log into private registry
and force re-authorization] *****
local_file.deployer_vars (local-exec): changed: [localhost] => {"changed":
true, "login_result": {"IdentityToken": "", "Status": "Login Succeeded"}}

local_file.deployer_vars (local-exec): TASK [acr : Build an image and push it
to a private repo] *****
local_file.deployer_vars (local-exec): ok: [localhost] => {"actions":
["Pushed image acrunir.azurecr.io/nginx to
acrunir.azurecr.io/nginx:casopractico2"], "changed": false, "image":
{"Architecture": "amd64", "Author": "", "Comment": "", "Config":
{"ArgsEscaped": true, "AttachStderr": false, "AttachStdin": false, "AttachStdout": false, "Cmd":
["nginx", "-g", "daemon off;"], "Domainname": "", "Entrypoint": ["/docker-
entrypoint.sh"], "Env":
["PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin",
"NGINX_VERSION=1.27.0", "NJS_VER
SION=0.8.4", "NJS_RELEASE=2~bookworm", "PKG_RELEASE=2~bookworm"],
"ExposedPorts": {"80/tcp": {}}, "Hostname": "", "Image":
"sha256:ffffc90d343cbcb01a5032edac86db5998c536cd0a366514121a45c6723765c",
"Labels": {"maintainer": "NGINX Docker Maintainers <docker-maint@nginx.com
>"}, "OnBuild": null, "OpenStdin": false, "StdinOnce": false, "StopSignal":
"SIGQUIT", "Tty": false, "User": "", "Volumes": null, "WorkingDir": ""},
```

```
"Created": "2024-07-11T18:46:54.758761849Z", "DockerVersion": "27.0.3",
"GraphDriver": {"Data": {"LowerDir": "/var/lib/dock
er/overlay2/258e3bbe89b5578ddf08900095b227301e6033c7c2ad216621be8025c6bb18ab
/diff:/var/lib/docker/overlay2/73e17abdbb2867b7b0306cb87fa3ffc89b62414eb3736d
1be333df274eb7362/diff:/var/lib/docker/overlay2/ce284eac98e12f667768fdf8cddb1
1f90086cabbfa0daeeec8749af8c3b623a6/diff:
/var/lib/docker/overlay2/1a51e2dee7720f5409cf2dba0164537c93a83bb7b397ab62e85
1a921c246db33/diff:/var/lib/docker/overlay2/8211cefae52a61695d649ff334e092f081
2eaa903fe5caf55e51b01e8f7c567f/diff:/var/lib/docker/overlay2/74ff2ec56456d0f06
8f72e36d5505db5c17be49c05e824948c53a1501
4011794/diff:/var/lib/docker/overlay2/59802c0296bc288883c3a3fc0536ae7e780bbd7
3ee0dfafbdcae63cd8c72dc29/diff", "MergedDir":
"/var/lib/docker/overlay2/be8aded44ae885e894e40c3e0b03554397de649adeb4d6cdfa
8f2a0c1bbe8e79/merged", "UpperDir": "/var/lib/docker/overlay2/be8aded44ae
885e894e40c3e0b03554397de649adeb4d6cdfa8f2a0c1bbe8e79/diff", "WorkDir":
"/var/lib/docker/overlay2/be8aded44ae885e894e40c3e0b03554397de649adeb4d6cdfa
8f2a0c1bbe8e79/work"}, "Name": "overlay2"}, "Id":
"sha256:49cb80b89f4db979c82074ab9b819cfe1073979c0a29a95b76433473916ece83",
  "Metadata": {"LastTagTime": "2024-07-12T17:43:41.381775162+02:00"}, "Os":
"linux", "Parent":
"sha256:ffffc90d343cbcb01a5032edac86db5998c536cd0a366514121a45c6723765c",
"RepoDigests":
["acrunir.azurecr.io/nginx@sha256:0735841db86586d2334856ddf6813ee8c1f732a830
70a8c1caf3df
184f172bfe"], "RepoTags": ["acrunir.azurecr.io/nginx:casopractico2",
"acrunir.azurecr.io/nginx:latest"], "RootFS": {"Layers":
["sha256:32148f9f6c5aadfa167ee7b146b9703c59307049d68b090c19db019fd15c5406",
"sha256:32cfaf91376fefc4934558561815920002a94dffaf52bc67b7382ea7869553
b6",
"sha256:933a3ce2c78a89e4646f6314e7e5b10ffd5d7f4d72ecf7a320d5a3e110f1e146",
"sha256:7190c87a0e8aa018977179a7cbd6641a7375b779d6343da75e64a7087fc6b6b6",
"sha256:92d0d4e970195d93096f50d58d18073db14c1dee98ee544eb0ae7c280dd8c783",
"sha256:0c6c257920c89c180f2268de1db0125753
1eb1aa719b58a33b8962b67b2036f4",
"sha256:56b6d3be75f9f95b71571e4deb8437aed91c26195fcb679207818bee6c9e3c78",
"sha256:975400489662a414698ae23c5fd7b2e574f3e2071ef6b3d6c978ae81ad031903"],
"Type": "layers"}, "Size": 187574143, "push_status": null}}
```

```
local_file.deployer_vars (local-exec): PLAY RECAP
```

```
*****
```

```
local_file.deployer_vars (local-exec): localhost : ok=3
```

```
changed=1    unreachable=0    failed=0    skipped=0    rescued=0
```

```
ignored=0
```



```
local_file.deployer_vars: Creation complete after 6s
[id=de9686b0a87ca980661ac502ff2e6e9a36elf974]
local_file.remote_inventory: Creating...
local_file.remote_vars: Creating...
local_file.remote_vars: Creation complete after 0s
[id=7f184eeef05f491289681e4893638831c9e02a6a]
local_file.remote_inventory: Provisioning with 'local-exec'...
local_file.remote_inventory (local-exec): Executing: ["/bin/sh" "-c"
"ansible-playbook -v -i inventories/azure/hosts.yml -u ubuntu tfnode.yml"]
local_file.remote_inventory (local-exec): No config file found; using defaults

local_file.remote_inventory (local-exec): PLAY [tfnode]
*****

local_file.remote_inventory (local-exec): TASK [Gathering Facts]
*****

local_file.remote_inventory (local-exec): [WARNING]: Platform linux on host
tfnode is using the discovered Python
local_file.remote_inventory (local-exec): interpreter at /usr/bin/python3.10,
but future installation of another Python
local_file.remote_inventory (local-exec): interpreter could change the
meaning of that path. See
local_file.remote_inventory (local-exec): https://docs.ansible.com/ansible-
local_file.remote_inventory (local-exec):
core/2.17/reference_appendices/interpreter_discovery.html for more
information.
local_file.remote_inventory (local-exec): ok: [tfnode]

local_file.remote_inventory (local-exec): TASK [podman : Update packages]
*****

local_file.remote_inventory (local-exec): changed: [tfnode] =>
{"cache_update_time": 1720806583, "cache_updated": true, "changed": true}

local_file.remote_inventory (local-exec): TASK [podman : Install stuff]
*****

local_file.remote_inventory: Still creating... [10s elapsed]
local_file.remote_inventory (local-exec): ok: [tfnode] => (item=podman) =>
{"ansible_loop_var": "item", "cache_update_time": 1720806583,
"cache_updated": false, "changed": false, "item": "podman"}
local_file.remote_inventory (local-exec): ok: [tfnode] => (item=tree) =>
{"ansible_loop_var": "item", "cache_update_time": 1720806583,
"cache_updated": false, "changed": false, "item": "tree"}
```



```

local_file.remote_inventory (local-exec): TASK [podman : Get podman version]
*****

local_file.remote_inventory (local-exec): changed: [tfnode] => {"changed":
true, "cmd": ["podman", "--version"], "delta": "0:00:00.030402", "end":
"2024-07-12 17:49:49.480896", "msg": "", "rc": 0, "start": "2024-07-12
17:49:49.450494", "stderr": "", "stderr_lines": [], "s
tdout": "podman version 3.4.4", "stdout_lines": ["podman version 3.4.4"]}

local_file.remote_inventory (local-exec): TASK [podman : Print podman
version] *****

local_file.remote_inventory (local-exec): ok: [tfnode] => {
local_file.remote_inventory (local-exec):      "msg": [
local_file.remote_inventory (local-exec):      "Podman running on version
podman version 3.4.4"
local_file.remote_inventory (local-exec):      ]
local_file.remote_inventory (local-exec): }

local_file.remote_inventory (local-exec): TASK [podman : Create statefull
vol] *****

local_file.remote_inventory (local-exec): ok: [tfnode] => {"changed": false,
"gid": 1000, "group": "ubuntu", "mode": "0755", "owner": "ubuntu", "path":
"/data", "size": 4096, "state": "directory", "uid": 1000}

local_file.remote_inventory (local-exec): TASK [podman : Dummy Conntent]
*****

local_file.remote_inventory (local-exec): changed: [tfnode] => {"changed":
true, "cmd": "echo $(date) >> /data/date", "delta": "0:00:00.005289", "end":
"2024-07-12 17:49:52.057384", "msg": "", "rc": 0, "start": "2024-07-12
17:49:52.052095", "stderr": "", "stderr_lines": [
], "stdout": "", "stdout_lines": []}

local_file.remote_inventory (local-exec): TASK [podman : Login to ACR]
*****

local_file.remote_inventory (local-exec): changed: [tfnode] => {"changed":
true, "stderr": "", "stderr_lines": [], "stdout": "Login Succeeded!\n",
"stdout_lines": ["Login Succeeded!"]}

local_file.remote_inventory (local-exec): TASK [podman : Run container]
*****

local_file.remote_inventory (local-exec): ok: [tfnode] => {"actions": [],
"changed": false, "container": {"AppArmorProfile": "", "Args": ["nginx", "-
g", "daemon off;"], "BoundingCaps": ["CAP_CHOWN", "CAP_DAC_OVERRIDE",

```

```
"CAP_FOWNER", "CAP_FSETID", "CAP_KILL", "CAP_NET_BIND
_SERVICE", "CAP_SETFCAP", "CAP_SETGID", "CAP_SETPCAP", "CAP_SETUID",
"CAP_SYS_CHROOT"], "Config": {"Annotations": {"io.container.manager":
"libpod", "io.kubernetes.cri-o.Created": "2024-07-12T17:18:45.824634462Z",
"io.kubernetes.cri-o.TTY": "false", "io.podman.annotations
.autoremove": "FALSE", "io.podman.annotations.init": "FALSE",
"io.podman.annotations.privileged": "FALSE", "io.podman.annotations.publish-
all": "FALSE", "org.opencontainers.image.stopSignal": "3"}, "AttachStderr":
false, "AttachStdin": false, "AttachStdout": false, "Cmd":
["nginx", "-g", "daemon off;"], "CreateCommand": ["podman", "container",
"run", "--name", "webserver", "--authfile",
"/home/ubuntu/.config/containers/auth.json", "--volume",
"/data:/usr/share/nginx/html", "--publish", "8080:80", "--detach=True",
"acrunir.azurecr.io/nginx
:casopractico2"], "Domainname": "", "Entrypoint": "/docker-entrypoint.sh",
"Env": ["PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin",
"TERM=xterm", "container=podman", "NGINX_VERSION=1.27.0",
"NJS_VERSION=0.8.4", "NJS_RELEASE=2~bookworm", "PKG_RELEASE=2~
bookworm", "HOME=/root", "HOSTNAME=ccb2f7c142bb"], "Hostname":
"ccb2f7c142bb", "Image": "acrunir.azurecr.io/nginx:casopractico2", "Labels":
{"maintainer": "NGINX Docker Maintainers <docker-maint@nginx.com>"},
"OnBuild": null, "OpenStdin": false, "StdinOnce": false, "StopS
ignal": 3, "StopTimeout": 10, "Timeout": 0, "Tty": false, "Umask": "0022",
"User": "", "Volumes": null, "WorkingDir": "/"}, "ConmonPidFile":
"/run/user/1000/containers/overlay-
containers/ccb2f7c142bb8d61fcbd17f957596e4c16e3f550b34cdbf800a7f74ec5e42e9/
userdata/common.pid"
, "Created": "2024-07-12T17:18:45.824634462Z", "Dependencies": [], "Driver":
"overlay", "EffectiveCaps": ["CAP_CHOWN", "CAP_DAC_OVERRIDE", "CAP_FOWNER",
"CAP_FSETID", "CAP_KILL", "CAP_NET_BIND_SERVICE", "CAP_SETFCAP",
"CAP_SETGID", "CAP_SETPCAP", "CAP_SETUID", "CAP_SYS_CH
ROOT"], "ExecIDs": [], "ExitCommand": ["/usr/bin/podman", "--root",
"/home/ubuntu/.local/share/containers/storage", "--runroot",
"/run/user/1000/containers", "--log-level", "warning", "--cgroup-manager",
"systemd", "--tmpdir", "/run/user/1000/libpod/tmp", "--runtime", "cr
un", "--storage-driver", "overlay", "--events-backend", "journald",
"container", "cleanup",
"ccb2f7c142bb8d61fcbd17f957596e4c16e3f550b34cdbf800a7f74ec5e42e9"],
"GraphDriver": {"Data": {"LowerDir":
"/home/ubuntu/.local/share/containers/storage/overlay/0fd1afee374d0bd7ec03
3b214165d97cdd1908444e6fde080f520ebd47e887e1/diff:/home/ubuntu/.local/share/c
ontainers/storage/overlay/682ed5b6b2b0ea32c5d535393db4af1aae4f6e8c6f038d0ce4
f006581456427a/diff:/home/ubuntu/.local/share/containers/storage/overlay/0ad4
```

```
caba2a1b48e17253c6671f4b8b0293234e287df796
3a66e93d30c5791ac1/diff:/home/ubuntu/.local/share/containers/storage/overlay/
803b3e3283a63a2c0c304060b2b6a8cc2b08c69437845945470f991c93balab3/diff:/home/u
buntu/.local/share/containers/storage/overlay/464d81937ca53c1e619b705ac415df
a10749cee5de55eb7c51e47fbab5ae8a21/diff:/h
ome/ubuntu/.local/share/containers/storage/overlay/83b52e5fc8f281259e159bd6b
301a653a07be7452c46ee73fc9c04170464343c/diff:/home/ubuntu/.local/share/contai
ners/storage/overlay/e897c7ac975919a39208ba23c0c1eb09c10d063480808c740e90bb8
balf61b2c/diff:/home/ubuntu/.local/share/co
ntainers/storage/overlay/32148f9f6c5aadfa167ee7b146b9703c59307049d68b090c19d
b019fd15c5406/diff", "MergedDir":
"/home/ubuntu/.local/share/containers/storage/overlay/2dd4a87fd08ad760d4a8ab
5173368671a229224c7d0611ecc9560d24d2373e7a/merged", "UpperDir":
"/home/ubuntu/.local/s
hare/containers/storage/overlay/2dd4a87fd08ad760d4a8ab5173368671a229224c7d06
11ecc9560d24d2373e7a/diff", "WorkDir":
"/home/ubuntu/.local/share/containers/storage/overlay/2dd4a87fd08ad760d4a8ab
5173368671a229224c7d0611ecc9560d24d2373e7a/work"}, "Name": "overlay"},
"HostConfi
g": {"AutoRemove": false, "Binds":
["/data:/usr/share/nginx/html:rw,rprivate,rbind"], "BlkioDeviceReadBps":
null, "BlkioDeviceReadIOps": null, "BlkioDeviceWriteBps": null,
"BlkioDeviceWriteIOps": null, "BlkioWeight": 0, "BlkioWeightDevice": null,
"CapAdd": [], "CapDrop":
["CAP_AUDIT_WRITE", "CAP_MKNOD", "CAP_NET_RAW"], "Cgroup": "", "CgroupConf":
null, "CgroupManager": "systemd", "CgroupMode": "private", "CgroupParent":
"user.slice", "Cgroups": "default", "ConsoleSize": [0, 0],
"ContainerIDFile": "", "CpuCount": 0, "CpuPercent": 0, "CpuPe
riod": 0, "CpuQuota": 0, "CpuRealtimePeriod": 0, "CpuRealtimeRuntime": 0,
"CpuShares": 0, "CpusetCpus": "", "CpusetMems": "", "Devices": [],
"DiskQuota": 0, "Dns": [], "DnsOptions": [], "DnsSearch": [], "ExtraHosts":
[], "GroupAdd": [], "IOMaximumBandwidth": 0, "IOMaximum
IOps": 0, "IpcMode": "private", "Isolation": "", "KernelMemory": 0, "Links":
null, "LogConfig": {"Config": null, "Path": "", "Size": "0B", "Tag": "",
"Type": "journald"}, "Memory": 0, "MemoryReservation": 0, "MemorySwap": 0,
"MemorySwappiness": 0, "NanoCpus": 0, "NetworkM
ode": "slirp4netns", "OomKillDisable": false, "OomScoreAdj": 0, "PidMode":
"private", "PidsLimit": 2048, "PortBindings": {"80/tcp": [{"HostIp": "",
"HostPort": "8080"}]}, "Privileged": false, "PublishAllPorts": false,
"ReadonlyRootfs": false, "RestartPolicy": {"MaximumRet
ryCount": 0, "Name": ""}, "Runtime": "oci", "SecurityOpt": [], "ShmSize":
65536000, "Tmpfs": {}, "UTSMode": "private", "Ulimits": [], "UsersnsMode":
"", "VolumeDriver": "", "VolumesFrom": null}, "HostnamePath":
```

```
"/run/user/1000/containers/overlay-containers/ccb2f7c142bba8d6
1fcbd17f957596e4c16e3f550b34cdbf800a7f74ec5e42e9/userdata/hostname",
"HostsPath": "/run/user/1000/containers/overlay-
containers/ccb2f7c142bba8d61fcbd17f957596e4c16e3f550b34cdbf800a7f74ec5e42e9/
userdata/hosts", "Id":
"ccb2f7c142bba8d61fcbd17f957596e4c16e3f550b34cdbf800a7f7
4ec5e42e9", "Image":
"49cb80b89f4db979c82074ab9b819cfe1073979c0a29a95b76433473916ece83",
"ImageName": "acrunir.azurecr.io/nginx:casopractico2", "IsInfra": false,
"MountLabel": "", "Mounts": [{"Destination": "/usr/share/nginx/html",
"Driver": "", "Mode": "", "Options": ["r
bind"], "Propagation": "rprivate", "RW": true, "Source": "/data", "Type":
"bind"}], "Name": "webserver", "Namespace": "", "NetworkSettings":
{"Bridge": "", "EndpointID": "", "Gateway": "", "GlobalIPv6Address": "",
"GlobalIPv6PrefixLen": 0, "HairpinMode": false, "IPAddress
": "", "IPPrefixLen": 0, "IPv6Gateway": "", "LinkLocalIPv6Address": "",
"LinkLocalIPv6PrefixLen": 0, "MacAddress": "", "Ports": {"80/tcp":
[{"HostIp": "", "HostPort": "8080"}]}, "SandboxID": "", "SandboxKey":
"/run/user/1000/netns/cni-65e16882-f1a4-5729-1912-2656e0c6d95b"
}, "OCIConfigPath": "/home/ubuntu/.local/share/containers/storage/overlay-
containers/ccb2f7c142bba8d61fcbd17f957596e4c16e3f550b34cdbf800a7f74ec5e42e9/
userdata/config.json", "OCIRuntime": "crun", "Path": "/docker-entrypoint.sh",
"PidFile": "/run/user/1000/containers/overla
y-
containers/ccb2f7c142bba8d61fcbd17f957596e4c16e3f550b34cdbf800a7f74ec5e42e9/
userdata/pidfile", "Pod": "", "ProcessLabel": "", "ResolvConfPath":
"/run/user/1000/containers/overlay-
containers/ccb2f7c142bba8d61fcbd17f957596e4c16e3f550b34cdbf800a7f74ec5e42e9/
userdata/resolv
.conf", "RestartCount": 0, "Rootfs": "", "State": {"CgroupPath":
"/user.slice/user-1000.slice/user@1000.service/user.slice/libpod-
ccb2f7c142bba8d61fcbd17f957596e4c16e3f550b34cdbf800a7f74ec5e42e9.scope",
"ConmonPid": 6798, "Dead": false, "Error": "", "ExitCode": 0, "Finish
edAt": "0001-01-01T00:00:00Z", "Healthcheck": {"FailingStreak": 0, "Log":
null, "Status": ""}, "OOMKilled": false, "OciVersion": "1.0.2-dev",
"Paused": false, "Pid": 6801, "Restarting": false, "Running": true,
"StartedAt": "2024-07-12T17:18:46.066232254Z", "Status": "runn
ing"}, "StaticDir": "/home/ubuntu/.local/share/containers/storage/overlay-
containers/ccb2f7c142bba8d61fcbd17f957596e4c16e3f550b34cdbf800a7f74ec5e42e9/
userdata"}, "podman_actions": [], "podman_quadlet": "
[Container]\nContainerName=webserver\nImage=acrunir.azurecr.io/nginx:
casopractico2\nPublishPort=8080:80\nVolume=/data:/usr/share/nginx/html\nPodm
anArgs=--authfile /home/ubuntu/.config/containers/auth.json\n",
```

```
"podman_systemd": {"container-webserver": "# container-webserver.service\n# autogenerated by Podman 3.4.4\n# Fri Jul 12 17:49:55 UTC 2024\n\n[Unit]\nDescription=Podman container-webserver.service\nDocumentation=man:podman-generate-systemd(1)\nWants=network-online.target\nAfter=network-online.target\nRequiresMountsFor=/run/user/1000/containers\n\n[Service]\nEnvironment=PODMAN_SYSTEMD_UNIT=%n\nRestart=on-failure\nTimeoutStopSec=70\nExecStart=/usr/bin/podman start webserver\nExecStop=/usr/bin/podman stop -t 10 webserver\nExecStopPost=/usr/bin/podman stop -t 10 webserver\nPIDFile=/run/user/1000/containers/overlay-containers/ccb2f7c142bba8d61fcdbd17f957596e4c16e3f550b34cdbf800a7f74ec5e42e9/userdata/common.pid\nType=forking\n\n[Install]\nWantedBy=default.target\n"}, "stderr": "", "stderr_lines": [], "stdout": "", "stdout_lines": []}
```

```
local_file.remote_inventory (local-exec): TASK [podman : Logout to ACR]
```

```
*****
```

```
local_file.remote_inventory: Still creating... [20s elapsed]
```

```
local_file.remote_inventory (local-exec): changed: [tfnode] => {"changed": true, "stderr": "", "stderr_lines": [], "stdout": "Removed login credentials for all registries\n", "stdout_lines": ["Removed login credentials for all registries"]}
```

```
local_file.remote_inventory (local-exec): TASK [podman : Check that you can connect (GET) to a page and it returns a status 200] ***
```

```
local_file.remote_inventory (local-exec): ok: [tfnode] => {"accept_ranges": "bytes", "changed": false, "connection": "close", "content": "Fri Jul 12 15:42:00 UTC 2024\nFri Jul 12 15:44:09 UTC 2024\nFri Jul 12 17:13:56 UTC 2024\nFri Jul 12 17:18:33 UTC 2024\nFri Jul 12 17:30:26 UTC 2024\nFri Jul 12 17:32:33 UTC 2024\nFri Jul 12 17:35:18 UTC 2024\nFri Jul 12 17:37:14 UTC 2024\nFri Jul 12 17:41:17 UTC 2024\nFri Jul 12 17:46:33 UTC 2024\nFri Jul 12 17:49:52 UTC 2024\n", "content_length": "319", "content_type": "application/octet-stream", "cookies": {}, "cookies_string": "", "date": "Fri, 12 Jul 2024 17:49:58 GMT", "elapsed": 0, "etag": "\"66916cc0-13f\"", "last_modified": "Fri, 12 Jul 2024 17:49:52 GMT", "msg": "OK (319 bytes)", "redirected": false, "server": "nginx/1.27.0", "status": 200, "url": "http://40.69.204.208:8080/date"}
```

```
local_file.remote_inventory (local-exec): TASK [podman : Show url content]
```

```
*****
```

```
local_file.remote_inventory (local-exec): ok: [tfnode] => {
```



```

local_file.remote_inventory (local-exec):      "msg": "URL:
\"http://40.69.204.208:8080/date\"\\nStatus: 200\\nContent: Fri Jul 12
15:42:00 UTC 2024\\nFri Jul 12 15:44:09 UTC 2024\\nFri Jul 12 17:13:56 UTC
2024\\nFri Jul 12 17:18:33 UTC 2024\\nFri Jul 12 17:30:26 UTC 2024\\nFri J
ul 12 17:32:33 UTC 2024\\nFri Jul 12 17:35:18 UTC 2024\\nFri Jul 12 17:37:14
UTC 2024\\nFri Jul 12 17:41:17 UTC 2024\\nFri Jul 12 17:46:33 UTC 2024\\nFri
Jul 12 17:49:52 UTC 2024\\n"
local_file.remote_inventory (local-exec): }

local_file.remote_inventory (local-exec): PLAY RECAP
*****
local_file.remote_inventory (local-exec): tfnode                                : ok=12
changed=5      unreachable=0      failed=0      skipped=0      rescued=0
ignored=0

local_file.remote_inventory: Creation complete after 22s
[id=e4bbbeb1c88df65ca55f1e72dc3481dd1e385fa28]

Apply complete! Resources: 3 added, 0 changed, 0 destroyed.

```

La siguiente imagen muestra el acceso a nuestro servidor web



```

[dani@draco ~/projects/anjana/repos/ansible-inventories ][develop ?] $ curl http://13.74.176.255:8080/date
Fri Jul 12 13:03:25 UTC 2024
Fri Jul 12 13:04:07 UTC 2024
Fri Jul 12 13:06:12 UTC 2024
Fri Jul 12 13:07:31 UTC 2024
Fri Jul 12 13:17:04 UTC 2024
Fri Jul 12 13:32:12 UTC 2024
Fri Jul 12 13:42:22 UTC 2024
Fri Jul 12 13:43:42 UTC 2024
Fri Jul 12 13:44:21 UTC 2024
Fri Jul 12 13:45:30 UTC 2024
Fri Jul 12 13:46:05 UTC 2024
Fri Jul 12 13:48:11 UTC 2024
Fri Jul 12 14:10:28 UTC 2024
Fri Jul 12 14:11:55 UTC 2024
Fri Jul 12 14:13:39 UTC 2024
Fri Jul 12 14:16:24 UTC 2024
Fri Jul 12 14:29:39 UTC 2024
Fri Jul 12 14:44:43 UTC 2024

```

## Configuración de AKS utilizando Terraform y Ansible

Para esta practica utilizaremos el enfoque que venimos usando. *Terraform* crea la infraestructura y *Ansible* la configura. Ciertamente es que para *K8S* este enfoque no parece el adecuado. Quizás usando *HELM* y alguna herramienta de CI/CD específica como *ARGOCD* o *Flux CD* sería una solución más robusta y menos artesanal.

Justo después de crear nuestro cluster de *K8S*. De esta forma podemos sin intervención humana desplegar las aplicaciones que nos interese

```

resource "azurerm_kubernetes_cluster" "aks" {
  for_each = { for k, v in var.aks : k => v if v.enabled == true } #only

```

*enabled ones*

```
name                = "${each.key}unir"
location            = each.value.location
resource_group_name = azurerm_resource_group.rgs[each.value.rg].name
dns_prefix          = each.value.dns_prefix

default_node_pool {
  name          = "default"
  node_count    = 1
  vm_size       = "Standard_D2s_v3"
  tags          = merge(each.value.tags, var.provider_default_tags)
  upgrade_settings {
    drain_timeout_in_minutes = 0
    max_surge                = "10%"
    node_soak_duration_in_minutes = 0
  }
}

identity {
  type = "SystemAssigned"
}

linux_profile {
  admin_username = each.value.lp.user
  ssh_key {
    key_data = each.value.lp.ssh_key
  }
}

key_vault_secrets_provider {
  secret_rotation_enabled = true
}

tags = merge(each.value.tags, var.provider_default_tags)
}
```

En el instalador creamos un recurso que genera la configuración de acceso a nuestro cluster

```
resource "local_file" "k8s_config" {
  content = azurerm_kubernetes_cluster.aks["aks"].kube_config_raw
  filename = "../ansible/roles/aks/files/config.yml"

  provisioner "local-exec" {
```

```

    working_dir = "../ansible"
    command      = "ansible-playbook -v -i inventories/local/hosts.yml -u
dani aks.yml"
  }
  depends_on = [
    azurerm_kubernetes_cluster.aks["aks"]
  ]
}

```

Y ejecuta nuestro rol, la salida se puede ver a continuación.

```

local_file.k8s_config: Destroying...
[id=eed0e4f705bb1088635b29adaecb21b00b3a4c08]
local_file.k8s_config: Destruction complete after 0s
local_file.k8s_config: Creating...
local_file.k8s_config: Provisioning with 'local-exec'...
local_file.k8s_config (local-exec): Executing: ["/bin/sh" "-c" "ansible-
playbook -v -i inventories/local/hosts.yml -u dani aks.yml"]
local_file.k8s_config (local-exec): No config file found; using defaults

local_file.k8s_config (local-exec): PLAY [localhost]
*****

local_file.k8s_config (local-exec): TASK [Gathering Facts]
*****

local_file.k8s_config (local-exec): [WARNING]: Platform linux on host
localhost is using the discovered Python
local_file.k8s_config (local-exec): interpreter at /usr/bin/python3.12, but
future installation of another Python
local_file.k8s_config (local-exec): interpreter could change the meaning of
that path. See
local_file.k8s_config (local-exec): https://docs.ansible.com/ansible-
local_file.k8s_config (local-exec):
core/2.17/reference_appendices/interpreter_discovery.html for more
information.
local_file.k8s_config (local-exec): ok: [localhost]

local_file.k8s_config (local-exec): TASK [aks : Get Cluster information]
*****

local_file.k8s_config (local-exec): ok: [localhost] => {"apis":
{"admissionregistration.k8s.io/v1": {"MutatingWebhookConfiguration":
{"categories": ["api-extensions"], "name": "mutatingwebhookconfigurations",
"namespaced": false, "preferred": true, "short_names": [], "singular_name":
"mutatingwebhookconfiguration"}, "ValidatingWebhookConfiguration":

```



```
{
  "categories": ["api-extensions"],
  "name": "validatingwebhookconfigurations",
  "namespaced": false,
  "preferred": true,
  "short_names": [],
  "singular_name": "validatingwebhookconfiguration"
},
{
  "categories": ["api-extensions"],
  "name": "validatingadmissionpolicies",
  "namespaced": false,
  "preferred": false,
  "short_names": [],
  "singular_name": "validatingadmissionpolicy"
},
{
  "categories": ["api-extensions"],
  "name": "validatingadmissionpolicybindings",
  "namespaced": false,
  "preferred": false,
  "short_names": [],
  "singular_name": "validatingadmissionpolicybinding"
},
{
  "categories": ["api-extensions"],
  "name": "customresourcedefinitions",
  "namespaced": false,
  "preferred": true,
  "short_names": ["crd", "crds"],
  "singular_name": "customresourcedefinition"
},
{
  "categories": ["api-extensions"],
  "name": "apiservices",
  "namespaced": false,
  "preferred": true,
  "short_names": [],
  "singular_name": "apiservice"
},
{
  "categories": [],
  "name": "controllerrevisions",
  "namespaced": true,
  "preferred": true,
  "short_names": [],
  "singular_name": "controllerrevision"
},
{
  "categories": ["all"],
  "name": "daemonsets",
  "namespaced": true,
  "preferred": true,
  "short_names": ["ds"],
  "singular_name": "daemonset"
},
{
  "categories": ["all"],
  "name": "deployments",
  "namespaced": true,
  "preferred": true,
  "short_names": ["deploy"],
  "singular_name": "deployment"
},
{
  "categories": ["all"],
  "name": "replicasets",
  "namespaced": true,
  "preferred": true,
  "short_names": ["rs"],
  "singular_name": "replicaset"
},
{
  "categories": ["all"],
  "name": "statefulsets",
  "namespaced": true,
  "preferred": true,
  "short_names": ["sts"],
  "singular_name": "statefulset"
},
{
  "categories": [],
  "name": "appprojects",
  "namespaced": true,
  "preferred": true,
  "short_names": ["appproj", "appprojs"],
  "singular_name": "appproject"
},
{
  "categories": [],
  "name": "applications",
  "namespaced": true,
  "preferred": true,
  "short_names": ["app", "apps"],
  "singular_name": "application"
},
{
  "categories": [],
  "name": "applicationsets",
  "namespaced": true,
  "preferred": true,
  "short_names": ["appset", "appsets"],
  "singular_name": "applicationset"
},
{
  "categories": [],
  "name": "selfsubjectreviews",
  "namespaced": false,
  "preferred": true,
  "short_names": [],
  "singular_name": "selfsubjectreview"
},
{
  "categories": [],
  "name": "tokenreviews",
  "namespaced": false,
  "preferred": true,
  "short_names": [],
  "singular_name": "tokenreview"
},
{
  "categories": [],
  "name": "selfsubjectreviews",
  "namespaced": false,
  "preferred": false,
  "short_names": [],
  "singular_name": "selfsubjectreview"
}
```

```
"authorization.k8s.io/v1": {"LocalSubjectAccessReview": {"categories": [],
"name": "localsubjectaccessreviews", "namespaced": true, "preferred": true,
"short_names": [], "singular_name": "localsubjectaccessreview"},
"SelfSubjectAccessReview": {"categories": [], "name":
"selfsubjectaccessreviews", "namespaced": false, "preferred": true,
"short_names": [], "singular_name": "selfsubjectaccessreview"},
"SelfSubjectRulesReview": {"categories": [], "name":
"selfsubjectrulesreviews", "namespaced": false, "preferred": true,
"short_names": [], "singular_name": "selfsubjectrulesreview"},
"SubjectAccessReview": {"categories": [], "name": "subjectaccessreviews",
"namespaced": false, "preferred": true, "short_names": [], "singular_name":
"subjectaccessreview"}}, "autoscaling/v1": {"HorizontalPodAutoscaler":
{"categories": ["all"], "name": "horizontalpodautoscalers", "namespaced":
true, "preferred": false, "short_names": ["hpa"], "singular_name":
"horizontalpodautoscaler"}}, "autoscaling/v2": {"HorizontalPodAutoscaler":
{"categories": ["all"], "name": "horizontalpodautoscalers", "namespaced":
true, "preferred": true, "short_names": ["hpa"], "singular_name":
"horizontalpodautoscaler"}}, "batch/v1": {"CronJob": {"categories": ["all"],
"name": "cronjobs", "namespaced": true, "preferred": true, "short_names":
["cj"], "singular_name": "cronjob"}, "Job": {"categories": ["all"], "name":
"jobs", "namespaced": true, "preferred": true, "short_names": [],
"singular_name": "job"}}, "certificates.k8s.io/v1":
{"CertificateSigningRequest": {"categories": [], "name":
"certificatesigningrequests", "namespaced": false, "preferred": true,
"short_names": ["csr"], "singular_name": "certificatesigningrequest"}},
"coordination.k8s.io/v1": {"Lease": {"categories": [], "name": "leases",
"namespaced": true, "preferred": true, "short_names": [], "singular_name":
"lease"}}, "discovery.k8s.io/v1": {"EndpointSlice": {"categories": [],
"name": "endpointslices", "namespaced": true, "preferred": true,
"short_names": [], "singular_name": "endpointslice"}}, "events.k8s.io/v1":
{"Event": {"categories": [], "name": "events", "namespaced": true,
"preferred": true, "short_names": ["ev"], "singular_name": "event"}},
"flowcontrol.apiserver.k8s.io/v1beta2": {"FlowSchema": {"categories": [],
"name": "flowschemas", "namespaced": false, "preferred": false,
"short_names": [], "singular_name": "flowschema"},
"PriorityLevelConfiguration": {"categories": [], "name":
"prioritylevelconfigurations", "namespaced": false, "preferred": false,
"short_names": [], "singular_name": "prioritylevelconfiguration"}},
"flowcontrol.apiserver.k8s.io/v1beta3": {"FlowSchema": {"categories": [],
"name": "flowschemas", "namespaced": false, "preferred": true, "short_names":
[], "singular_name": "flowschema"}, "PriorityLevelConfiguration":
{"categories": [], "name": "prioritylevelconfigurations", "namespaced":
false, "preferred": true, "short_names": [], "singular_name":
```

```
"prioritylevelconfiguration"}}}, "networking.k8s.io/v1": {"Ingress": {"categories": [], "name": "ingresses", "namespaced": true, "preferred": true, "short_names": ["ing"], "singular_name": "ingress"}, "IngressClass": {"categories": [], "name": "ingressclasses", "namespaced": false, "preferred": true, "short_names": [], "singular_name": "ingressclass"}, "NetworkPolicy": {"categories": [], "name": "networkpolicies", "namespaced": true, "preferred": true, "short_names": ["netpol"], "singular_name": "networkpolicy"}}, "node.k8s.io/v1": {"RuntimeClass": {"categories": [], "name": "runtimeclasses", "namespaced": false, "preferred": true, "short_names": [], "singular_name": "runtimeclass"}}, "policy/v1": {"PodDisruptionBudget": {"categories": [], "name": "poddisruptionbudgets", "namespaced": true, "preferred": true, "short_names": ["pdb"], "singular_name": "poddisruptionbudget"}}, "rbac.authorization.k8s.io/v1": {"ClusterRole": {"categories": [], "name": "clusterroles", "namespaced": false, "preferred": true, "short_names": [], "singular_name": "clusterrole"}, "ClusterRoleBinding": {"categories": [], "name": "clusterrolebindings", "namespaced": false, "preferred": true, "short_names": [], "singular_name": "clusterrolebinding"}, "Role": {"categories": [], "name": "roles", "namespaced": true, "preferred": true, "short_names": [], "singular_name": "role"}, "RoleBinding": {"categories": [], "name": "rolebindings", "namespaced": true, "preferred": true, "short_names": [], "singular_name": "rolebinding"}}, "scheduling.k8s.io/v1": {"PriorityClass": {"categories": [], "name": "priorityclasses", "namespaced": false, "preferred": true, "short_names": ["pc"], "singular_name": "priorityclass"}}, "secrets-store.csi.x-k8s.io/v1": {"SecretProviderClass": {"categories": [], "name": "secretproviderclasses", "namespaced": true, "preferred": true, "short_names": [], "singular_name": "secretproviderclass"}, "SecretProviderClassPodStatus": {"categories": [], "name": "secretproviderclasspodstatuses", "namespaced": true, "preferred": true, "short_names": [], "singular_name": "secretproviderclasspodstatus"}}, "secrets-store.csi.x-k8s.io/v1alpha1": {"SecretProviderClass": {"categories": [], "name": "secretproviderclasses", "namespaced": true, "preferred": false, "short_names": [], "singular_name": "secretproviderclass"}, "SecretProviderClassPodStatus": {"categories": [], "name": "secretproviderclasspodstatuses", "namespaced": true, "preferred": false, "short_names": [], "singular_name": "secretproviderclasspodstatus"}}, "snapshot.storage.k8s.io/v1": {"VolumeSnapshot": {"categories": [], "name": "volumesnapshots", "namespaced": true, "preferred": true, "short_names": ["vs"], "singular_name": "volumesnapshot"}, "VolumeSnapshotClass": {"categories": [], "name": "volumesnapshotclasses", "namespaced": false, "preferred": true, "short_names": ["vsclass", "vsclasses"], "singular_name": "volumesnapshotclass"}, "VolumeSnapshotContent": {"categories": [], "name": "volumesnapshotcontents", "namespaced": false, "preferred": true,
```

```
"short_names": ["vsc", "vscs"], "singular_name": "volumesnapshotcontent"}},
"storage.k8s.io/v1": {"CSIDriver": {"categories": [], "name": "csidrivers",
"namespaced": false, "preferred": true, "short_names": [], "singular_name":
"csidriver"}, "CSINode": {"categories": [], "name": "csinodes",
"namespaced": false, "preferred": true, "short_names": [], "singular_name":
"csinode"}, "CSIStorageCapacity": {"categories": [], "name":
"csistoragecapacities", "namespaced": true, "preferred": true,
"short_names": [], "singular_name": "csistoragecapacity"}, "StorageClass":
{"categories": [], "name": "storageclasses", "namespaced": false,
"preferred": true, "short_names": ["sc"], "singular_name": "storageclass"},
"VolumeAttachment": {"categories": [], "name": "volumeattachments",
"namespaced": false, "preferred": true, "short_names": [], "singular_name":
"volumeattachment"}}, "v1": {"Binding": {"categories": [], "name":
"bindings", "namespaced": true, "preferred": true, "short_names": [],
"singular_name": "binding"}, "ComponentStatus": {"categories": [], "name":
"componentstatuses", "namespaced": false, "preferred": true, "short_names":
["cs"], "singular_name": "componentstatus"}, "ConfigMap": {"categories": [],
"name": "configmaps", "namespaced": true, "preferred": true, "short_names":
["cm"], "singular_name": "configmap"}, "Endpoints": {"categories": [],
"name": "endpoints", "namespaced": true, "preferred": true, "short_names":
["ep"], "singular_name": "endpoints"}, "Event": {"categories": [], "name":
"events", "namespaced": true, "preferred": true, "short_names": ["ev"],
"singular_name": "event"}, "LimitRange": {"categories": [], "name":
"limitranges", "namespaced": true, "preferred": true, "short_names":
["limits"], "singular_name": "limitrange"}, "List": {"categories": [],
"name": null, "namespaced": null, "preferred": null, "short_names": [],
"singular_name": null}, "Namespace": {"categories": [], "name":
"namespaces", "namespaced": false, "preferred": true, "short_names": ["ns"],
"singular_name": "namespace"}, "Node": {"categories": [], "name": "nodes",
"namespaced": false, "preferred": true, "short_names": ["no"],
"singular_name": "node"}, "PersistentVolume": {"categories": [], "name":
"persistentvolumes", "namespaced": false, "preferred": true, "short_names":
["pv"], "singular_name": "persistentvolume"}, "PersistentVolumeClaim":
{"categories": [], "name": "persistentvolumeclaims", "namespaced": true,
"preferred": true, "short_names": ["pvc"], "singular_name":
"persistentvolumeclaim"}, "Pod": {"categories": ["all"], "name": "pods",
"namespaced": true, "preferred": true, "short_names": ["po"],
"singular_name": "pod"}, "PodTemplate": {"categories": [], "name":
"podtemplates", "namespaced": true, "preferred": true, "short_names": [],
"singular_name": "podtemplate"}, "ReplicationController": {"categories":
["all"], "name": "replicationcontrollers", "namespaced": true, "preferred":
true, "short_names": ["rc"], "singular_name": "replicationcontroller"},
"ResourceQuota": {"categories": [], "name": "resourcequotas", "namespaced":
```

```
true, "preferred": true, "short_names": ["quota"], "singular_name":
"resourcequota"}, "Secret": {"categories": [], "name": "secrets",
"namespaced": true, "preferred": true, "short_names": [], "singular_name":
"secret"}, "Service": {"categories": ["all"], "name": "services",
"namespaced": true, "preferred": true, "short_names": ["svc"],
"singular_name": "service"}, "ServiceAccount": {"categories": [], "name":
"serviceaccounts", "namespaced": true, "preferred": true, "short_names":
["sa"], "singular_name": "serviceaccount"}}}, "changed": false,
"connection": {"cert_file": "/tmp/tmp9eaz8jc5", "host": "https://unir-
ik5kq7b4.hcp.northeurope.azmk8s.io:443", "password": null, "proxy": null,
"ssl_ca_cert": "/tmp/tmpo63m3of2", "username": null, "verify_ssl": true},
"version": {"client": "28.1.0", "server": {"kubernetes": {"buildDate":
"2024-05-22T15:00:24Z", "compiler": "gc", "gitCommit":
"21be1d76a90bc00e2b0f6676a664bdf097224155", "gitTreeState": "clean",
"gitVersion": "v1.28.10", "goVersion": "go1.21.9", "major": "1", "minor":
"28", "platform": "linux/amd64"}}}}
```

```
local_file.k8s_config (local-exec): TASK [aks : Show Cluster info]
*****
local_file.k8s_config (local-exec): ok: [localhost] => {
local_file.k8s_config (local-exec):      "msg": {
local_file.k8s_config (local-exec):          "apis": {
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):      "admissionregistration.k8s.io/v1": {
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):      "MutatingWebhookConfiguration": {
local_file.k8s_config (local-exec):          "categories": [
local_file.k8s_config (local-exec):              "api-extensions"
local_file.k8s_config (local-exec):          ],
local_file.k8s_config (local-exec):          "name":
local_file.k8s_config (local-exec):      "mutatingwebhookconfigurations",
local_file.k8s_config (local-exec):          "namespaced": false,
local_file.k8s_config (local-exec):          "preferred": true,
local_file.k8s_config (local-exec):          "short_names": [],
local_file.k8s_config (local-exec):          "singular_name":
local_file.k8s_config (local-exec):      "mutatingwebhookconfiguration"
local_file.k8s_config (local-exec):      },
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):      "ValidatingWebhookConfiguration": {
local_file.k8s_config (local-exec):          "categories": [
local_file.k8s_config (local-exec):              "api-extensions"
local_file.k8s_config (local-exec):          ],
local_file.k8s_config (local-exec):          "name":
```



```

"validatingwebhookconfigurations",
local_file.k8s_config (local-exec):      "namespaced": false,
local_file.k8s_config (local-exec):      "preferred": true,
local_file.k8s_config (local-exec):      "short_names": [],
local_file.k8s_config (local-exec):      "singular_name":
"validatingwebhookconfiguration"
local_file.k8s_config (local-exec):      }
local_file.k8s_config (local-exec):      },
local_file.k8s_config (local-exec):
"admissionregistration.k8s.io/v1beta1": {
local_file.k8s_config (local-exec):
"ValidatingAdmissionPolicy": {
local_file.k8s_config (local-exec):      "categories": [
local_file.k8s_config (local-exec):          "api-extensions"
local_file.k8s_config (local-exec):      ],
local_file.k8s_config (local-exec):      "name":
"validatingadmissionpolicies",
local_file.k8s_config (local-exec):      "namespaced": false,
local_file.k8s_config (local-exec):      "preferred": false,
local_file.k8s_config (local-exec):      "short_names": [],
local_file.k8s_config (local-exec):      "singular_name":
"validatingadmissionpolicy"
local_file.k8s_config (local-exec):      },
local_file.k8s_config (local-exec):
"ValidatingAdmissionPolicyBinding": {
local_file.k8s_config (local-exec):      "categories": [
local_file.k8s_config (local-exec):          "api-extensions"
local_file.k8s_config (local-exec):      ],
local_file.k8s_config (local-exec):      "name":
"validatingadmissionpolicybindings",
local_file.k8s_config (local-exec):      "namespaced": false,
local_file.k8s_config (local-exec):      "preferred": false,
local_file.k8s_config (local-exec):      "short_names": [],
local_file.k8s_config (local-exec):      "singular_name":
"validatingadmissionpolicybinding"
local_file.k8s_config (local-exec):      }
local_file.k8s_config (local-exec):      },
local_file.k8s_config (local-exec):      "apiextensions.k8s.io/v1": {
local_file.k8s_config (local-exec):          "CustomResourceDefinition":
{
local_file.k8s_config (local-exec):      "categories": [
local_file.k8s_config (local-exec):          "api-extensions"
local_file.k8s_config (local-exec):      ],

```

```

local_file.k8s_config (local-exec):           "name":
"customresourcedefinitions",
local_file.k8s_config (local-exec):           "namespaced": false,
local_file.k8s_config (local-exec):           "preferred": true,
local_file.k8s_config (local-exec):           "short_names": [
local_file.k8s_config (local-exec):             "crd",
local_file.k8s_config (local-exec):             "crds"
local_file.k8s_config (local-exec):         ],
local_file.k8s_config (local-exec):           "singular_name":
"customresourcedefinition"
local_file.k8s_config (local-exec):         }
local_file.k8s_config (local-exec):     },
local_file.k8s_config (local-exec):     "apiregistration.k8s.io/v1": {
local_file.k8s_config (local-exec):         "APIService": {
local_file.k8s_config (local-exec):             "categories": [
local_file.k8s_config (local-exec):                 "api-extensions"
local_file.k8s_config (local-exec):             ],
local_file.k8s_config (local-exec):             "name": "apiservices",
local_file.k8s_config (local-exec):             "namespaced": false,
local_file.k8s_config (local-exec):             "preferred": true,
local_file.k8s_config (local-exec):             "short_names": [],
local_file.k8s_config (local-exec):             "singular_name":
"apiservice"
local_file.k8s_config (local-exec):         }
local_file.k8s_config (local-exec):     },
local_file.k8s_config (local-exec):     "apps/v1": {
local_file.k8s_config (local-exec):         "ControllerRevision": {
local_file.k8s_config (local-exec):             "categories": [],
local_file.k8s_config (local-exec):             "name":
"controllerrevisions",
local_file.k8s_config (local-exec):             "namespaced": true,
local_file.k8s_config (local-exec):             "preferred": true,
local_file.k8s_config (local-exec):             "short_names": [],
local_file.k8s_config (local-exec):             "singular_name":
"controllerrevision"
local_file.k8s_config (local-exec):         },
local_file.k8s_config (local-exec):         "DaemonSet": {
local_file.k8s_config (local-exec):             "categories": [
local_file.k8s_config (local-exec):                 "all"
local_file.k8s_config (local-exec):             ],
local_file.k8s_config (local-exec):             "name": "daemonsets",
local_file.k8s_config (local-exec):             "namespaced": true,
local_file.k8s_config (local-exec):             "preferred": true,

```

```

local_file.k8s_config (local-exec):      "short_names": [
local_file.k8s_config (local-exec):      "ds"
local_file.k8s_config (local-exec):      ],
local_file.k8s_config (local-exec):      "singular_name":
"daemonset"
local_file.k8s_config (local-exec):      },
local_file.k8s_config (local-exec):      "Deployment": {
local_file.k8s_config (local-exec):          "categories": [
local_file.k8s_config (local-exec):              "all"
local_file.k8s_config (local-exec):          ],
local_file.k8s_config (local-exec):          "name": "deployments",
local_file.k8s_config (local-exec):          "namespaced": true,
local_file.k8s_config (local-exec):          "preferred": true,
local_file.k8s_config (local-exec):          "short_names": [
local_file.k8s_config (local-exec):              "deploy"
local_file.k8s_config (local-exec):          ],
local_file.k8s_config (local-exec):          "singular_name":
"deployment"
local_file.k8s_config (local-exec):      },
local_file.k8s_config (local-exec):      "ReplicaSet": {
local_file.k8s_config (local-exec):          "categories": [
local_file.k8s_config (local-exec):              "all"
local_file.k8s_config (local-exec):          ],
local_file.k8s_config (local-exec):          "name": "replicasets",
local_file.k8s_config (local-exec):          "namespaced": true,
local_file.k8s_config (local-exec):          "preferred": true,
local_file.k8s_config (local-exec):          "short_names": [
local_file.k8s_config (local-exec):              "rs"
local_file.k8s_config (local-exec):          ],
local_file.k8s_config (local-exec):          "singular_name":
"replicaset"
local_file.k8s_config (local-exec):      },
local_file.k8s_config (local-exec):      "StatefulSet": {
local_file.k8s_config (local-exec):          "categories": [
local_file.k8s_config (local-exec):              "all"
local_file.k8s_config (local-exec):          ],
local_file.k8s_config (local-exec):          "name":
"statefulsets",
local_file.k8s_config (local-exec):          "namespaced": true,
local_file.k8s_config (local-exec):          "preferred": true,
local_file.k8s_config (local-exec):          "short_names": [
local_file.k8s_config (local-exec):              "sts"
local_file.k8s_config (local-exec):          ],

```



```

local_file.k8s_config (local-exec):          "singular_name":
"statefulset"
local_file.k8s_config (local-exec):          }
local_file.k8s_config (local-exec):    },
local_file.k8s_config (local-exec):    "argoproj.io/v1alpha1": {
local_file.k8s_config (local-exec):      "AppProject": {
local_file.k8s_config (local-exec):        "categories": [],
local_file.k8s_config (local-exec):        "name": "appprojects",
local_file.k8s_config (local-exec):        "namespaced": true,
local_file.k8s_config (local-exec):        "preferred": true,
local_file.k8s_config (local-exec):        "short_names": [
local_file.k8s_config (local-exec):          "appproj",
local_file.k8s_config (local-exec):          "approjs"
local_file.k8s_config (local-exec):        ],
local_file.k8s_config (local-exec):        "singular_name":
"appproject"
local_file.k8s_config (local-exec):    },
local_file.k8s_config (local-exec):    "Application": {
local_file.k8s_config (local-exec):      "categories": [],
local_file.k8s_config (local-exec):      "name":
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):      "namespaced": true,
local_file.k8s_config (local-exec):      "preferred": true,
local_file.k8s_config (local-exec):      "short_names": [
local_file.k8s_config (local-exec):        "app",
local_file.k8s_config (local-exec):        "apps"
local_file.k8s_config (local-exec):      ],
local_file.k8s_config (local-exec):      "singular_name":
"application"
local_file.k8s_config (local-exec):    },
local_file.k8s_config (local-exec):    "ApplicationSet": {
local_file.k8s_config (local-exec):      "categories": [],
local_file.k8s_config (local-exec):      "name":
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):      "namespaced": true,
local_file.k8s_config (local-exec):      "preferred": true,
local_file.k8s_config (local-exec):      "short_names": [
local_file.k8s_config (local-exec):        "appset",
local_file.k8s_config (local-exec):        "appsets"
local_file.k8s_config (local-exec):      ],
local_file.k8s_config (local-exec):      "singular_name":
"applicationset"
local_file.k8s_config (local-exec):    }

```

```

local_file.k8s_config (local-exec): },
local_file.k8s_config (local-exec): "authentication.k8s.io/v1": {
local_file.k8s_config (local-exec):     "SelfSubjectReview": {
local_file.k8s_config (local-exec):         "categories": [],
local_file.k8s_config (local-exec):         "name":
"selfsubjectreviews",
local_file.k8s_config (local-exec):         "namespaced": false,
local_file.k8s_config (local-exec):         "preferred": true,
local_file.k8s_config (local-exec):         "short_names": [],
local_file.k8s_config (local-exec):         "singular_name":
"selfsubjectreview"
local_file.k8s_config (local-exec):     },
local_file.k8s_config (local-exec):     "TokenReview": {
local_file.k8s_config (local-exec):         "categories": [],
local_file.k8s_config (local-exec):         "name":
"tokenreviews",
local_file.k8s_config (local-exec):         "namespaced": false,
local_file.k8s_config (local-exec):         "preferred": true,
local_file.k8s_config (local-exec):         "short_names": [],
local_file.k8s_config (local-exec):         "singular_name":
"tokenreview"
local_file.k8s_config (local-exec):     }
local_file.k8s_config (local-exec): },
local_file.k8s_config (local-exec): "authentication.k8s.io/v1beta1": {
local_file.k8s_config (local-exec):     "SelfSubjectReview": {
local_file.k8s_config (local-exec):         "categories": [],
local_file.k8s_config (local-exec):         "name":
"selfsubjectreviews",
local_file.k8s_config (local-exec):         "namespaced": false,
local_file.k8s_config (local-exec):         "preferred": false,
local_file.k8s_config (local-exec):         "short_names": [],
local_file.k8s_config (local-exec):         "singular_name":
"selfsubjectreview"
local_file.k8s_config (local-exec):     }
local_file.k8s_config (local-exec): },
local_file.k8s_config (local-exec): "authorization.k8s.io/v1": {
local_file.k8s_config (local-exec):     "LocalSubjectAccessReview": {
local_file.k8s_config (local-exec):         "categories": [],
local_file.k8s_config (local-exec):         "name":
"localsubjectaccessreviews",
local_file.k8s_config (local-exec):         "namespaced": true,

```

```

local_file.k8s_config (local-exec):           "preferred": true,
local_file.k8s_config (local-exec):           "short_names": [],
local_file.k8s_config (local-exec):           "singular_name":
"localsubjectaccessreview"
local_file.k8s_config (local-exec):           },
local_file.k8s_config (local-exec):           "SelfSubjectAccessReview":
{
local_file.k8s_config (local-exec):           "categories": [],
local_file.k8s_config (local-exec):           "name":
"selfsubjectaccessreviews",
local_file.k8s_config (local-exec):           "namespaced": false,
local_file.k8s_config (local-exec):           "preferred": true,
local_file.k8s_config (local-exec):           "short_names": [],
local_file.k8s_config (local-exec):           "singular_name":
"selfsubjectaccessreview"
local_file.k8s_config (local-exec):           },
local_file.k8s_config (local-exec):           "SelfSubjectRulesReview":
{
local_file.k8s_config (local-exec):           "categories": [],
local_file.k8s_config (local-exec):           "name":
"selfsubjectrulesreviews",
local_file.k8s_config (local-exec):           "namespaced": false,
local_file.k8s_config (local-exec):           "preferred": true,
local_file.k8s_config (local-exec):           "short_names": [],
local_file.k8s_config (local-exec):           "singular_name":
"selfsubjectrulesreview"
local_file.k8s_config (local-exec):           },
local_file.k8s_config (local-exec):           "SubjectAccessReview": {
local_file.k8s_config (local-exec):           "categories": [],
local_file.k8s_config (local-exec):           "name":
"subjectaccessreviews",
local_file.k8s_config (local-exec):           "namespaced": false,
local_file.k8s_config (local-exec):           "preferred": true,
local_file.k8s_config (local-exec):           "short_names": [],
local_file.k8s_config (local-exec):           "singular_name":
"subjectaccessreview"
local_file.k8s_config (local-exec):           }
local_file.k8s_config (local-exec):           },
local_file.k8s_config (local-exec):           "autoscaling/v1": {
local_file.k8s_config (local-exec):           "HorizontalPodAutoscaler":
{
local_file.k8s_config (local-exec):           "categories": [
local_file.k8s_config (local-exec):           "all"

```

```

local_file.k8s_config (local-exec): ],
local_file.k8s_config (local-exec): "name":
"horizontalpodautoscalers",
local_file.k8s_config (local-exec): "namespaced": true,
local_file.k8s_config (local-exec): "preferred": false,
local_file.k8s_config (local-exec): "short_names": [
local_file.k8s_config (local-exec): "hpa"
local_file.k8s_config (local-exec): ],
local_file.k8s_config (local-exec): "singular_name":
"horizontalpodautoscaler"
local_file.k8s_config (local-exec): }
local_file.k8s_config (local-exec): },
local_file.k8s_config (local-exec): "autoscaling/v2": {
local_file.k8s_config (local-exec): "HorizontalPodAutoscaler":
{
local_file.k8s_config (local-exec): "categories": [
local_file.k8s_config (local-exec): "all"
local_file.k8s_config (local-exec): ],
local_file.k8s_config (local-exec): "name":
"horizontalpodautoscalers",
local_file.k8s_config (local-exec): "namespaced": true,
local_file.k8s_config (local-exec): "preferred": true,
local_file.k8s_config (local-exec): "short_names": [
local_file.k8s_config (local-exec): "hpa"
local_file.k8s_config (local-exec): ],
local_file.k8s_config (local-exec): "singular_name":
"horizontalpodautoscaler"
local_file.k8s_config (local-exec): }
local_file.k8s_config (local-exec): },
local_file.k8s_config (local-exec): "batch/v1": {
local_file.k8s_config (local-exec): "CronJob": {
local_file.k8s_config (local-exec): "categories": [
local_file.k8s_config (local-exec): "all"
local_file.k8s_config (local-exec): ],
local_file.k8s_config (local-exec): "name": "cronjobs",
local_file.k8s_config (local-exec): "namespaced": true,
local_file.k8s_config (local-exec): "preferred": true,
local_file.k8s_config (local-exec): "short_names": [
local_file.k8s_config (local-exec): "cj"
local_file.k8s_config (local-exec): ],
local_file.k8s_config (local-exec): "singular_name":
"cronjob"
local_file.k8s_config (local-exec): },

```

```
local_file.k8s_config (local-exec) :           "Job": {
local_file.k8s_config (local-exec) :             "categories": [
local_file.k8s_config (local-exec) :               "all"
local_file.k8s_config (local-exec) :             ],
local_file.k8s_config (local-exec) :             "name": "jobs",
local_file.k8s_config (local-exec) :             "namespaced": true,
local_file.k8s_config (local-exec) :             "preferred": true,
local_file.k8s_config (local-exec) :             "short_names": [],
local_file.k8s_config (local-exec) :             "singular_name": "job"
local_file.k8s_config (local-exec) :           },
local_file.k8s_config (local-exec) :         "certificates.k8s.io/v1": {
local_file.k8s_config (local-exec) :           "CertificateSigningRequest": {
local_file.k8s_config (local-exec) :             "categories": [],
local_file.k8s_config (local-exec) :             "name":
"certificatesigningrequests",
local_file.k8s_config (local-exec) :             "namespaced": false,
local_file.k8s_config (local-exec) :             "preferred": true,
local_file.k8s_config (local-exec) :             "short_names": [
local_file.k8s_config (local-exec) :               "csr"
local_file.k8s_config (local-exec) :             ],
local_file.k8s_config (local-exec) :             "singular_name":
"certificatesigningrequest"
local_file.k8s_config (local-exec) :           },
local_file.k8s_config (local-exec) :         },
local_file.k8s_config (local-exec) :         "coordination.k8s.io/v1": {
local_file.k8s_config (local-exec) :           "Lease": {
local_file.k8s_config (local-exec) :             "categories": [],
local_file.k8s_config (local-exec) :             "name": "leases",
local_file.k8s_config (local-exec) :             "namespaced": true,
local_file.k8s_config (local-exec) :             "preferred": true,
local_file.k8s_config (local-exec) :             "short_names": [],
local_file.k8s_config (local-exec) :             "singular_name":
"lease"
local_file.k8s_config (local-exec) :           },
local_file.k8s_config (local-exec) :         },
local_file.k8s_config (local-exec) :         "discovery.k8s.io/v1": {
local_file.k8s_config (local-exec) :           "EndpointSlice": {
local_file.k8s_config (local-exec) :             "categories": [],
local_file.k8s_config (local-exec) :             "name":
"endpointslices",
local_file.k8s_config (local-exec) :             "namespaced": true,
```

```

local_file.k8s_config (local-exec):           "preferred": true,
local_file.k8s_config (local-exec):           "short_names": [],
local_file.k8s_config (local-exec):           "singular_name":
"endpointslice"
local_file.k8s_config (local-exec):           }
local_file.k8s_config (local-exec): },
local_file.k8s_config (local-exec): "events.k8s.io/v1": {
local_file.k8s_config (local-exec):     "Event": {
local_file.k8s_config (local-exec):       "categories": [],
local_file.k8s_config (local-exec):       "name": "events",
local_file.k8s_config (local-exec):       "namespaced": true,
local_file.k8s_config (local-exec):       "preferred": true,
local_file.k8s_config (local-exec):       "short_names": [
local_file.k8s_config (local-exec):         "ev"
local_file.k8s_config (local-exec):       ],
local_file.k8s_config (local-exec):       "singular_name":
"event"
local_file.k8s_config (local-exec):     }
local_file.k8s_config (local-exec): },
local_file.k8s_config (local-exec): "flowcontrol.apiserver.k8s.io/v1beta2": {
local_file.k8s_config (local-exec):     "FlowSchema": {
local_file.k8s_config (local-exec):       "categories": [],
local_file.k8s_config (local-exec):       "name": "flowschemas",
local_file.k8s_config (local-exec):       "namespaced": false,
local_file.k8s_config (local-exec):       "preferred": false,
local_file.k8s_config (local-exec):       "short_names": [],
local_file.k8s_config (local-exec):       "singular_name":
"flowschema"
local_file.k8s_config (local-exec):     },
local_file.k8s_config (local-exec):     "PriorityLevelConfiguration": {
local_file.k8s_config (local-exec):       "categories": [],
local_file.k8s_config (local-exec):       "name":
"prioritylevelconfigurations",
local_file.k8s_config (local-exec):       "namespaced": false,
local_file.k8s_config (local-exec):       "preferred": false,
local_file.k8s_config (local-exec):       "short_names": [],
local_file.k8s_config (local-exec):       "singular_name":
"prioritylevelconfiguration"
local_file.k8s_config (local-exec):     }
local_file.k8s_config (local-exec): },
local_file.k8s_config (local-exec):

```

```

"flowcontrol.apiserver.k8s.io/v1beta3": {
  local_file.k8s_config (local-exec):
  local_file.k8s_config (local-exec):
  local_file.k8s_config (local-exec):
  local_file.k8s_config (local-exec):
  local_file.k8s_config (local-exec):
  local_file.k8s_config (local-exec):
  local_file.k8s_config (local-exec):
  "flowschema"
  local_file.k8s_config (local-exec):
  local_file.k8s_config (local-exec):
  "PriorityLevelConfiguration": {
    local_file.k8s_config (local-exec):
    local_file.k8s_config (local-exec):
    "prioritylevelconfigurations",
    local_file.k8s_config (local-exec):
    local_file.k8s_config (local-exec):
    local_file.k8s_config (local-exec):
    local_file.k8s_config (local-exec):
    "prioritylevelconfiguration"
    local_file.k8s_config (local-exec):
    local_file.k8s_config (local-exec):
    local_file.k8s_config (local-exec):
    local_file.k8s_config (local-exec):
    local_file.k8s_config (local-exec):
    local_file.k8s_config (local-exec):
    local_file.k8s_config (local-exec):
    local_file.k8s_config (local-exec):
    local_file.k8s_config (local-exec):
    local_file.k8s_config (local-exec):
    "ingress"
    local_file.k8s_config (local-exec):
    local_file.k8s_config (local-exec):
    local_file.k8s_config (local-exec):
    local_file.k8s_config (local-exec):
    "ingressclasses",
    local_file.k8s_config (local-exec):
    local_file.k8s_config (local-exec):
    local_file.k8s_config (local-exec):
    local_file.k8s_config (local-exec):
    "ingressclass"

"FlowSchema": {
  "categories": [],
  "name": "flowschemas",
  "namespaced": false,
  "preferred": true,
  "short_names": [],
  "singular_name":

},

  "categories": [],
  "name":

  "namespaced": false,
  "preferred": true,
  "short_names": [],
  "singular_name":

}
},
"networking.k8s.io/v1": {
  "Ingress": {
    "categories": [],
    "name": "ingresses",
    "namespaced": true,
    "preferred": true,
    "short_names": [
      "ing"
    ],
    "singular_name":

  },
  "IngressClass": {
    "categories": [],
    "name":

    "namespaced": false,
    "preferred": true,
    "short_names": [],
    "singular_name":

```



```

local_file.k8s_config (local-exec): },
local_file.k8s_config (local-exec): "NetworkPolicy": {
local_file.k8s_config (local-exec):     "categories": [],
local_file.k8s_config (local-exec):     "name":
"networkpolicies",
local_file.k8s_config (local-exec):     "namespaced": true,
local_file.k8s_config (local-exec):     "preferred": true,
local_file.k8s_config (local-exec):     "short_names": [
local_file.k8s_config (local-exec):         "netpol"
local_file.k8s_config (local-exec):     ],
local_file.k8s_config (local-exec):     "singular_name":
"networkpolicy"
local_file.k8s_config (local-exec): }
local_file.k8s_config (local-exec): },
local_file.k8s_config (local-exec): "node.k8s.io/v1": {
local_file.k8s_config (local-exec):     "RuntimeClass": {
local_file.k8s_config (local-exec):         "categories": [],
local_file.k8s_config (local-exec):         "name":
"runtimeclasses",
local_file.k8s_config (local-exec):         "namespaced": false,
local_file.k8s_config (local-exec):         "preferred": true,
local_file.k8s_config (local-exec):         "short_names": [],
local_file.k8s_config (local-exec):         "singular_name":
"runtimeclass"
local_file.k8s_config (local-exec):     }
local_file.k8s_config (local-exec): },
local_file.k8s_config (local-exec): "policy/v1": {
local_file.k8s_config (local-exec):     "PodDisruptionBudget": {
local_file.k8s_config (local-exec):         "categories": [],
local_file.k8s_config (local-exec):         "name":
"poddisruptionbudgets",
local_file.k8s_config (local-exec):         "namespaced": true,
local_file.k8s_config (local-exec):         "preferred": true,
local_file.k8s_config (local-exec):         "short_names": [
local_file.k8s_config (local-exec):             "pdb"
local_file.k8s_config (local-exec):         ],
local_file.k8s_config (local-exec):         "singular_name":
"poddisruptionbudget"
local_file.k8s_config (local-exec):     }
local_file.k8s_config (local-exec): },
local_file.k8s_config (local-exec): "rbac.authorization.k8s.io/v1": {
local_file.k8s_config (local-exec):     "ClusterRole": {

```



```

local_file.k8s_config (local-exec):      "categories": [],
local_file.k8s_config (local-exec):      "name":
"clusterroles",
local_file.k8s_config (local-exec):      "namespaced": false,
local_file.k8s_config (local-exec):      "preferred": true,
local_file.k8s_config (local-exec):      "short_names": [],
local_file.k8s_config (local-exec):      "singular_name":
"clusterrole"
local_file.k8s_config (local-exec):      },
local_file.k8s_config (local-exec):      "ClusterRoleBinding": {
local_file.k8s_config (local-exec):          "categories": [],
local_file.k8s_config (local-exec):          "name":
"clusterrolebindings",
local_file.k8s_config (local-exec):          "namespaced": false,
local_file.k8s_config (local-exec):          "preferred": true,
local_file.k8s_config (local-exec):          "short_names": [],
local_file.k8s_config (local-exec):          "singular_name":
"clusterrolebinding"
local_file.k8s_config (local-exec):      },
local_file.k8s_config (local-exec):      "Role": {
local_file.k8s_config (local-exec):          "categories": [],
local_file.k8s_config (local-exec):          "name": "roles",
local_file.k8s_config (local-exec):          "namespaced": true,
local_file.k8s_config (local-exec):          "preferred": true,
local_file.k8s_config (local-exec):          "short_names": [],
local_file.k8s_config (local-exec):          "singular_name":
"role"
local_file.k8s_config (local-exec):      },
local_file.k8s_config (local-exec):      "RoleBinding": {
local_file.k8s_config (local-exec):          "categories": [],
local_file.k8s_config (local-exec):          "name":
"rolebindings",
local_file.k8s_config (local-exec):          "namespaced": true,
local_file.k8s_config (local-exec):          "preferred": true,
local_file.k8s_config (local-exec):          "short_names": [],
local_file.k8s_config (local-exec):          "singular_name":
"rolebinding"
local_file.k8s_config (local-exec):      }
local_file.k8s_config (local-exec):  },
local_file.k8s_config (local-exec):  "scheduling.k8s.io/v1": {
local_file.k8s_config (local-exec):      "PriorityClass": {
local_file.k8s_config (local-exec):          "categories": [],
local_file.k8s_config (local-exec):          "name":

```

```

"priorityclasses",
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
"priorityclass"
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
k8s.io/v1": {
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
"secretproviderclasses",
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
"secretproviderclass"
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
"SecretProviderClassPodStatus": {
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
"secretproviderclasspodstatuses",
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
"secretproviderclasspodstatus"
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
k8s.io/v1alpha1": {
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
"secretproviderclasses",
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):

"namespaced": false,
"preferred": true,
"short_names": [
    "pc"
],
"singular_name":

}
},
"secrets-store.csi.x-

"SecretProviderClass": {
    "categories": [],
    "name":

    "namespaced": true,
    "preferred": true,
    "short_names": [],
    "singular_name":

},

    "categories": [],
    "name":

    "namespaced": true,
    "preferred": true,
    "short_names": [],
    "singular_name":

}
},
"secrets-store.csi.x-

"SecretProviderClass": {
    "categories": [],
    "name":

    "namespaced": true,
    "preferred": false,
    "short_names": [],

```

```

local_file.k8s_config (local-exec):           "singular_name":
"secretproviderclass"
local_file.k8s_config (local-exec):           },
local_file.k8s_config (local-exec):
"SecretProviderClassPodStatus": {
local_file.k8s_config (local-exec):           "categories": [],
local_file.k8s_config (local-exec):           "name":
"secretproviderclasspodstatuses",
local_file.k8s_config (local-exec):           "namespaced": true,
local_file.k8s_config (local-exec):           "preferred": false,
local_file.k8s_config (local-exec):           "short_names": [],
local_file.k8s_config (local-exec):           "singular_name":
"secretproviderclasspodstatus"
local_file.k8s_config (local-exec):           }
local_file.k8s_config (local-exec):           },
local_file.k8s_config (local-exec):           "snapshot.storage.k8s.io/v1":
{
local_file.k8s_config (local-exec):           "VolumeSnapshot": {
local_file.k8s_config (local-exec):           "categories": [],
local_file.k8s_config (local-exec):           "name":
"volumesnapshots",
local_file.k8s_config (local-exec):           "namespaced": true,
local_file.k8s_config (local-exec):           "preferred": true,
local_file.k8s_config (local-exec):           "short_names": [
local_file.k8s_config (local-exec):           "vs"
local_file.k8s_config (local-exec):           ],
local_file.k8s_config (local-exec):           "singular_name":
"volumesnapshot"
local_file.k8s_config (local-exec):           },
local_file.k8s_config (local-exec):           "VolumeSnapshotClass": {
local_file.k8s_config (local-exec):           "categories": [],
local_file.k8s_config (local-exec):           "name":
"volumesnapshotclasses",
local_file.k8s_config (local-exec):           "namespaced": false,
local_file.k8s_config (local-exec):           "preferred": true,
local_file.k8s_config (local-exec):           "short_names": [
local_file.k8s_config (local-exec):           "vsclass",
local_file.k8s_config (local-exec):           "vsclasses"
local_file.k8s_config (local-exec):           ],
local_file.k8s_config (local-exec):           "singular_name":
"volumesnapshotclass"
local_file.k8s_config (local-exec):           },
local_file.k8s_config (local-exec):           "VolumeSnapshotContent": {

```

```

local_file.k8s_config (local-exec):      "categories": [],
local_file.k8s_config (local-exec):      "name":
"volumesnapshotcontents",
local_file.k8s_config (local-exec):      "namespaced": false,
local_file.k8s_config (local-exec):      "preferred": true,
local_file.k8s_config (local-exec):      "short_names": [
local_file.k8s_config (local-exec):          "vsc",
local_file.k8s_config (local-exec):          "vscs"
local_file.k8s_config (local-exec):      ],
local_file.k8s_config (local-exec):      "singular_name":
"volumesnapshotcontent"
local_file.k8s_config (local-exec):      }
local_file.k8s_config (local-exec):  },
local_file.k8s_config (local-exec):  "storage.k8s.io/v1": {
local_file.k8s_config (local-exec):      "CSIDriver": {
local_file.k8s_config (local-exec):          "categories": [],
local_file.k8s_config (local-exec):          "name": "csidrivers",
local_file.k8s_config (local-exec):          "namespaced": false,
local_file.k8s_config (local-exec):          "preferred": true,
local_file.k8s_config (local-exec):          "short_names": [],
local_file.k8s_config (local-exec):          "singular_name":
"csidriver"
local_file.k8s_config (local-exec):      },
local_file.k8s_config (local-exec):      "CSINode": {
local_file.k8s_config (local-exec):          "categories": [],
local_file.k8s_config (local-exec):          "name": "csinodes",
local_file.k8s_config (local-exec):          "namespaced": false,
local_file.k8s_config (local-exec):          "preferred": true,
local_file.k8s_config (local-exec):          "short_names": [],
local_file.k8s_config (local-exec):          "singular_name":
"csinode"
local_file.k8s_config (local-exec):      },
local_file.k8s_config (local-exec):      "CSIStorageCapacity": {
local_file.k8s_config (local-exec):          "categories": [],
local_file.k8s_config (local-exec):          "name":
"csistoragecapacities",
local_file.k8s_config (local-exec):          "namespaced": true,
local_file.k8s_config (local-exec):          "preferred": true,
local_file.k8s_config (local-exec):          "short_names": [],
local_file.k8s_config (local-exec):          "singular_name":
"csistoragecapacity"
local_file.k8s_config (local-exec):      },
local_file.k8s_config (local-exec):      "StorageClass": {

```

```

local_file.k8s_config (local-exec):      "categories": [],
local_file.k8s_config (local-exec):      "name":
"storageclasses",
local_file.k8s_config (local-exec):      "namespaced": false,
local_file.k8s_config (local-exec):      "preferred": true,
local_file.k8s_config (local-exec):      "short_names": [
local_file.k8s_config (local-exec):          "sc"
local_file.k8s_config (local-exec):      ],
local_file.k8s_config (local-exec):      "singular_name":
"storageclass"
local_file.k8s_config (local-exec):      },
local_file.k8s_config (local-exec):      "VolumeAttachment": {
local_file.k8s_config (local-exec):          "categories": [],
local_file.k8s_config (local-exec):          "name":
"namespaced": false,
local_file.k8s_config (local-exec):          "preferred": true,
local_file.k8s_config (local-exec):          "short_names": [],
local_file.k8s_config (local-exec):          "singular_name":
"volumeattachments",
local_file.k8s_config (local-exec):      "namespaced": false,
local_file.k8s_config (local-exec):      "preferred": true,
local_file.k8s_config (local-exec):      "short_names": [],
local_file.k8s_config (local-exec):      "singular_name":
"volumeattachment"
local_file.k8s_config (local-exec):      }
local_file.k8s_config (local-exec):      },
local_file.k8s_config (local-exec):      "v1": {
local_file.k8s_config (local-exec):          "Binding": {
local_file.k8s_config (local-exec):              "categories": [],
local_file.k8s_config (local-exec):              "name": "bindings",
local_file.k8s_config (local-exec):              "namespaced": true,
local_file.k8s_config (local-exec):              "preferred": true,
local_file.k8s_config (local-exec):              "short_names": [],
local_file.k8s_config (local-exec):              "singular_name":
"binding"
local_file.k8s_config (local-exec):          },
local_file.k8s_config (local-exec):          "ComponentStatus": {
local_file.k8s_config (local-exec):              "categories": [],
local_file.k8s_config (local-exec):              "name":
"namespaced": false,
local_file.k8s_config (local-exec):              "preferred": true,
local_file.k8s_config (local-exec):              "short_names": [
local_file.k8s_config (local-exec):                  "cs"
local_file.k8s_config (local-exec):              ],
local_file.k8s_config (local-exec):              "singular_name":
"componentstatuses",
local_file.k8s_config (local-exec):      "namespaced": false,
local_file.k8s_config (local-exec):      "preferred": true,
local_file.k8s_config (local-exec):      "short_names": [
local_file.k8s_config (local-exec):          "cs"
local_file.k8s_config (local-exec):      ],
local_file.k8s_config (local-exec):      "singular_name":
"componentstatus"

```

```

local_file.k8s_config (local-exec): },
local_file.k8s_config (local-exec): "ConfigMap": {
local_file.k8s_config (local-exec):     "categories": [],
local_file.k8s_config (local-exec):     "name": "configmaps",
local_file.k8s_config (local-exec):     "namespaced": true,
local_file.k8s_config (local-exec):     "preferred": true,
local_file.k8s_config (local-exec):     "short_names": [
local_file.k8s_config (local-exec):         "cm"
local_file.k8s_config (local-exec):     ],
local_file.k8s_config (local-exec):     "singular_name":
"configmap"
local_file.k8s_config (local-exec): },
local_file.k8s_config (local-exec): "Endpoints": {
local_file.k8s_config (local-exec):     "categories": [],
local_file.k8s_config (local-exec):     "name": "endpoints",
local_file.k8s_config (local-exec):     "namespaced": true,
local_file.k8s_config (local-exec):     "preferred": true,
local_file.k8s_config (local-exec):     "short_names": [
local_file.k8s_config (local-exec):         "ep"
local_file.k8s_config (local-exec):     ],
local_file.k8s_config (local-exec):     "singular_name":
"endpoints"
local_file.k8s_config (local-exec): },
local_file.k8s_config (local-exec): "Event": {
local_file.k8s_config (local-exec):     "categories": [],
local_file.k8s_config (local-exec):     "name": "events",
local_file.k8s_config (local-exec):     "namespaced": true,
local_file.k8s_config (local-exec):     "preferred": true,
local_file.k8s_config (local-exec):     "short_names": [
local_file.k8s_config (local-exec):         "ev"
local_file.k8s_config (local-exec):     ],
local_file.k8s_config (local-exec):     "singular_name":
"event"
local_file.k8s_config (local-exec): },
local_file.k8s_config (local-exec): "LimitRange": {
local_file.k8s_config (local-exec):     "categories": [],
local_file.k8s_config (local-exec):     "name": "limitranges",
local_file.k8s_config (local-exec):     "namespaced": true,
local_file.k8s_config (local-exec):     "preferred": true,
local_file.k8s_config (local-exec):     "short_names": [
local_file.k8s_config (local-exec):         "limits"
local_file.k8s_config (local-exec):     ],
local_file.k8s_config (local-exec):     "singular_name":

```

```

"limitrange"
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
"namespace"
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
"node"
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
"persistentvolumes",
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
"persistentvolume"

},
"List": {
    "categories": [],
    "name": null,
    "namespaced": null,
    "preferred": null,
    "short_names": [],
    "singular_name": null
},
"Namespace": {
    "categories": [],
    "name": "namespaces",
    "namespaced": false,
    "preferred": true,
    "short_names": [
        "ns"
    ],
    "singular_name":
},
"Node": {
    "categories": [],
    "name": "nodes",
    "namespaced": false,
    "preferred": true,
    "short_names": [
        "no"
    ],
    "singular_name":
},
"PersistentVolume": {
    "categories": [],
    "name":
    "namespaced": false,
    "preferred": true,
    "short_names": [
        "pv"
    ],
    "singular_name":

```



```

local_file.k8s_config (local-exec): },
local_file.k8s_config (local-exec): "PersistentVolumeClaim": {
local_file.k8s_config (local-exec):     "categories": [],
local_file.k8s_config (local-exec):     "name":
"persistentvolumeclaims",
local_file.k8s_config (local-exec):     "namespaced": true,
local_file.k8s_config (local-exec):     "preferred": true,
local_file.k8s_config (local-exec):     "short_names": [
local_file.k8s_config (local-exec):         "pvc"
local_file.k8s_config (local-exec):     ],
local_file.k8s_config (local-exec):     "singular_name":
"persistentvolumeclaim"
local_file.k8s_config (local-exec): },
local_file.k8s_config (local-exec): "Pod": {
local_file.k8s_config (local-exec):     "categories": [
local_file.k8s_config (local-exec):         "all"
local_file.k8s_config (local-exec):     ],
local_file.k8s_config (local-exec):     "name": "pods",
local_file.k8s_config (local-exec):     "namespaced": true,
local_file.k8s_config (local-exec):     "preferred": true,
local_file.k8s_config (local-exec):     "short_names": [
local_file.k8s_config (local-exec):         "po"
local_file.k8s_config (local-exec):     ],
local_file.k8s_config (local-exec):     "singular_name": "pod"
local_file.k8s_config (local-exec): },
local_file.k8s_config (local-exec): "PodTemplate": {
local_file.k8s_config (local-exec):     "categories": [],
local_file.k8s_config (local-exec):     "name":
"podtemplates",
local_file.k8s_config (local-exec):     "namespaced": true,
local_file.k8s_config (local-exec):     "preferred": true,
local_file.k8s_config (local-exec):     "short_names": [],
local_file.k8s_config (local-exec):     "singular_name":
"podtemplate"
local_file.k8s_config (local-exec): },
local_file.k8s_config (local-exec): "ReplicationController": {
local_file.k8s_config (local-exec):     "categories": [
local_file.k8s_config (local-exec):         "all"
local_file.k8s_config (local-exec):     ],
local_file.k8s_config (local-exec):     "name":
"replicationcontrollers",
local_file.k8s_config (local-exec):     "namespaced": true,
local_file.k8s_config (local-exec):     "preferred": true,

```



```

local_file.k8s_config (local-exec):           "short_names": [
local_file.k8s_config (local-exec):           "rc"
local_file.k8s_config (local-exec):           ],
local_file.k8s_config (local-exec):           "singular_name":
"replicationcontroller"
local_file.k8s_config (local-exec):           },
local_file.k8s_config (local-exec):           "ResourceQuota": {
local_file.k8s_config (local-exec):           "categories": [],
local_file.k8s_config (local-exec):           "name":
"resourcequotas",
local_file.k8s_config (local-exec):           "namespaced": true,
local_file.k8s_config (local-exec):           "preferred": true,
local_file.k8s_config (local-exec):           "short_names": [
local_file.k8s_config (local-exec):           "quota"
local_file.k8s_config (local-exec):           ],
local_file.k8s_config (local-exec):           "singular_name":
"resourcequota"
local_file.k8s_config (local-exec):           },
local_file.k8s_config (local-exec):           "Secret": {
local_file.k8s_config (local-exec):           "categories": [],
local_file.k8s_config (local-exec):           "name": "secrets",
local_file.k8s_config (local-exec):           "namespaced": true,
local_file.k8s_config (local-exec):           "preferred": true,
local_file.k8s_config (local-exec):           "short_names": [],
local_file.k8s_config (local-exec):           "singular_name":
"secret"
local_file.k8s_config (local-exec):           },
local_file.k8s_config (local-exec):           "Service": {
local_file.k8s_config (local-exec):           "categories": [
local_file.k8s_config (local-exec):           "all"
local_file.k8s_config (local-exec):           ],
local_file.k8s_config (local-exec):           "name": "services",
local_file.k8s_config (local-exec):           "namespaced": true,
local_file.k8s_config (local-exec):           "preferred": true,
local_file.k8s_config (local-exec):           "short_names": [
local_file.k8s_config (local-exec):           "svc"
local_file.k8s_config (local-exec):           ],
local_file.k8s_config (local-exec):           "singular_name":
"service"
local_file.k8s_config (local-exec):           },
local_file.k8s_config (local-exec):           "ServiceAccount": {
local_file.k8s_config (local-exec):           "categories": [],
local_file.k8s_config (local-exec):           "name":

```

```

"serviceaccounts",
local_file.k8s_config (local-exec):           "namespaced": true,
local_file.k8s_config (local-exec):           "preferred": true,
local_file.k8s_config (local-exec):           "short_names": [
local_file.k8s_config (local-exec):             "sa"
local_file.k8s_config (local-exec):           ],
local_file.k8s_config (local-exec):           "singular_name":
"serviceaccount"
local_file.k8s_config (local-exec):           }
local_file.k8s_config (local-exec):           }
local_file.k8s_config (local-exec):           },
local_file.k8s_config (local-exec):           "changed": false,
local_file.k8s_config (local-exec):           "connection": {
local_file.k8s_config (local-exec):             "cert_file":
"/tmp/tmp9eaz8jc5",
local_file.k8s_config (local-exec):             "host": "https://unir-
ik5kq7b4.hcp.northeurope.azurek8s.io:443",
local_file.k8s_config (local-exec):             "password": null,
local_file.k8s_config (local-exec):             "proxy": null,
local_file.k8s_config (local-exec):             "ssl_ca_cert":
"/tmp/tmpo63m3of2",
local_file.k8s_config (local-exec):             "username": null,
local_file.k8s_config (local-exec):             "verify_ssl": true
local_file.k8s_config (local-exec):           },
local_file.k8s_config (local-exec):           "failed": false,
local_file.k8s_config (local-exec):           "version": {
local_file.k8s_config (local-exec):             "client": "28.1.0",
local_file.k8s_config (local-exec):             "server": {
local_file.k8s_config (local-exec):               "kubernetes": {
local_file.k8s_config (local-exec):                 "buildDate": "2024-05-
22T15:00:24Z",
local_file.k8s_config (local-exec):                 "compiler": "gc",
local_file.k8s_config (local-exec):                 "gitCommit":
"21be1d76a90bc00e2b0f6676a664bdf097224155",
local_file.k8s_config (local-exec):                 "gitTreeState":
"clean",
local_file.k8s_config (local-exec):                 "gitVersion":
"v1.28.10",
local_file.k8s_config (local-exec):                 "goVersion":
"go1.21.9",
local_file.k8s_config (local-exec):                 "major": "1",
local_file.k8s_config (local-exec):                 "minor": "28",
local_file.k8s_config (local-exec):                 "platform":

```

```

"linux/amd64"

local_file.k8s_config (local-exec):          }
local_file.k8s_config (local-exec):          }
local_file.k8s_config (local-exec):          }
local_file.k8s_config (local-exec):          }
local_file.k8s_config (local-exec): }

local_file.k8s_config (local-exec): TASK [aks : include_tasks]
*****

local_file.k8s_config (local-exec): included:
/home/dani/Documents/asignaturas/unir/devops/actividades/act2/actividad2/IaC
/ansible/roles/aks/tasks/wp.yml for localhost

local_file.k8s_config (local-exec): TASK [aks : Create namespace wp]
*****

local_file.k8s_config (local-exec): ok: [localhost] => {"changed": false,
"method": "update", "result": {"apiVersion": "v1", "kind": "Namespace",
"metadata": {"creationTimestamp": "2024-07-15T15:39:56Z", "labels":
{"kubernetes.io/metadata.name": "wp"}, "managedFields": [{"apiVersion":
"v1", "fieldsType": "FieldsV1", "fieldsV1": {"f:metadata": {"f:labels": {".":
{}}, "f:kubernetes.io/metadata.name": {}}}}, {"manager": "kubectl-create",
"operation": "Update", "time": "2024-07-15T15:39:56Z"}], "name": "wp",
"resourceVersion": "128344", "uid": "654c3e98-4196-4150-a027-d1fb993e9b96"},
"spec": {"finalizers": ["kubernetes"]}], "status": {"phase": "Active"}}}

local_file.k8s_config (local-exec): TASK [aks : Create the deployments]
*****

local_file.k8s_config (local-exec): ok: [localhost] => (item=registry-
credentials.yaml.j2) => {"ansible_loop_var": "item", "changed": false,
"item": "registry-credentials.yaml.j2", "method": "update", "result":
{"apiVersion": "v1", "data": {".dockerconfigjson":
"eyJhdXRob3RocyI6eyJhY3JlbmlyLmF6dXJlY3IuaW8iOnsidXNlcm5hbWUiOiJhY3JlbmlyIiwicGF
zc3dvcmQiOiJlam9JZGhCeU1qVkd1QytETW9hQkVBTkltby9ialVJdmx2WkRNN0ZNNG4rQUNSQnZ
iMVhvIiwiaXV0aCI6I1lXTnlkVzVwY2pwbGFtOUpaR2hDZVUxcVZrZDFReXRFVFc5aFFrVkJUa2x
0Ynk5aWFsVkpkbXgyV2tSTk4wWk5ORzRyUVVOU1FuWmlNVmh2In19fQ=="}, "kind":
"Secret", "metadata": {"creationTimestamp": "2024-07-16T09:43:40Z",
"managedFields": [{"apiVersion": "v1", "fieldsType": "FieldsV1", "fieldsV1":
{"f:data": {".": {}}, "f:.dockerconfigjson": {}}, {"f:type": {}}, {"manager":
"kubectl-create", "operation": "Update", "time": "2024-07-16T09:43:40Z"}],
"name": "registry-credentials", "namespace": "wp", "resourceVersion":
"362904", "uid": "30d845da-6578-46ad-a0ae-dc9174d6535f"}, "type":
"kubernetes.io/dockerconfigjson"}}
local_file.k8s_config (local-exec): ok: [localhost] => (item=mysql-

```

```
deployment.yaml.j2) => {"ansible_loop_var": "item", "changed": false,
"item": "mysql-deployment.yaml.j2", "result": {"results": [{"changed":
false, "method": "update", "result": {"apiVersion": "v1", "kind":
"Namespace", "metadata": {"creationTimestamp": "2024-07-15T15:39:56Z",
"labels": {"kubernetes.io/metadata.name": "wp"}, "managedFields":
[{"apiVersion": "v1", "fieldsType": "FieldsV1", "fieldsV1": {"f:metadata":
{"f:labels": {".": {}, "f:kubernetes.io/metadata.name": {}}}}, {"manager":
"kubectl-create", "operation": "Update", "time": "2024-07-15T15:39:56Z"}]},
"name": "wp", "resourceVersion": "128344", "uid": "654c3e98-4196-4150-a027-
d1fb993e9b96"}, {"spec": {"finalizers": ["kubernetes"]}, "status": {"phase":
"Active"}}}], {"changed": false, "method": "update", "result": {"apiVersion":
"v1", "kind": "Service", "metadata": {"creationTimestamp": "2024-07-
16T06:16:55Z", "labels": {"app": "wordpress"}, "managedFields":
[{"apiVersion": "v1", "fieldsType": "FieldsV1", "fieldsV1": {"f:metadata":
{"f:labels": {".": {}, "f:app": {}}}, {"f:spec": {"f:clusterIP": {},
"f:internalTrafficPolicy": {}, "f:ports": {".": {}, "k:
{"port": 3306, "protocol": "TCP"}": {".": {}, "f:port": {}, "f:protocol":
{}, "f:targetPort": {}}}, {"f:selector": {}, "f:sessionAffinity": {}, "f:type":
{}}}, {"manager": "OpenAPI-Generator", "operation": "Update", "time": "2024-
07-16T06:16:55Z"}]}, {"name": "wordpress-mysql", "namespace": "wp",
"resourceVersion": "318116", "uid": "ac439768-36e7-4f83-8509-8e1724c68d77"},
"spec": {"clusterIP": "None", "clusterIPs": ["None"], "internalTrafficPolicy":
"Cluster", "ipFamilies": ["IPv4"], "ipFamilyPolicy": "SingleStack", "ports":
[{"port": 3306, "protocol": "TCP", "targetPort": 3306}], "selector": {"app":
"wordpress", "tier": "mysql"}, "sessionAffinity": "None", "type":
"ClusterIP"}, {"status": {"loadBalancer": {}}}], {"changed": false, "method":
"update", "result": {"apiVersion": "v1", "kind": "PersistentVolumeClaim",
"metadata": {"annotations": {"pv.kubernetes.io/bind-completed": "yes",
"pv.kubernetes.io/bound-by-controller": "yes",
"volume.beta.kubernetes.io/storage-provisioner": "disk.csi.azure.com",
"volume.kubernetes.io/selected-node": "aks-default-12959642-vmss000000",
"volume.kubernetes.io/storage-provisioner": "disk.csi.azure.com"},
"creationTimestamp": "2024-07-16T06:17:38Z", "finalizers":
["kubernetes.io/pvc-protection"], "labels": {"app": "wordpress"},
"managedFields": [{"apiVersion": "v1", "fieldsType": "FieldsV1", "fieldsV1":
{"f:metadata": {"f:labels": {".": {}, "f:app": {}}}, {"f:spec":
{"f:accessModes": {}, "f:resources": {"f:requests": {".": {}, "f:storage":
{}}}, {"f:volumeMode": {}}}, {"manager": "OpenAPI-Generator", "operation":
"Update", "time": "2024-07-16T06:17:38Z"}, {"apiVersion": "v1", "fieldsType":
"FieldsV1", "fieldsV1": {"f:metadata": {"f:annotations": {".": {},
"f:volume.kubernetes.io/selected-node": {}}}], {"manager": "kube-scheduler",
"operation": "Update", "time": "2024-07-16T06:17:39Z"}, {"apiVersion": "v1",
"fieldsType": "FieldsV1", "fieldsV1": {"f:metadata": {"f:annotations":
```

```
{"f:pv.kubernetes.io/bind-completed": {}, "f:pv.kubernetes.io/bound-by-controller": {}, "f:volume.beta.kubernetes.io/storage-provisioner": {}, "f:volume.kubernetes.io/storage-provisioner": {}}, "f:spec": {"f:volumeName": {}}, "manager": "kube-controller-manager", "operation": "Update", "time": "2024-07-16T06:17:42Z"}, {"apiVersion": "v1", "fieldsType": "FieldsV1", "fieldsV1": {"f:status": {"f:accessModes": {}, "f:capacity": {"f:storage": {}}, "f:phase": {}}, "manager": "kube-controller-manager", "operation": "Update", "subresource": "status", "time": "2024-07-16T06:17:42Z"}], "name": "mysql-pv-claim", "namespace": "wp", "resourceVersion": "318310", "uid": "7d11c5ed-cb37-474d-b555-a79570d4208b"}, "spec": {"accessModes": ["ReadWriteOnce"], "resources": {"requests": {"storage": "20Gi"}}, "storageClassName": "default", "volumeMode": "Filesystem", "volumeName": "pvc-7d11c5ed-cb37-474d-b555-a79570d4208b"}, "status": {"accessModes": ["ReadWriteOnce"], "capacity": {"storage": "20Gi"}, "phase": "Bound"}}, {"changed": false, "method": "update", "result": {"apiVersion": "apps/v1", "kind": "Deployment", "metadata": {"annotations": {"deployment.kubernetes.io/revision": "1"}, "creationTimestamp": "2024-07-16T10:42:03Z", "generation": 1, "labels": {"app": "wordpress"}, "managedFields": [{"apiVersion": "apps/v1", "fieldsType": "FieldsV1", "fieldsV1": {"f:metadata": {"f:labels": {"f:app": {}}, "f:spec": {"f:progressDeadlineSeconds": {}, "f:replicas": {}, "f:revisionHistoryLimit": {}, "f:selector": {}, "f:strategy": {"f:type": {}}, "f:template": {"f:metadata": {"f:labels": {"f:app": {}}, "f:tier": {}}, "f:spec": {"f:containers": {"k:{"name": "mysql"}": {"f:env": {"f:name": {}, "f:value": {}}, "k:{"name": "MYSQL_DATABASE"}": {"f:name": {}, "f:value": {}}, "k:{"name": "MYSQL_PASSWORD"}": {"f:name": {}, "f:value": {}}, "k:{"name": "MYSQL_ROOT_PASSWORD"}": {"f:name": {}, "f:value": {}}, "k:{"name": "MYSQL_USER"}": {"f:name": {}, "f:value": {}}, "f:image": {}, "f:imagePullPolicy": {}, "f:ports": {"f:containerPort": {"f:protocol": {"f:terminationMessagePath": {}, "f:terminationMessagePolicy": {}}, "f:volumeMounts": {"k:{"mountPath": "/var/lib/mysql"}": {"f:mountPath": {}, "f:name": {}}, "f:dnsPolicy": {}, "f:imagePullSecrets": {"k:{"name": "registry-credentials"}": {"f:restartPolicy": {}, "f:schedulerName": {}, "f:securityContext": {}, "f:terminationGracePeriodSeconds": {}, "f:volumes": {"k:{"name": "mysql-persistent-storage"}": {"f:persistentVolumeClaim": {"f:claimName": {}}, "manager": "OpenAPI-Generator", "operation": "Update", "time": "2024-07-16T10:42:03Z"}, {"apiVersion": "apps/v1", "fieldsType": "FieldsV1", "fieldsV1": {"f:metadata": {"f:annotations": {"f:deployment.kubernetes.io/revision": {}},
```



```
"f:status": {"f:conditions": {".": {}, "k:{"type": "Available"}": {".": {}, "f:lastTransitionTime": {}, "f:lastUpdateTime": {}, "f:message": {}, "f:reason": {}, "f:status": {}, "f:type": {}}}, "k: {"type": "Progressing"}": {".": {}, "f:lastTransitionTime": {}, "f:lastUpdateTime": {}, "f:message": {}, "f:reason": {}, "f:status": {}, "f:type": {}}}, "f:observedGeneration": {}, "f:replicas": {}, "f:unavailableReplicas": {}, "f:updatedReplicas": {}}}, "manager": "kube-controller-manager", "operation": "Update", "subresource": "status", "time": "2024-07-16T11:14:15Z", "name": "wordpress-mysql", "namespace": "wp", "resourceVersion": "382735", "uid": "e741290a-0498-4b6a-b6a1-eeda210430ae", "spec": {"progressDeadlineSeconds": 600, "replicas": 1, "revisionHistoryLimit": 10, "selector": {"matchLabels": {"app": "wordpress", "tier": "mysql"}}, "strategy": {"type": "Recreate"}, "template": {"metadata": {"creationTimestamp": null, "labels": {"app": "wordpress", "tier": "mysql"}}, "spec": {"containers": [{"env": [{"name": "MYSQL_ROOT_PASSWORD", "value": "pepe"}, {"name": "MYSQL_DATABASE", "value": "wordpress"}, {"name": "MYSQL_USER", "value": "wordpress"}, {"name": "MYSQL_PASSWORD", "value": "pepe"}], "image": "acrunir.azurecr.io/mysql:8.0", "imagePullPolicy": "IfNotPresent", "name": "mysql", "ports": [{"containerPort": 3306, "name": "mysql", "protocol": "TCP"}], "resources": {}, "terminationMessagePath": "/dev/termination-log", "terminationMessagePolicy": "File", "volumeMounts": [{"mountPath": "/var/lib/mysql", "name": "mysql-persistent-storage"}]}], "dnsPolicy": "ClusterFirst", "imagePullSecrets": [{"name": "registry-credentials"}], "restartPolicy": "Always", "schedulerName": "default-scheduler", "securityContext": {}, "terminationGracePeriodSeconds": 30, "volumes": [{"name": "mysql-persistent-storage", "persistentVolumeClaim": {"claimName": "mysql-pv-claim"}}]}], "status": {"conditions": [{"lastTransitionTime": "2024-07-16T10:42:03Z", "lastUpdateTime": "2024-07-16T10:42:15Z", "message": "ReplicaSet \"wordpress-mysql-864bdc5c56\" has successfully progressed.", "reason": "NewReplicaSetAvailable", "status": "True", "type": "Progressing"}, {"lastTransitionTime": "2024-07-16T11:14:15Z", "lastUpdateTime": "2024-07-16T11:14:15Z", "message": "Deployment does not have minimum availability.", "reason": "MinimumReplicasUnavailable", "status": "False", "type": "Available"}], "observedGeneration": 1, "replicas": 1, "unavailableReplicas": 1, "updatedReplicas": 1}}]}
local_file.k8s_config: Still creating... [10s elapsed]
local_file.k8s_config (local-exec): ok: [localhost] => (item=wordpress-deployment.yaml.j2) => {"ansible_loop_var": "item", "changed": false, "item": "wordpress-deployment.yaml.j2", "result": {"results": [{"changed": false, "method": "update", "result": {"apiVersion": "v1", "kind": "Service", "metadata": {"creationTimestamp": "2024-07-16T08:34:52Z", "finalizers": ["service.kubernetes.io/load-balancer-cleanup"], "labels": {"app":
```

```

"wordpress"}, {"managedFields": [{"apiVersion": "v1", "fieldsType":
"FieldsV1", "fieldsV1": {"f:metadata": {"f:labels": {".": {}}, "f:app": {}}},
"f:spec": {"f:allocateLoadBalancerNodePorts": {}, "f:externalTrafficPolicy":
{}, "f:internalTrafficPolicy": {}, "f:ports": {".": {}}, "k:
{"port": 80, "protocol": "TCP"}": {"f:port": {}, "f:protocol":
{}, "f:targetPort": {}}}, {"f:selector": {}, "f:sessionAffinity": {}, "f:type":
{}}}, {"manager": "OpenAPI-Generator", "operation": "Update", "time": "2024-
07-16T08:34:52Z"}, {"apiVersion": "v1", "fieldsType": "FieldsV1", "fieldsV1":
{"f:metadata": {"f:finalizers": {".": {}}, "v:service.kubernetes.io/load-
balancer-cleanup": {}}}, {"f:status": {"f:loadBalancer": {"f:ingress":
{}}}}, {"manager": "cloud-controller-manager", "operation": "Update",
"subresource": "status", "time": "2024-07-16T10:30:35Z"}], "name":
"wordpress", "namespace": "wp", "resourceVersion": "373226", "uid":
"58727b9b-0af6-4db6-9056-0448a021a253"}, {"spec":
{"allocateLoadBalancerNodePorts": true, "clusterIP": "10.0.159.20",
"clusterIPs": ["10.0.159.20"], "externalTrafficPolicy": "Cluster",
"internalTrafficPolicy": "Cluster", "ipFamilies": ["IPv4"], "ipFamilyPolicy":
"SingleStack", "ports": [{"nodePort": 32083, "port": 80, "protocol": "TCP",
"targetPort": 80}], "selector": {"app": "wordpress", "tier": "frontend"},
"sessionAffinity": "None", "type": "LoadBalancer"}, {"status": {"loadBalancer":
{"ingress": [{"ip": "135.236.178.212"}]}}, {"changed": false, "method":
"update", "result": {"apiVersion": "v1", "kind": "PersistentVolumeClaim",
"metadata": {"annotations": {"pv.kubernetes.io/bind-completed": "yes",
"pv.kubernetes.io/bound-by-controller": "yes",
"volume.beta.kubernetes.io/storage-provisioner": "disk.csi.azure.com",
"volume.kubernetes.io/selected-node": "aks-default-12959642-vmss000000",
"volume.kubernetes.io/storage-provisioner": "disk.csi.azure.com"},
"creationTimestamp": "2024-07-16T08:34:52Z", "finalizers":
["kubernetes.io/pvc-protection"], "labels": {"app": "wordpress"},
"managedFields": [{"apiVersion": "v1", "fieldsType": "FieldsV1", "fieldsV1":
{"f:metadata": {"f:labels": {".": {}}, "f:app": {}}}, {"f:spec":
{"f:accessModes": {}, "f:resources": {"f:requests": {".": {}}, "f:storage":
{}}}, {"f:volumeMode": {}}}, {"manager": "OpenAPI-Generator", "operation":
"Update", "time": "2024-07-16T08:34:52Z"}, {"apiVersion": "v1", "fieldsType":
"FieldsV1", "fieldsV1": {"f:metadata": {"f:annotations": {".": {}},
"f:volume.kubernetes.io/selected-node": {}}}}, {"manager": "kube-scheduler",
"operation": "Update", "time": "2024-07-16T08:34:52Z"}, {"apiVersion": "v1",
"fieldsType": "FieldsV1", "fieldsV1": {"f:metadata": {"f:annotations":
{"f:pv.kubernetes.io/bind-completed": {}, "f:pv.kubernetes.io/bound-by-
controller": {}, "f:volume.beta.kubernetes.io/storage-provisioner": {},
"f:volume.kubernetes.io/storage-provisioner": {}}}, {"f:spec":
{"f:volumeName": {}}}, {"manager": "kube-controller-manager", "operation":
"Update", "time": "2024-07-16T08:35:13Z"}, {"apiVersion": "v1", "fieldsType":

```

```
"FieldsV1", "fieldsV1": {"f:status": {"f:accessModes": {}, "f:capacity": {"f:storage": {}}, "f:phase": {}}, "manager": "kube-controller-manager", "operation": "Update", "subresource": "status", "time": "2024-07-16T08:35:13Z"}, {"name": "wp-pv-claim", "namespace": "wp", "resourceVersion": "347956", "uid": "568acbcc-d6d7-4c81-8a00-d6c9398377af"}, {"spec": {"accessModes": ["ReadWriteOnce"], "resources": {"requests": {"storage": "20Gi"}}, "storageClassName": "default", "volumeMode": "Filesystem", "volumeName": "pvc-568acbcc-d6d7-4c81-8a00-d6c9398377af"}, {"status": {"accessModes": ["ReadWriteOnce"], "capacity": {"storage": "20Gi"}, "phase": "Bound"}}, {"changed": false, "method": "update", "result": {"apiVersion": "apps/v1", "kind": "Deployment", "metadata": {"annotations": {"deployment.kubernetes.io/revision": "2"}, "creationTimestamp": "2024-07-16T08:34:52Z", "generation": 2, "labels": {"app": "wordpress"}, "managedFields": [{"apiVersion": "apps/v1", "fieldsType": "FieldsV1", "fieldsV1": {"f:metadata": {"f:labels": {"f:app": {}}, "f:spec": {"f:progressDeadlineSeconds": {}, "f:replicas": {}, "f:revisionHistoryLimit": {}, "f:selector": {}, "f:strategy": {"f:type": {}}, "f:template": {"f:metadata": {"f:labels": {"f:app": {}}, "f:tier": {}}, "f:spec": {"f:containers": {"k:{"name":"wordpress"}: {"f:env": {"f:env": {"k:{"name":"WORDPRESS_DB_HOST"}: {"f:name": {"f:value": {}}, "k: {"name":"WORDPRESS_DB_PASSWORD"}: {"f:name": {"f:value": {}}, "k: {"name":"WORDPRESS_DB_USER"}: {"f:name": {"f:value": {}}, "f:image": {"f:imagePullPolicy": {"f:name": {"f:ports": {"f:containerPort": {"f:name": {"f:protocol": {}}, "f:resources": {"f:terminationMessagePath": {"f:terminationMessagePolicy": {"f:volumeMounts": {"f:mountPath": {"f:name": {}}, "f:mountPath": {"f:name": {}}, "f:dnsPolicy": {"f:imagePullSecrets": {"f:restartPolicy": {"f:schedulerName": {"f:securityContext": {"f:terminationGracePeriodSeconds": {"f:volumes": {"f:persistentVolumeClaim": {"f:claimName": {}}, "manager": "OpenAPI-Generator", "operation": "Update", "time": "2024-07-16T09:53:10Z"}, {"apiVersion": "apps/v1", "fieldsType": "FieldsV1", "fieldsV1": {"f:metadata": {"f:annotations": {"f:deployment.kubernetes.io/revision": {}}, "f:status": {"f:conditions": {"f:lastTransitionTime": {"f:lastUpdateTime": {"f:message": {"f:reason": {"f:status": {"f:type": {"k: {"type":"Progressing"}: {"f:lastTransitionTime": {"f:lastUpdateTime": {"f:message": {"f:reason": {"f:status": {"f:type": {}}, "f:observedGeneration": {"f:replicas": {}},
```



```

"f:unavailableReplicas": {}, "f:updatedReplicas": {}}, "manager": "kube-
controller-manager", "operation": "Update", "subresource": "status", "time":
"2024-07-16T11:14:15Z"}], "name": "wordpress", "namespace": "wp",
"resourceVersion": "382706", "uid": "2ald9cf4-cd4c-466c-8aa0-b94ae245f4dc"},
"spec": {"progressDeadlineSeconds": 600, "replicas": 1,
"revisionHistoryLimit": 10, "selector": {"matchLabels": {"app": "wordpress",
"tier": "frontend"}}, "strategy": {"type": "Recreate"}, "template":
{"metadata": {"creationTimestamp": null, "labels": {"app": "wordpress",
"tier": "frontend"}}, "spec": {"containers": [{"env": [{"name":
"WORDPRESS_DB_HOST", "value": "wordpress-mysql"}, {"name":
"WORDPRESS_DB_PASSWORD", "value": "pepe"}, {"name": "WORDPRESS_DB_USER",
"value": "wordpress"}], "image": "acrunir.azurecr.io/wordpress:6.2.1-
apache", "imagePullPolicy": "IfNotPresent", "name": "wordpress", "ports":
[{"containerPort": 80, "name": "wordpress", "protocol": "TCP"}]},
"resources": {}, "terminationMessagePath": "/dev/termination-log",
"terminationMessagePolicy": "File", "volumeMounts": [{"mountPath":
"/var/www/html", "name": "wordpress-persistent-storage"}]}, "dnsPolicy":
"ClusterFirst", "imagePullSecrets": [{"name": "registry-credentials"}],
"restartPolicy": "Always", "schedulerName": "default-scheduler",
"securityContext": {}, "terminationGracePeriodSeconds": 30, "volumes":
[{"name": "wordpress-persistent-storage", "persistentVolumeClaim":
{"claimName": "wp-pv-claim"}}]}], "status": {"conditions":
[{"lastTransitionTime": "2024-07-16T09:53:10Z", "lastUpdateTime": "2024-07-
16T09:53:21Z", "message": "ReplicaSet \"wordpress-7bbdcb465\" has
successfully progressed.", "reason": "NewReplicaSetAvailable", "status":
"True", "type": "Progressing"}, {"lastTransitionTime": "2024-07-
16T11:14:15Z", "lastUpdateTime": "2024-07-16T11:14:15Z", "message":
"Deployment does not have minimum availability.", "reason":
"MinimumReplicasUnavailable", "status": "False", "type": "Available"}],
"observedGeneration": 2, "replicas": 1, "unavailableReplicas": 1,
"updatedReplicas": 1}}]}]}

```

```

local_file.k8s_config (local-exec): TASK [aks : Get wordpress service]

```

```

*****

```

```

local_file.k8s_config (local-exec): ok: [localhost] => {"api_found": true,
"changed": false, "resources": [{"apiVersion": "v1", "kind": "Service",
"metadata": {"creationTimestamp": "2024-07-16T08:34:52Z", "finalizers":
["service.kubernetes.io/load-balancer-cleanup"], "labels": {"app":
"wordpress"}, "managedFields": [{"apiVersion": "v1", "fieldsType":
"FieldsV1", "fieldsV1": {"f:metadata": {"f:labels": {".": {}, "f:app": {}},
"f:spec": {"f:allocateLoadBalancerNodePorts": {}, "f:externalTrafficPolicy":
{}, "f:internalTrafficPolicy": {}, "f:ports": {".": {}}, "k:
{"port": 80, "protocol": "TCP"}": {".": {}, "f:port": {}, "f:protocol":

```

```
{}, "f:targetPort": {}}, "f:selector": {}, "f:sessionAffinity": {}, "f:type": {}}, "manager": "OpenAPI-Generator", "operation": "Update", "time": "2024-07-16T08:34:52Z"}, {"apiVersion": "v1", "fieldsType": "FieldsV1", "fieldsV1": {"f:metadata": {"f:finalizers": {".": {}, "v:\\"service.kubernetes.io/load-balancer-cleanup\\"": {}}, "f:status": {"f:loadBalancer": {"f:ingress": {}}, "manager": "cloud-controller-manager", "operation": "Update", "subresource": "status", "time": "2024-07-16T10:30:35Z"}}, "name": "wordpress", "namespace": "wp", "resourceVersion": "373226", "uid": "58727b9b-0af6-4db6-9056-0448a021a253"}, "spec": {"allocateLoadBalancerNodePorts": true, "clusterIP": "10.0.159.20", "clusterIPs": ["10.0.159.20"], "externalTrafficPolicy": "Cluster", "internalTrafficPolicy": "Cluster", "ipFamilies": ["IPv4"], "ipFamilyPolicy": "SingleStack", "ports": [{"nodePort": 32083, "port": 80, "protocol": "TCP", "targetPort": 80}], "selector": {"app": "wordpress", "tier": "frontend"}, "sessionAffinity": "None", "type": "LoadBalancer"}, "status": {"loadBalancer": {"ingress": [{"ip": "135.236.178.212"}]}}, {"apiVersion": "v1", "kind": "Service", "metadata": {"creationTimestamp": "2024-07-16T06:16:55Z", "labels": {"app": "wordpress"}, "managedFields": [{"apiVersion": "v1", "fieldsType": "FieldsV1", "fieldsV1": {"f:metadata": {"f:labels": {".": {}, "f:app": {}}, "f:spec": {"f:clusterIP": {}, "f:internalTrafficPolicy": {}, "f:ports": {".": {}, "k:\\"port\\":3306,\\"protocol\\":\\"TCP\\"": {".": {}, "f:port": {}, "f:protocol": {}, "f:targetPort": {}}, "f:selector": {}, "f:sessionAffinity": {}, "f:type": {}}, "manager": "OpenAPI-Generator", "operation": "Update", "time": "2024-07-16T06:16:55Z"}}, "name": "wordpress-mysql", "namespace": "wp", "resourceVersion": "318116", "uid": "ac439768-36e7-4f83-8509-8e1724c68d77"}, "spec": {"clusterIP": "None", "clusterIPs": ["None"], "internalTrafficPolicy": "Cluster", "ipFamilies": ["IPv4"], "ipFamilyPolicy": "SingleStack", "ports": [{"port": 3306, "protocol": "TCP", "targetPort": 3306}], "selector": {"app": "wordpress", "tier": "mysql"}, "sessionAffinity": "None", "type": "ClusterIP"}, "status": {"loadBalancer": {}}}}]}
```

```
local_file.k8s_config (local-exec): TASK [aks : Show wordpress service]
```

```
*****
```

```
local_file.k8s_config (local-exec): ok: [localhost] => {
```

```
local_file.k8s_config (local-exec):     "msg": {
```

```
local_file.k8s_config (local-exec):         "api_found": true,
```

```
local_file.k8s_config (local-exec):         "changed": false,
```

```
local_file.k8s_config (local-exec):         "failed": false,
```

```
local_file.k8s_config (local-exec):         "resources": [
```

```
local_file.k8s_config (local-exec):             {
```

```
local_file.k8s_config (local-exec):                 "apiVersion": "v1",
```

```
local_file.k8s_config (local-exec):                 "kind": "Service",
```

```

local_file.k8s_config (local-exec): "metadata": {
local_file.k8s_config (local-exec):     "creationTimestamp":
"2024-07-16T08:34:52Z",
local_file.k8s_config (local-exec):     "finalizers": [
local_file.k8s_config (local-exec):
"service.kubernetes.io/load-balancer-cleanup"
local_file.k8s_config (local-exec): ],
local_file.k8s_config (local-exec): "labels": {
local_file.k8s_config (local-exec):     "app": "wordpress"
local_file.k8s_config (local-exec): },
local_file.k8s_config (local-exec): "managedFields": [
local_file.k8s_config (local-exec):     {
local_file.k8s_config (local-exec):         "apiVersion":
"v1",
local_file.k8s_config (local-exec):         "fieldsType":
"FieldsV1",
local_file.k8s_config (local-exec):         "fieldsV1": {
local_file.k8s_config (local-exec):             "f:metadata": {
local_file.k8s_config (local-exec):                 "f:labels": {
local_file.k8s_config (local-exec):                     ".": {},
local_file.k8s_config (local-exec):                 "f:app": {}
local_file.k8s_config (local-exec):             },
local_file.k8s_config (local-exec):             "f:spec":
{
local_file.k8s_config (local-exec):                 "f:allocateLoadBalancerNodePorts": {},
local_file.k8s_config (local-exec):                 "f:externalTrafficPolicy": {},
local_file.k8s_config (local-exec):                 "f:internalTrafficPolicy": {},
local_file.k8s_config (local-exec):                 "f:ports": {
local_file.k8s_config (local-exec):                     ".": {},
local_file.k8s_config (local-exec):                 "k:{\"port\":80,\"protocol\":\"TCP\"}": {
local_file.k8s_config (local-exec):                     ".": {},

```

```

local_file.k8s_config (local-exec):
"f:port": {},
local_file.k8s_config (local-exec):
"f:protocol": {},
local_file.k8s_config (local-exec):
"f:targetPort": {}
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
"f:selector": {},
local_file.k8s_config (local-exec):
"f:sessionAffinity": {},
local_file.k8s_config (local-exec):
"f:type": {}
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
"OpenAPI-Generator",
local_file.k8s_config (local-exec):
"Update",
local_file.k8s_config (local-exec):
"07-16T08:34:52Z"
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
"v1",
local_file.k8s_config (local-exec):
"FieldsV1",
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
"f:metadata": {
local_file.k8s_config (local-exec):
"f:finalizers": {
local_file.k8s_config (local-exec):
".": {},
local_file.k8s_config (local-exec):
"v:\\"service.kubernetes.io/load-balancer-cleanup\\"": {}
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
"f:status": {
local_file.k8s_config (local-exec):
"f:loadBalancer": {

```

```

local_file.k8s_config (local-exec):
"f:ingress": {}
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
"cloud-controller-manager",
local_file.k8s_config (local-exec):
"Update",
local_file.k8s_config (local-exec):
"status",
local_file.k8s_config (local-exec):
"time": "2024-
07-16T10:30:35Z"
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
"373226",
local_file.k8s_config (local-exec):
"4db6-9056-0448a021a253"
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
"allocateLoadBalancerNodePorts": true,
local_file.k8s_config (local-exec):
"10.0.159.20",
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
"Cluster",
local_file.k8s_config (local-exec):
"Cluster",
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
"SingleStack",
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
local_file.k8s_config (local-exec):
32083,

```

```

}
}
},
"manager":
"operation":
"subresource":
"uid": "58727b9b-0af6-
"},
"spec": {
"clusterIP":
"clusterIPs": [
"10.0.159.20"
],
"externalTrafficPolicy":
"internalTrafficPolicy":
"ipFamilies": [
"IPv4"
],
"ipFamilyPolicy":
"ports": [
{
"nodePort":

```

```

local_file.k8s_config (local-exec):                                     "port": 80,
local_file.k8s_config (local-exec):                                     "protocol":
"TCP",
local_file.k8s_config (local-exec):                                     "targetPort":
80
local_file.k8s_config (local-exec):                                     }
local_file.k8s_config (local-exec):                                     ],
local_file.k8s_config (local-exec):                                     "selector": {
local_file.k8s_config (local-exec):                                     "app":
"wordpress",
local_file.k8s_config (local-exec):                                     "tier": "frontend"
local_file.k8s_config (local-exec):                                     },
local_file.k8s_config (local-exec):                                     "sessionAffinity":
"None",
local_file.k8s_config (local-exec):                                     "type": "LoadBalancer"
local_file.k8s_config (local-exec):                                     },
local_file.k8s_config (local-exec):                                     "status": {
local_file.k8s_config (local-exec):                                     "loadBalancer": {
local_file.k8s_config (local-exec):                                     "ingress": [
local_file.k8s_config (local-exec):                                     {
local_file.k8s_config (local-exec):                                     "ip":
"135.236.178.212"
local_file.k8s_config (local-exec):                                     }
local_file.k8s_config (local-exec):                                     ]
local_file.k8s_config (local-exec):                                     }
local_file.k8s_config (local-exec):                                     },
local_file.k8s_config (local-exec):                                     {
local_file.k8s_config (local-exec):                                     "apiVersion": "v1",
local_file.k8s_config (local-exec):                                     "kind": "Service",
local_file.k8s_config (local-exec):                                     "metadata": {
local_file.k8s_config (local-exec):                                     "creationTimestamp":
"2024-07-16T06:16:55Z",
local_file.k8s_config (local-exec):                                     "labels": {
local_file.k8s_config (local-exec):                                     "app": "wordpress"
local_file.k8s_config (local-exec):                                     },
local_file.k8s_config (local-exec):                                     "managedFields": [
local_file.k8s_config (local-exec):                                     {
local_file.k8s_config (local-exec):                                     "apiVersion":
"v1",
local_file.k8s_config (local-exec):                                     "fieldsType":
"FieldsV1",
local_file.k8s_config (local-exec):                                     "fieldsV1": {

```

```

local_file.k8s_config (local-exec):
"f:metadata": {
local_file.k8s_config (local-exec):
"f:labels": {
local_file.k8s_config (local-exec):
".": {},
local_file.k8s_config (local-exec):
"f:app": {}
local_file.k8s_config (local-exec):
},
local_file.k8s_config (local-exec):
"f:spec":
{
local_file.k8s_config (local-exec):
"f:clusterIP": {},
local_file.k8s_config (local-exec):
"f:internalTrafficPolicy": {},
local_file.k8s_config (local-exec):
"f:ports": {
local_file.k8s_config (local-exec):
".": {},
local_file.k8s_config (local-exec):
"k:{\"port\":3306,\"protocol\":\"TCP\"}": {
local_file.k8s_config (local-exec):
".": {},
local_file.k8s_config (local-exec):
"f:port": {},
local_file.k8s_config (local-exec):
"f:protocol": {},
local_file.k8s_config (local-exec):
"f:targetPort": {}
local_file.k8s_config (local-exec):
},
local_file.k8s_config (local-exec):
},
local_file.k8s_config (local-exec):
"f:selector": {},
local_file.k8s_config (local-exec):
"f:sessionAffinity": {},
local_file.k8s_config (local-exec):
"f:type": {}
local_file.k8s_config (local-exec):
},
local_file.k8s_config (local-exec):
},
local_file.k8s_config (local-exec):
"manager":
"OpenAPI-Generator",
local_file.k8s_config (local-exec):
"operation":

```

```

"Update",
local_file.k8s_config (local-exec):                                "time": "2024-
07-16T06:16:55Z"
local_file.k8s_config (local-exec):                                }
local_file.k8s_config (local-exec):                                ],
local_file.k8s_config (local-exec):                                "name": "wordpress-
mysql",
local_file.k8s_config (local-exec):                                "namespace": "wp",
local_file.k8s_config (local-exec):                                "resourceVersion":
"318116",
local_file.k8s_config (local-exec):                                "uid": "ac439768-36e7-
4f83-8509-8e1724c68d77"
local_file.k8s_config (local-exec):                                },
local_file.k8s_config (local-exec):                                "spec": {
local_file.k8s_config (local-exec):                                "clusterIP": "None",
local_file.k8s_config (local-exec):                                "clusterIPs": [
local_file.k8s_config (local-exec):                                "None"
local_file.k8s_config (local-exec):                                ],
local_file.k8s_config (local-exec):                                "internalTrafficPolicy":
"Cluster",
local_file.k8s_config (local-exec):                                "ipFamilies": [
local_file.k8s_config (local-exec):                                "IPv4"
local_file.k8s_config (local-exec):                                ],
local_file.k8s_config (local-exec):                                "ipFamilyPolicy":
"SingleStack",
local_file.k8s_config (local-exec):                                "ports": [
local_file.k8s_config (local-exec):                                {
local_file.k8s_config (local-exec):                                "port": 3306,
local_file.k8s_config (local-exec):                                "protocol":
"TCP",
local_file.k8s_config (local-exec):                                "targetPort":
3306
local_file.k8s_config (local-exec):                                }
local_file.k8s_config (local-exec):                                ],
local_file.k8s_config (local-exec):                                "selector": {
local_file.k8s_config (local-exec):                                "app":
"wordpress",
local_file.k8s_config (local-exec):                                "tier": "mysql"
local_file.k8s_config (local-exec):                                },
local_file.k8s_config (local-exec):                                "sessionAffinity":
"None",
local_file.k8s_config (local-exec):                                "type": "ClusterIP"
local_file.k8s_config (local-exec):                                },

```



```

local_file.k8s_config (local-exec):           "status": {
local_file.k8s_config (local-exec):           "loadBalancer": {}
local_file.k8s_config (local-exec):           }
local_file.k8s_config (local-exec):           }
local_file.k8s_config (local-exec):           ]
local_file.k8s_config (local-exec):           }
local_file.k8s_config (local-exec):           }

local_file.k8s_config (local-exec): PLAY RECAP
*****
local_file.k8s_config (local-exec): localhost           : ok=8
changed=0    unreachable=0    failed=0    skipped=0    rescued=0
ignored=0

local_file.k8s_config: Creation complete after 12s
[id=eed0e4f705bb1088635b29adaecb21b00b3a4c08]

Apply complete! Resources: 1 added, 0 changed, 1 destroyed.

```

## K8S

La aplicación que vamos desplegar para probar K8S es un Wordpress. En esta app probaremos

- Funcionamiento de general AKS
- Acceso público mediante un *Service* de tipo *LoadBalancer*
- Persistencia del *Deployment* vía *PersistentVolume*

## Imágenes

Las imágenes están presentes ya en nuestro *Registry*, al no haber un rol específico para hacer *mirror* de *registry* se ha preferido hacerlo así

```

skopeo copy --dest-username acrunir --dest-password "XXXX"
docker://docker.io/mysql:8.0 docker://acrunir.azurecr.io/mysql:8.0
Getting image source signatures
Copying blob d9a40b27c30f done |
Copying blob ff7ba837a054 done |
Copying blob 36aa6d82134c done |
Copying blob 8c7ce6c9ed47 done |
Copying blob c3e750b66554 done |
Copying blob 07b15d10f125 done |
Copying blob 6bf60bd2cde3 done |
Copying blob 2434679df156 done |
Copying blob 4914dbc1b9fc done |
Copying blob a24e196bc810 done |

```

```
Copying blob 068a7d33dd38 done |
Copying config 6c54cbcf77 done |
Writing manifest to image destination
```

## Secretos

Hay que acordarse de añadir la propiedad *imagePullSecrets* a nuestro deployment sino los contenedores de los *Pods* no se descargarán

```
vents:
  Type          Reason          Age          From
Message
-----
-----
Normal        Scheduled              57m          default-scheduler
Successfully assigned wp/wordpress-555c954d89-4bc75 to aks-default-12959642-vmss000000
Normal        SuccessfulAttachVolume  57m          attachdetach-controller
AttachVolume.Attach succeeded for volume "pvc-568acbcc-d6d7-4c81-8a00-d6c9398377af"
Normal        Pulling              55m (x4 over 57m)    kubelet
Pulling image "acrunir.azurecr.io/wordpress:6.2.1-apache"
Warning       Failed              55m (x4 over 57m)    kubelet
Failed to pull image "acrunir.azurecr.io/wordpress:6.2.1-apache": failed to pull and unpack image "acrunir.azurecr.io/wordpress:6.2.1-apache": failed to resolve reference "acrunir.azurecr.io/wordpress:6.2.1-apache": failed to authorize: failed to fetch anonymous token: unexpected status from GET request to https://acrunir.azurecr.io/oauth2/token?scope=repository%3Awordpress%3Apull&service=acrunir.azurecr.io: 401 Unauthorized
Warning       Failed              55m (x4 over 57m)    kubelet
Error: ErrImagePull
Warning       Failed              55m (x6 over 57m)    kubelet
Error: ImagePullBackOff
Normal        BackOff              2m7s (x244 over 57m)  kubelet
Back-off pulling image "acrunir.azurecr.io/wordpress:6.2.1-apache"
```

Para ello hay que crear un secret en el *namespace* de *k8s* análogo a este

```
apiVersion: v1
kind: Secret
metadata:
  name: registry-credentials
  namespace: intelygenz-prod
type: kubernetes.io/dockerconfigjson
```

```
data:
  .dockerconfigjson: |
    BASE64DATA0000111122223333444455556666777788889999AAABBCCDDDEEFF
```

Una vez creado y subido

```
$ kubectl get all -n wp

NAME                                READY   STATUS    RESTARTS   AGE
pod/wordpress-7bbdcb465-5gj27      1/1     Running   0           116s
pod/wordpress-mysql-864bdc5c56-tqtpq 1/1     Running   0           117s

NAME                                TYPE                CLUSTER-IP      EXTERNAL-IP      PORT(S)
AGE
service/wordpress                  LoadBalancer        10.0.159.20     <pending>
80:32083/TCP      80m
service/wordpress-mysql            ClusterIP             None             <none>
3306/TCP          3h38m

NAME                                READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/wordpress          1/1     1             1           80m
deployment.apps/wordpress-mysql    1/1     1             1           3h37m

NAME                                DESIRED   CURRENT   READY   AGE
replicaset.apps/wordpress-555c954d89 0         0         0       80m
replicaset.apps/wordpress-7bbdcb465 1         1         1       116s
replicaset.apps/wordpress-mysql-7f6f45c6d9 0         0         0       4m22s
replicaset.apps/wordpress-mysql-864b5d575b 0         0         0       56m
replicaset.apps/wordpress-mysql-864bdc5c56 1         1         1       117s
replicaset.apps/wordpress-mysql-d74dd94bf 0         0         0       3h37m

$ kubectl describe pod/wordpress-7bbdcb465-5gj27 -n wp
Name:                wordpress-7bbdcb465-5gj27
Namespace:           wp
Priority:              0
Service Account:     default
Node:                aks-default-12959642-vmss000000/10.224.0.4
Start Time:          Tue, 16 Jul 2024 11:53:12 +0200
Labels:               app=wordpress
                     pod-template-hash=7bbdcb465
                     tier=frontend
```

```
Annotations:      <none>
Status:           Running
IP:               10.244.0.66
IPs:
  IP:             10.244.0.66
Controlled By:    ReplicaSet/wordpress-7bbdcb465
Containers:
  wordpress:
    Container ID:  containerd://155f53fcb48a02fc25adee0731eff56181aa618acb2a4b0b6811b0cd54067d59
    Image:         acrunir.azurecr.io/wordpress:6.2.1-apache
    Image ID:      acrunir.azurecr.io/wordpress@sha256:0d68783274943232adebe7f4c653fd7d141eebdb129b685371003dac999ea715
    Port:          80/TCP
    Host Port:     0/TCP
    State:         Running
      Started:     Tue, 16 Jul 2024 11:53:20 +0200
    Ready:         True
    Restart Count:  0
    Environment:
      WORDPRESS_DB_HOST:      wordpress-mysql
      WORDPRESS_DB_PASSWORD:  pepe
      WORDPRESS_DB_USER:     wordpress
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-jpwm2 (ro)
      /var/www/html from wordpress-persistent-storage (rw)
Conditions:
  Type            Status
  Initialized      True
  Ready           True
  ContainersReady True
  PodScheduled    True
Volumes:
  wordpress-persistent-storage:
    Type: PersistentVolumeClaim (a reference to a
PersistentVolumeClaim in the same namespace)
    ClaimName: wp-pv-claim
    ReadOnly:  false
  kube-api-access-jpwm2:
    Type: Projected (a volume that contains injected data
from multiple sources)
```

```
TokenExpirationSeconds: 3607
ConfigMapName: kube-root-ca.crt
ConfigMapOptional: <nil>
DownwardAPI: true
QoS Class: BestEffort
Node-Selectors: <none>
Tolerations: node.kubernetes.io/not-ready:NoExecute
op=Exists for 300s
node.kubernetes.io/unreachable:NoExecute
op=Exists for 300s
Events:
  Type      Reason      Age   From          Message
  ----      -
  Normal    Scheduled   3m6s  default-scheduler Successfully assigned
wp/wordpress-7bbdc465-5gj27 to aks-default-12959642-vmss000000
  Normal    Pulling     2m59s  kubelet        Pulling image
"acrunir.azurecr.io/wordpress:6.2.1-apache"
  Normal    Pulled      2m58s  kubelet        Successfully pulled image
"acrunir.azurecr.io/wordpress:6.2.1-apache" in 419ms (838ms including
waiting)
  Normal    Created     2m58s  kubelet        Created container wordpress
  Normal    Started     2m58s  kubelet        Started container wordpress
```

```
$ kubectl describe pod/wordpress-mysql-864bdc5c56-tqtpq -n wp
```

```
Name: wordpress-mysql-864bdc5c56-tqtpq
Namespace: wp
Priority: 0
Service Account: default
Node: aks-default-12959642-vmss000000/10.224.0.4
Start Time: Tue, 16 Jul 2024 11:53:11 +0200
Labels: app=wordpress
pod-template-hash=864bdc5c56
tier=mysql
Annotations: <none>
Status: Running
IP: 10.244.0.65
IPs:
  IP: 10.244.0.65
Controlled By: ReplicaSet/wordpress-mysql-864bdc5c56
Containers:
  mysql:
    Container ID:
```

```
containerd://e50fd65734e15dddc9fd500e50ae101b66b4340463ac60e1d0b8755a0c08340
6
  Image:          acrunir.azurecr.io/mysql:8.0
  Image ID:
acrunir.azurecr.io/mysql@sha256:5c8c4d6722f3cb66ab34ebb70a87c80f0d3dffc643241
80e416d90bd57e2878f
  Port:          3306/TCP
  Host Port:     0/TCP
  State:         Running
    Started:     Tue, 16 Jul 2024 11:53:20 +0200
  Ready:         True
  Restart Count: 0
  Environment:
    MYSQL_ROOT_PASSWORD: pepe
    MYSQL_DATABASE:      wordpress
    MYSQL_USER:          wordpress
    MYSQL_PASSWORD:      pepe
  Mounts:
    /var/lib/mysql from mysql-persistent-storage (rw)
    /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-
df7zv (ro)
Conditions:
  Type                Status
  Initialized         True
  Ready               True
  ContainersReady     True
  PodScheduled        True
Volumes:
  mysql-persistent-storage:
    Type:          PersistentVolumeClaim (a reference to a
PersistentVolumeClaim in the same namespace)
    ClaimName:     mysql-pv-claim
    ReadOnly:      false
  kube-api-access-df7zv:
    Type:          Projected (a volume that contains injected data
from multiple sources)
    TokenExpirationSeconds: 3607
    ConfigMapName:      kube-root-ca.crt
    ConfigMapOptional:  <nil>
    DownwardAPI:        true
QoS Class:           BestEffort
Node-Selectors:      <none>
Tolerations:         node.kubernetes.io/not-ready:NoExecute
```

```
op=Exists for 300s
```

```
node.kubernetes.io/unreachable:NoExecute
```

```
op=Exists for 300s
```

```
Events:
```

| Type   | Reason    | Age   | From              | Message  |
|--------|-----------|-------|-------------------|--|
| ----   | -----     | ----  | ----              | -----  |
| Normal | Scheduled | 4m4s  | default-scheduler | Successfully assigned wp/wordpress-mysql-864bdc5c56-tqtpq to aks-default-12959642-vmss000000 |
| Normal | Pulling   | 3m56s | kubelet           | Pulling image "acrunicr.azurecr.io/mysql:8.0"  |
| Normal | Pulled    | 3m56s | kubelet           | Successfully pulled image "acrunicr.azurecr.io/mysql:8.0" in 498ms (498ms including waiting) |
| Normal | Created   | 3m55s | kubelet           | Created container mysql  |
| Normal | Started   | 3m55s | kubelet           | Started container mysql  |

## Manifiestos

Los manifiestos de esta aplicación están separados en dos capas, *tiers*.

- *Frontend*
- *MySQL*

Para poder dinamizarlos se han convertido a una plantilla, vale que se podría hacer con HELM, pero ahora no me da la vida.

A continuación se muestran desglosados

### Frontend

Se crea un *service* de tipo *loadBalancer* escuchando en el puerto 80

```
apiVersion: v1
kind: Service
metadata:
  name: wordpress
  labels:
    app: wordpress
spec:
  ports:
    - port: 80
  selector:
    app: wordpress
    tier: frontend
  type: LoadBalancer
```

Se crea un *Claim* de 2GB para la capa front

```

---
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: wp-pv-claim
  labels:
    app: wordpress
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 2Gi
---

```

El service define las características de la aplicación *front* configura los volúmenes y los parámetros de la DB

```

apiVersion: apps/v1
kind: Deployment
metadata:
  name: wordpress
  labels:
    app: wordpress
spec:
  selector:
    matchLabels:
      app: wordpress
      tier: frontend
  strategy:
    type: Recreate
  template:
    metadata:
      labels:
        app: wordpress
        tier: frontend
    spec:
      containers:
        - image: {{ aks.wordpress.image }}
          name: wordpress
          env:
            - name: WORDPRESS_DB_HOST
              value: wordpress-mysql
            - name: WORDPRESS_DB_PASSWORD

```



```

      value: "{{ aks.mysql.password }}"
    - name: WORDPRESS_DB_USER
      value: wordpress
  ports:
    - containerPort: 80
      name: wordpress
  volumeMounts:
    - name: wordpress-persistent-storage
      mountPath: /var/www/html
  imagePullSecrets:
    - name: registry-credentials
  volumes:
    - name: wordpress-persistent-storage
      persistentVolumeClaim:
        claimName: wp-pv-claim

```

## Mysql

El Service creado en este caso es un ClusterIp, para comunicarse solo via local en el *namespace*

```

apiVersion: v1
kind: Service
metadata:
  name: wordpress-mysql
  labels:
    app: wordpress
spec:
  ports:
    - port: 3306
  selector:
    app: wordpress
    tier: mysql
  clusterIP: None

```

La capa de persistencia en este caso es mayor unos 20Gi

```

---
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: mysql-pv-claim
  labels:
    app: wordpress
spec:
  accessModes:

```

```
- ReadWriteOnce
resources:
  requests:
    storage: 20Gi
```

El despliegue de la MySQL, no presenta gran complejidad, solo se definen los volúmenes y las variables de entorno para configurar la mysql.

```
---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: wordpress-mysql
  labels:
    app: wordpress
spec:
  selector:
    matchLabels:
      app: wordpress
      tier: mysql
  strategy:
    type: Recreate
  template:
    metadata:
      labels:
        app: wordpress
        tier: mysql
    spec:
      containers:
        - image: "{{ aks.mysql.image }}"
          name: mysql
          env:
            - name: MYSQL_ROOT_PASSWORD
              value: "{{ aks.mysql.password }}"
            - name: MYSQL_DATABASE
              value: wordpress
            - name: MYSQL_USER
              value: wordpress
            - name: MYSQL_PASSWORD
              value: "{{ aks.mysql.password }}"
          ports:
            - containerPort: 3306
              name: mysql
          volumeMounts:
```

```

- name: mysql-persistent-storage
  mountPath: /var/lib/mysql
imagePullSecrets:
- name: registry-credentials
volumes:
- name: mysql-persistent-storage
  persistentVolumeClaim:
    claimName: mysql-pv-claim

```

## Problemas:

### Límite de ips públicas de la cuenta

La cuenta tiene una limitación de 3 ips públicas con lo cual nuestro *Service* no puede levantarse correctamente

```
$ kubectl describe service/wordpress -n wp
```

```

Name:                wordpress
Namespace:           wp
Labels:              app=wordpress
Annotations:         <none>
Selector:            app=wordpress,tier=frontend
Type:               LoadBalancer
IP Family Policy:    SingleStack
IP Families:         IPv4
IP:                 10.0.159.20
IPs:                10.0.159.20
Port:               <unset> 80/TCP
TargetPort:         80/TCP
NodePort:           <unset> 32083/TCP
Endpoints:          10.244.0.66:80
Session Affinity:    None
External Traffic Policy: Cluster

```

Events:

| Type    | Reason                 | Age                | From               |
|---------|------------------------|--------------------|--------------------|
| Warning | SyncLoadBalancerFailed | 30m (x19 over 96m) | service-controller |

Error syncing load balancer: failed to ensure load balancer: Retriable: **false**, RetryAfter: 0s, HTTPStatusCode: 400, RawError: {\r\n "error": {\r\n "code": "PublicIPCountLimitReached",\r\n "message": "Cannot create more than 3 public IP addresses for this subscription in this region.",\r\n

```

    "details": []\r
  }\r
}
Normal    EnsuringLoadBalancer    55s (x25 over 96m)    service-
controller    Ensuring load balancer
Warning   CreateOrUpdatePublicIPAddress    54s (x25 over 96m)    azure-cloud-
provider    Retriable: false, RetryAfter: 0s, HTTPStatusCode: 400, RawError:
{\r
  "error": {\r
    "code": "PublicIPCountLimitReached",\r
    "message": "Cannot create more than 3 public IP addresses for this
subscription in this region.",\r
    "details": []\r
  }\r
}

```

aksunir | Services and ingresses ☆ ...

Kubernetes service

Search « + Create Delete Refresh Show labels Give feedback

Overview  
Activity log  
Access control (IAM)  
Tags  
Diagnose and solve problems  
Microsoft Defender for Cloud  
Cost analysis (preview)

Kubernetes resources  
Namespaces  
Workloads  
Services and ingresses

Services Ingresses

Filter by service name Enter the full service name Filter by namespace All namespaces Add label filter

| <input type="checkbox"/> | Name            | Namespace   | Status    | Type         | Cluster IP  | External IP | Ports        | Age ↓   |     |
|--------------------------|-----------------|-------------|-----------|--------------|-------------|-------------|--------------|---------|-----|
| <input type="checkbox"/> | kube-dns        | kube-system | ✓ Ok      | ClusterIP    | 10.0.0.10   |             | 53/UDP53/TCP | 1 day   | ... |
| <input type="checkbox"/> | metrics-server  | kube-system | ✓ Ok      | ClusterIP    | 10.0.72.95  |             | 443/TCP      | 1 day   | ... |
| <input type="checkbox"/> | kubernetes      | default     | ✓ Ok      | ClusterIP    | 10.0.0.1    |             | 443/TCP      | 4 hours | ... |
| <input type="checkbox"/> | wordpress-mysql | wp          | ✓ Ok      | ClusterIP    | None        |             | 3306/TCP     | 4 hours | ... |
| <input type="checkbox"/> | wordpress       | wp          | ⚙ Pending | LoadBalancer | 10.0.159.20 |             | 80:32083/TCP | 2 hours | ... |

## Soluciones:

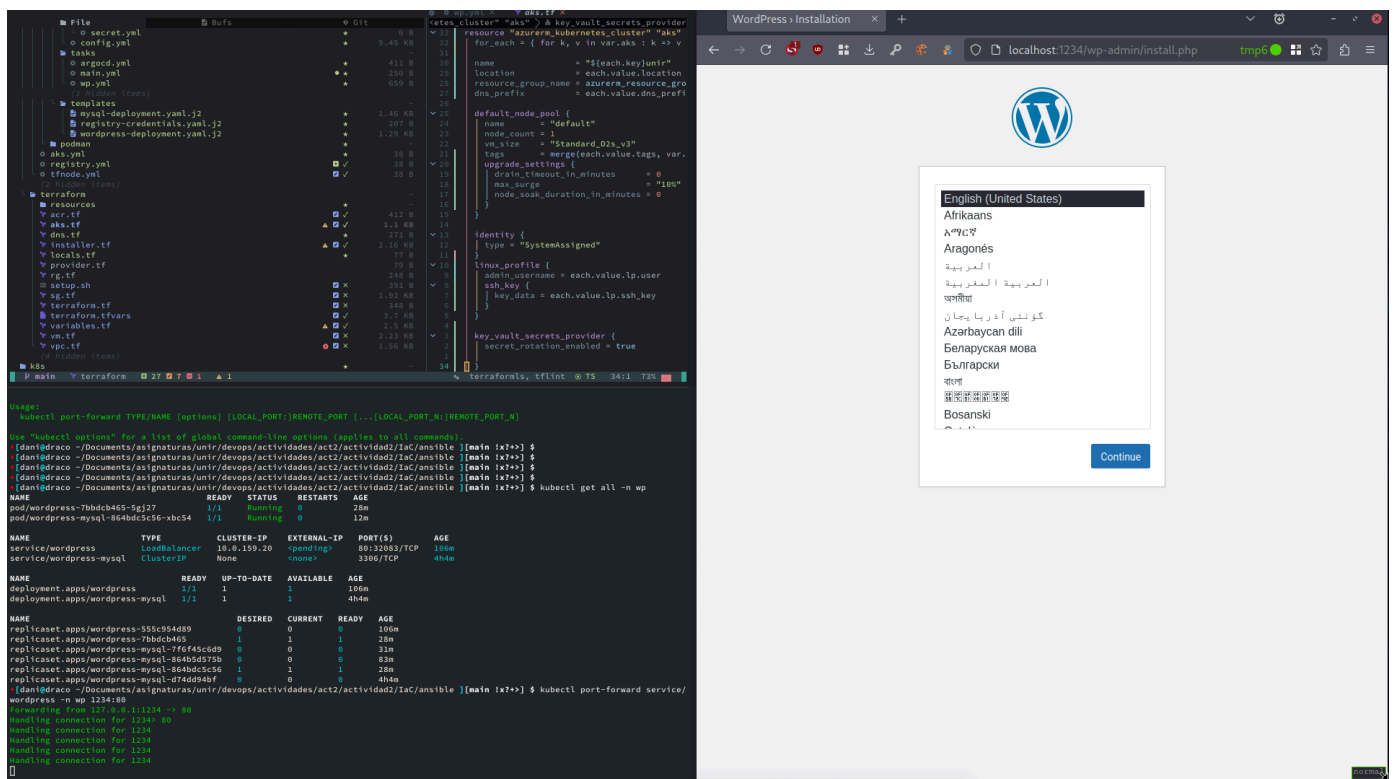
### Límite de ips públicas de la cuenta

Para hacer pruebas se puede hacer un port-forwarding desde nuestro local al nodo

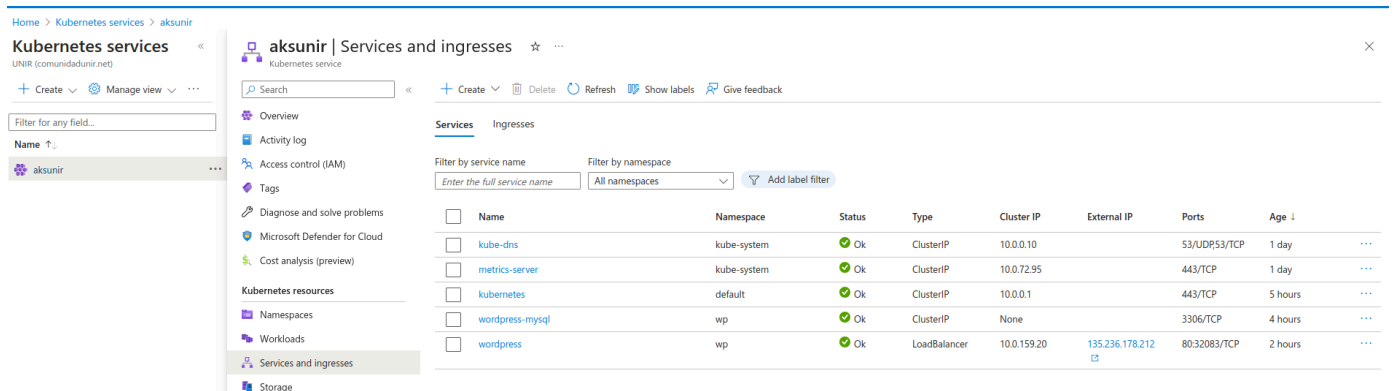
```

kubect1 port-forward service/wordpress -n wp 1234:80
Forwarding from 127.0.0.1:1234 -> 80
Handling connection for 1234> 80
Handling connection for 1234
Handling connection for 1234
Handling connection for 1234
Handling connection for 1234

```



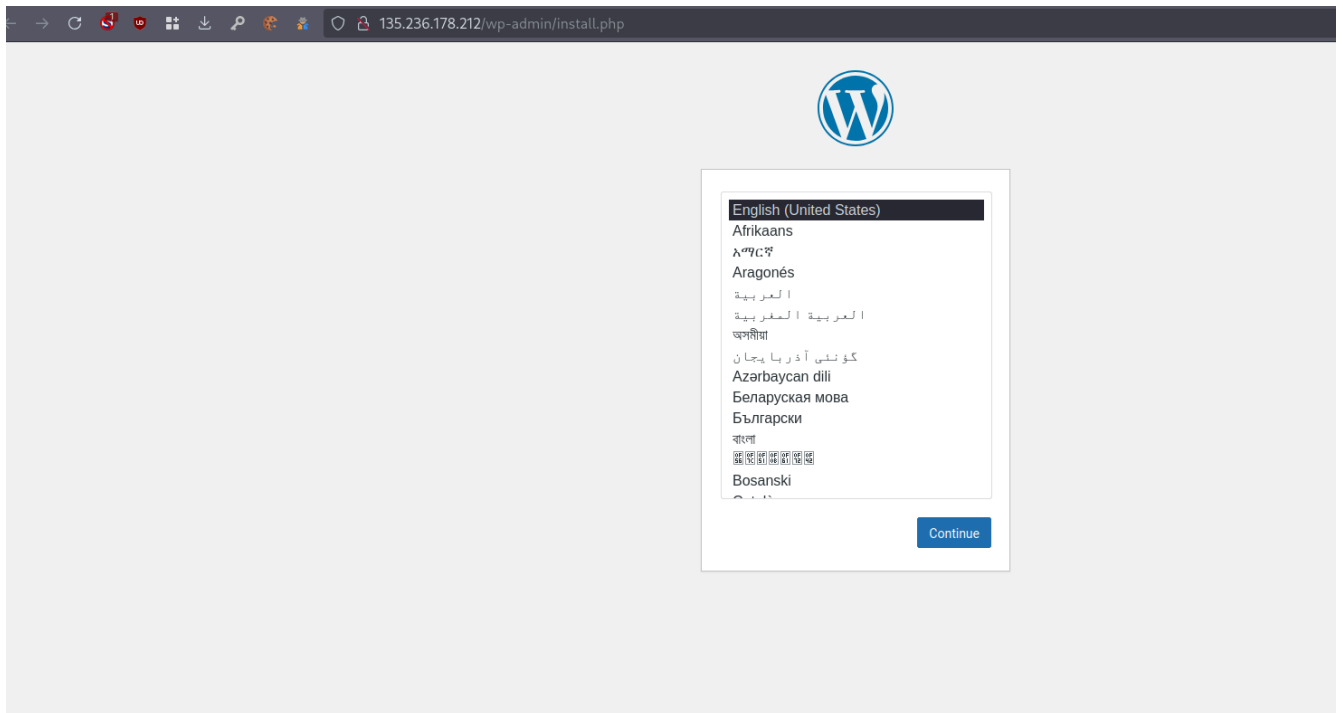
## O eliminar las ips sobrantes y redesplegar



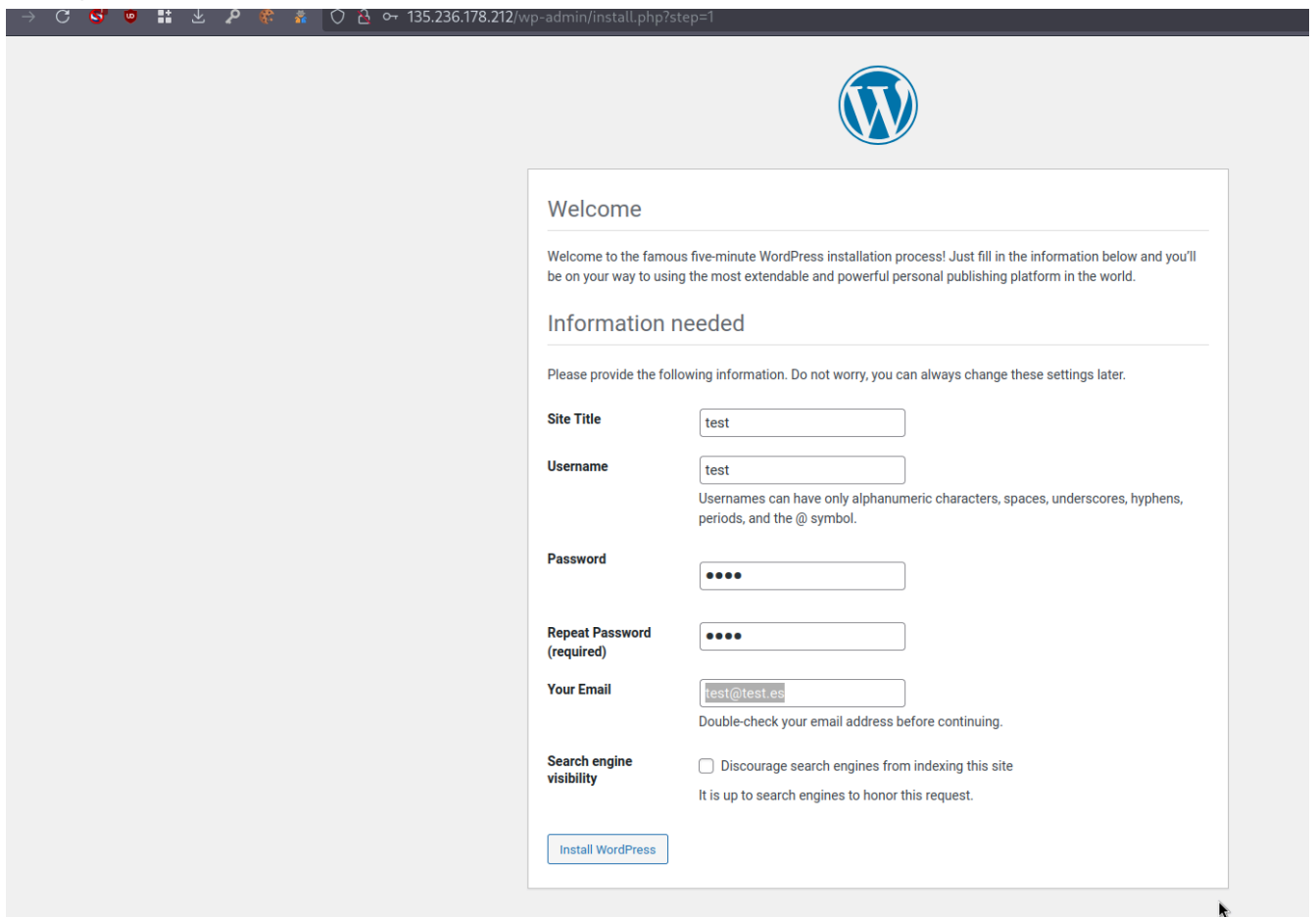
## User tests

Una vez solucionados los problemas con las ips, podemos acceder a nuestro *Wordpress* y probar la persistencia.

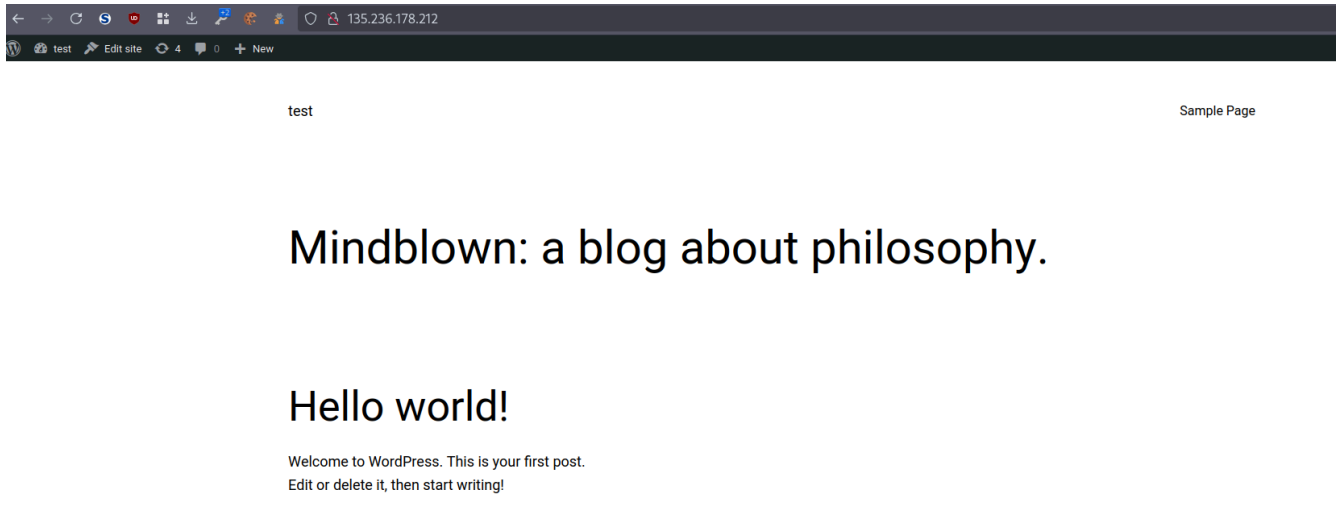
- Accedemos a nuestro service público



- Configuramos la instancia de WP

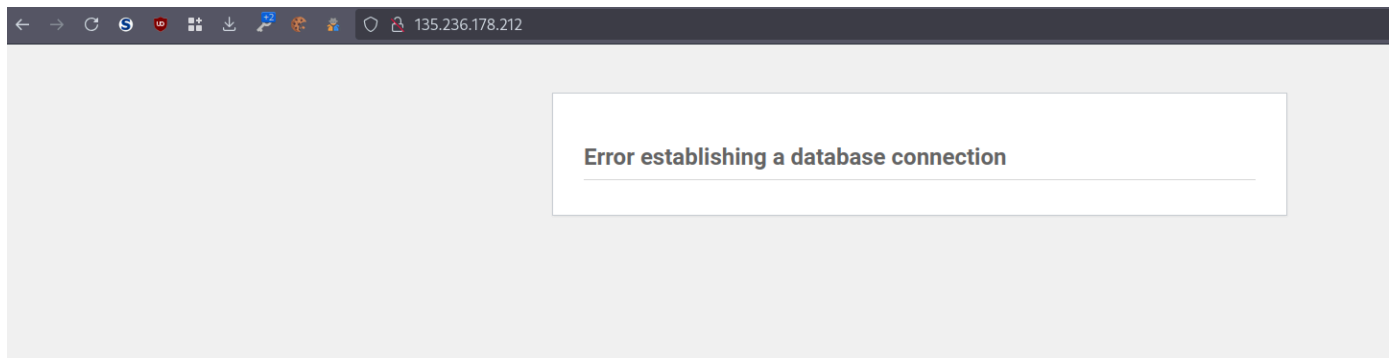


- Comprobamos que se crea un post de ejemplo.



- Eliminamos la base de datos.

```
$ kubectl delete deployment.apps/wordpress-mysql -n wp
deployment.apps "wordpress-mysql" deleted
```



- Redesplegamos

```
TASK [aks : Include tasks] *****
included: /home/dani/Documents/asignaturas/unir/devops/actividades/act2/actividad2/IaC/ansible/roles/aks/tasks/wp.yml for localhost

TASK [aks : Create namespace wp] *****
ok: [localhost]

TASK [aks : Create the deployments] *****
ok: [localhost] => (item=registry-credentials.yaml.j2)
changed: [localhost] => (item=mysql-deployment.yaml.j2)
ok: [localhost] => (item=wordpress-deployment.yaml.j2)

TASK [aks : Get wordpress service] *****
ok: [localhost]

TASK [aks : Show wordpress service] *****
ok: [localhost] => {
  "msg": {
    "api_found": true,
    "changed": false,
    "failed": false,
    "resources": f
```

Nuestro WP está en el mismo estado



test

Sam

# Mindblown: a blog about philosophy.

Hello world!

## Otras actividades no evaluables

Ya que se han hablado en el curso, vamos a probar un *pipeline* de despliegue continuo con *ArgoCD* y *Tekton*, aunque *tekton* no he podido probarlo.

### ArgoCD

#### Instalación

No es complicada, solo hay que crear el *namespace* y utilizar su manifiesto.

- Creamos el *namespace* e instalamos *ArgoCD*

```
$ kubectl create ns argocd
namespace/argocd created

$ kubectl apply -f https://raw.githubusercontent.com/argoproj/argo-
cd/v2.9.18/manifests/install.yaml -n argocd
customresourcedefinition.apiextensions.k8s.io/applications.argoproj.io
created
customresourcedefinition.apiextensions.k8s.io/applicationsets.argoproj.io
created
customresourcedefinition.apiextensions.k8s.io/appprojects.argoproj.io created
serviceaccount/argocd-application-controller created
serviceaccount/argocd-applicationset-controller created
serviceaccount/argocd-dex-server created
serviceaccount/argocd-notifications-controller created
serviceaccount/argocd-redis created
serviceaccount/argocd-repo-server created
serviceaccount/argocd-server created
role.rbac.authorization.k8s.io/argocd-application-controller created
role.rbac.authorization.k8s.io/argocd-applicationset-controller created
role.rbac.authorization.k8s.io/argocd-dex-server created
```



role.rbac.authorization.k8s.io/argocd-notifications-controller created  
role.rbac.authorization.k8s.io/argocd-redis created  
role.rbac.authorization.k8s.io/argocd-server created  
clusterrole.rbac.authorization.k8s.io/argocd-application-controller created  
clusterrole.rbac.authorization.k8s.io/argocd-server created  
rolebinding.rbac.authorization.k8s.io/argocd-application-controller created  
rolebinding.rbac.authorization.k8s.io/argocd-applicationset-controller created  
rolebinding.rbac.authorization.k8s.io/argocd-dex-server created  
rolebinding.rbac.authorization.k8s.io/argocd-notifications-controller created  
rolebinding.rbac.authorization.k8s.io/argocd-redis created  
rolebinding.rbac.authorization.k8s.io/argocd-server created  
clusterrolebinding.rbac.authorization.k8s.io/argocd-application-controller created  
clusterrolebinding.rbac.authorization.k8s.io/argocd-server created  
configmap/argocd-cm created  
configmap/argocd-cmd-params-cm created  
configmap/argocd-gpg-keys-cm created  
configmap/argocd-notifications-cm created  
configmap/argocd-rbac-cm created  
configmap/argocd-ssh-known-hosts-cm created  
configmap/argocd-tls-certs-cm created  
secret/argocd-notifications-secret created  
secret/argocd-secret created  
service/argocd-applicationset-controller created  
service/argocd-dex-server created  
service/argocd-metrics created  
service/argocd-notifications-controller-metrics created  
service/argocd-redis created  
service/argocd-repo-server created  
service/argocd-server created  
service/argocd-server-metrics created  
deployment.apps/argocd-applicationset-controller created  
deployment.apps/argocd-dex-server created  
deployment.apps/argocd-notifications-controller created  
deployment.apps/argocd-redis created  
deployment.apps/argocd-repo-server created  
deployment.apps/argocd-server created  
statefulset.apps/argocd-application-controller created  
networkpolicy.networking.k8s.io/argocd-application-controller-network-policy created  
networkpolicy.networking.k8s.io/argocd-applicationset-controller-network-policy created

```
networkpolicy.networking.k8s.io/argocd-dex-server-network-policy created
networkpolicy.networking.k8s.io/argocd-notifications-controller-network-
policy created
networkpolicy.networking.k8s.io/argocd-redis-network-policy created
networkpolicy.networking.k8s.io/argocd-repo-server-network-policy created
networkpolicy.networking.k8s.io/argocd-server-network-policy created
```

- Verificamos que la instalación de argocd funciona correctamente

```
$ kubectl get all -n argocd
```

| NAME   | READY | STATUS  |   |
|--|-------|---------|---|
| RESTARTS   | AGE   |         |   |
| pod/argocd-application-controller-0                  | 1/1   | Running | 0 |
| 48s  |       |         |   |
| pod/argocd-applicationset-controller-d648c7f59-jvsxk | 1/1   | Running | 0 |
| 49s  |       |         |   |
| pod/argocd-dex-server-698f479c5-vp7xw                | 1/1   | Running | 0 |
| 49s  |       |         |   |
| pod/argocd-notifications-controller-64574b4d56-ds6bn | 1/1   | Running | 0 |
| 49s  |       |         |   |
| pod/argocd-redis-6696fc96b8-4657h                    | 1/1   | Running | 0 |
| 49s  |       |         |   |
| pod/argocd-repo-server-6fd6bc85f9-tz5p5              | 1/1   | Running | 0 |
| 48s  |       |         |   |
| pod/argocd-server-9b6bb754-xbdmx                     | 1/1   | Running | 0 |
| 48s  |       |         |   |

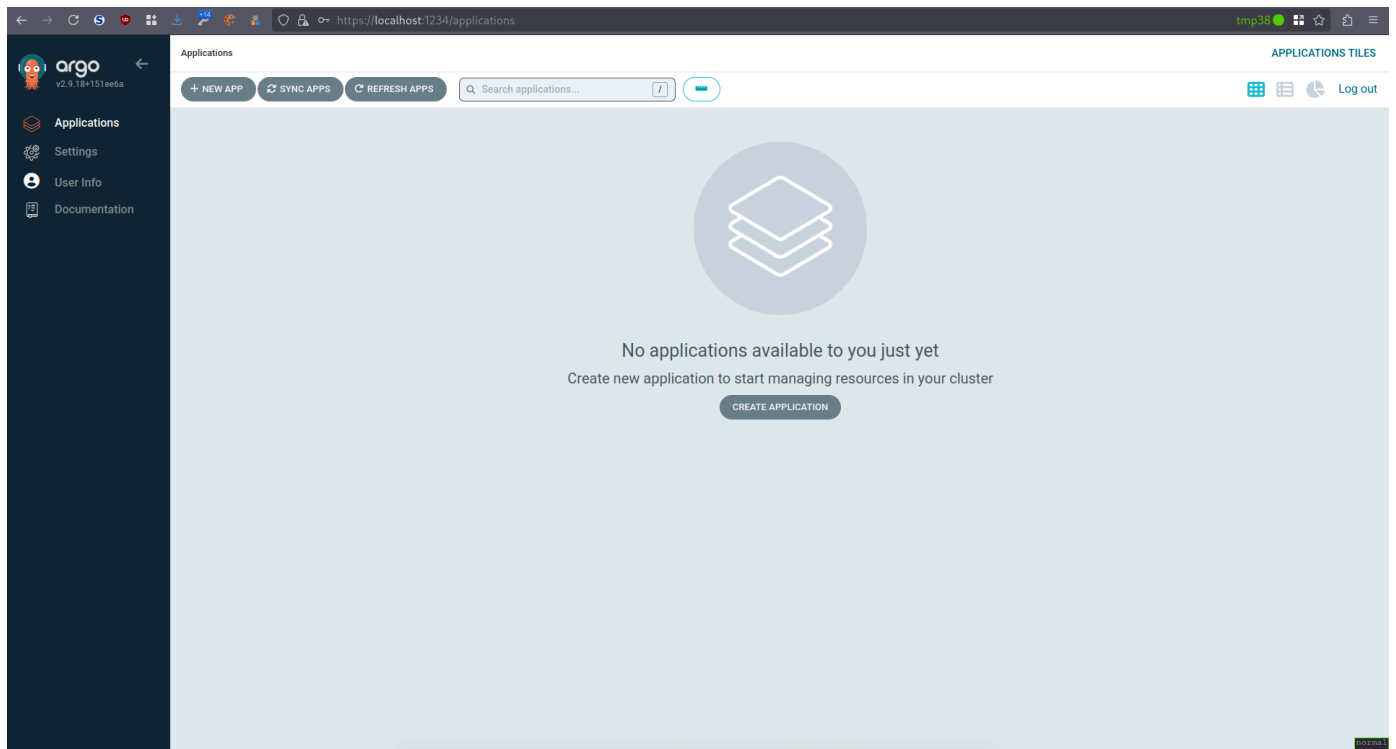
  

| NAME  | TYPE                         | CLUSTER-IP     |
|---|------------------------------|----------------|
| EXTERNAL-IP                                     | PORT(S)                      | AGE            |
| service/argocd-applicationset-controller        | ClusterIP                    | 10.110.180.199 |
| <none>  | 7000/TCP, 8080/TCP           | 49s            |
| service/argocd-dex-server                       | ClusterIP                    | 10.103.4.58    |
| <none>  | 5556/TCP, 5557/TCP, 5558/TCP | 49s            |
| service/argocd-metrics                          | ClusterIP                    | 10.102.74.41   |
| <none>  | 8082/TCP                     | 49s            |
| service/argocd-notifications-controller-metrics | ClusterIP                    | 10.105.3.141   |
| <none>  | 9001/TCP                     | 49s            |
| service/argocd-redis                            | ClusterIP                    | 10.97.212.216  |
| <none>  | 6379/TCP                     | 49s            |
| service/argocd-repo-server                      | ClusterIP                    | 10.103.96.176  |
| <none>  | 8081/TCP, 8084/TCP           | 49s            |
| service/argocd-server                           | ClusterIP                    | 10.108.55.161  |
| <none>  | 80/TCP, 443/TCP              | 49s            |
| service/argocd-server-metrics                   | ClusterIP                    | 10.103.228.122 |

|  |       |          |     |         |            |
|--|-------|----------|-----|---------|------------|
| <none>   |       | 8083/TCP |     | 49s     |            |
| NAME   |       |          |     | READY   | UP-TO-DATE |
| AVAILABLE  | AGE   |          |     |         |            |
| deployment.apps/argocd-applicationset-controller           |       |          | 1/1 | 1       | 1          |
| 49s  |       |          |     |         |            |
| deployment.apps/argocd-dex-server                          |       |          | 1/1 | 1       | 1          |
| 49s  |       |          |     |         |            |
| deployment.apps/argocd-notifications-controller            |       |          | 1/1 | 1       | 1          |
| 49s  |       |          |     |         |            |
| deployment.apps/argocd-redis                               |       |          | 1/1 | 1       | 1          |
| 49s  |       |          |     |         |            |
| deployment.apps/argocd-repo-server                         |       |          | 1/1 | 1       | 1          |
| 49s  |       |          |     |         |            |
| deployment.apps/argocd-server                              |       |          | 1/1 | 1       | 1          |
| 48s  |       |          |     |         |            |
| NAME   |       |          |     | DESIRED |            |
| CURRENT  | READY | AGE      |     |         |            |
| replicaset.apps/argocd-applicationset-controller-d648c7f59 |       |          |     | 1       | 1          |
| 1  | 49s   |          |     |         |            |
| replicaset.apps/argocd-dex-server-698f479c5                |       |          |     | 1       | 1          |
| 1  | 49s   |          |     |         |            |
| replicaset.apps/argocd-notifications-controller-64574b4d56 |       |          |     | 1       | 1          |
| 1  | 49s   |          |     |         |            |
| replicaset.apps/argocd-redis-6696fc96b8                    |       |          |     | 1       | 1          |
| 1  | 49s   |          |     |         |            |
| replicaset.apps/argocd-repo-server-6fd6bc85f9              |       |          |     | 1       | 1          |
| 1  | 48s   |          |     |         |            |
| replicaset.apps/argocd-server-9b6bb754                     |       |          |     | 1       | 1          |
| 1  | 48s   |          |     |         |            |
| NAME   |       |          |     | READY   | AGE        |
| statefulset.apps/argocd-application-controller             |       |          | 1/1 | 48s     |            |

- Establecemos el *port-forward* para acceder a la *wui*

```
kubectl port-forward -n argocd service/argocd-server 1234:443
Forwarding from 127.0.0.1:1234 -> 8080
Forwarding from [::1]:1234 -> 8080
```



## CICD circuito

Para la construcción del circuito de CICD usaremos una instancia de Gitlab. Con la siguiente *pipeline*. En esencia genera un contenedor con nuestro código utilizando un contenedor. Se hubiera podido utilizar *Kaniko*, pero por tiempo no se ha usado.

Para la instalación de la aplicación hacemos uso de kustomize para generar las nuevas imágenes automáticamente.

```
stages:
  - build
  - test
  - sec
  - publish
  - deploy

build-job:
  stage: build
  image: alpine:latest
  script:
    - echo "project directory $CI_PROJECT_DIR"
    - cat "$CI_PROJECT_DIR/main.py" ; echo
    - echo "$PWD"
  artifacts:
    paths:
      - $CI_PROJECT_DIR

unit-test:
```

```
stage: test
image: alpine:latest
script:
  - echo "unit test"
  - sleep 3
int-test:
stage: test
image: alpine:latest
script:
  - echo "integration test"
  - sleep 3

dependency-check:
stage: sec
image: alpine:latest
script:
  - echo "dependency check"
  - sleep 3

docker-build:
image: docker:latest
stage: publish
services:
  - docker:dind
before_script:
  - docker login $CI_DOCKER_REGISTRY -u $CI_REGISTRY_USER -p
$CI_REGISTRY_USER_PASS
script:
  - docker build -t
$CI_DOCKER_REGISTRY/$CI_DOCKER_REGISTRY_PATH:$CI_COMMIT_SHORT_SHA .
  - docker push
$CI_DOCKER_REGISTRY/$CI_DOCKER_REGISTRY_PATH:$CI_COMMIT_SHORT_SHA

deploy-prod:
stage: deploy
environment: production
image: alpine:latest
before_script:
  - apk add --no-cache git curl bash
  - curl -s "https://raw.githubusercontent.com/kubernetes-
sigs/kustomize/master/hack/install_kustomize.sh" | bash
  - mv kustomize /usr/local/bin/
```

```

- chmod +x /usr/local/bin/kustomize
- git remote set-url origin
https://oauth2:$TOKEN@gitlab.com/cicd8460412/unir.git
- git config --global user.email "services@paranoidworld.es"
- git config --global user.name "GitLab CI/CD"

script:
- git checkout -B main
- cd deployment/prod
- echo "kustomize edit set image
$CI_DOCKER_REGISTRY/$CI_DOCKER_REGISTRY_PATH:$CI_COMMIT_SHORT_SHA"
- kustomize edit set image
$CI_DOCKER_REGISTRY/$CI_DOCKER_REGISTRY_PATH:$CI_COMMIT_SHORT_SHA
- cat kustomization.yml
- git commit -am '[skip ci] PROD image update'
- git push origin main

only:
- main

```

- Verificamos los recursos del *namespace unir-prod*

```

$ kubectl get all -n unir-prod

```

| NAME                         | READY | STATUS  | RESTARTS | AGE   |
|------------------------------|-------|---------|----------|-------|
| pod/testapp-7fdd55cd76-fmnbh | 1/1   | Running | 0        | 7m12s |

| NAME         | TYPE     | CLUSTER-IP    | EXTERNAL-IP | PORT(S)        | AGE   |
|--------------|----------|---------------|-------------|----------------|-------|
| service/test | NodePort | 10.105.113.57 | <none>      | 8080:30477/TCP | 7m13s |

| NAME                    | READY | UP-TO-DATE | AVAILABLE | AGE |
|-------------------------|-------|------------|-----------|-----|
| deployment.apps/testapp | 1/1   | 1          | 1         | 50m |

| NAME                               | DESIRED | CURRENT | READY | AGE   |
|------------------------------------|---------|---------|-------|-------|
| replicaset.apps/testapp-7fdd55cd76 | 1       | 1       | 1     | 7m12s |
| replicaset.apps/testapp-8449c65bb6 | 0       | 0       | 0     | 50m   |

- Verificamos que la aplicación se despliega correctamente y sin errores.

argo

v2.9.18+151ee6a

Applications

Settings

User Info

Documentation

★ Favorites Only

SYNC STATUS

Unknown

0

Synced

0

OutOfSync

1

HEALTH STATUS

Unknown

0

Progressing

0

Suspended

0

Healthy

1

Degraded

0

Missing

0

LABELS

LABELS

PROJECTS

Applications

Search applications...

Log out

Sort: name Items per page: 10

unir

Project: default

Labels:

Status: Healthy OutOfSync

Repository: https://oauth2-gitpat-x9xyHFwV9Fpu5oGe...

Target Re... HEAD

Path: deployment/prod

Destinati... in-cluster

Namespa... unir-prod

Created At: 07/22/2024 15:57:46 (In a few seconds)

SYNC

REFRESH

DELETE

Applications / unir

APPLICATION DETAILS TREE

DETAILS

DIFF

SYNC

SYNC STATUS

HISTORY AND ROLLBACK

DELETE

REFRESH

APP HEALTH

Progressing

SYNC STATUS

Synced to HEAD (5cceb9c)

Auto sync is not enabled.

Author: GILab CI/CD <services@paranoidworld.es>

Comment: [skip ui] PROD Image update

LAST SYNC

Sync OK to 5cceb9c

Succeeded in a few seconds (Mon Jul 22 2024 15:58:48 GMT+0200)

Author: GILab CI/CD <services@paranoidworld.es>

Comment: [skip ui] PROD Image update

unir

unir-prod

test

testapp

test-vh9lh

testapp-7fd55cd76

testapp-7fd55cd76-fmnbh

testapp-8449c65bb6