



DOSSIER PROFESSIONNEL (DP)

Nom de naissance

► ZEMNI

Nom d'usage

► KASSAS

Prénom

► Sarra

Adresse

► 5 Rue de Bretagne 92140 Clamart

Titre professionnel visé

Administrateur système DevOps

MODALITE D'ACCES :

- Parcours de formation
- Validation des Acquis de l'Expérience (VAE)

DOSSIER PROFESSIONNEL (DP)

Présentation du dossier

Le dossier professionnel (DP) constitue un élément du système de validation du titre professionnel.
Ce titre est délivré par le Ministère chargé de l'emploi.

Le DP appartient au candidat. Il le conserve, l'actualise durant son parcours et le présente **obligatoirement à chaque session d'examen.**

Pour rédiger le DP, le candidat peut être aidé par un formateur ou par un accompagnateur VAE.

Il est consulté par le jury au moment de la session d'examen.

Pour prendre sa décision, le jury dispose :

1. des résultats de la mise en situation professionnelle complétés, éventuellement, du questionnaire professionnel ou de l'entretien professionnel ou de l'entretien technique ou du questionnement à partir de productions.
2. du **Dossier Professionnel** (DP) dans lequel le candidat a consigné les preuves de sa pratique professionnelle.
3. des résultats des évaluations passées en cours de formation lorsque le candidat évalué est issu d'un parcours de formation
4. de l'entretien final (dans le cadre de la session titre).

[Arrêté du 22 décembre 2015, relatif aux conditions de délivrance des titres professionnels du ministère chargé de l'Emploi]

Ce dossier comporte :

- ▶ pour chaque activité-type du titre visé, un à trois exemples de pratique professionnelle ;
- ▶ un tableau à renseigner si le candidat souhaite porter à la connaissance du jury la détention d'un titre, d'un diplôme, d'un certificat de qualification professionnelle (CQP) ou des attestations de formation ;
- ▶ une déclaration sur l'honneur à compléter et à signer ;
- ▶ des documents illustrant la pratique professionnelle du candidat (facultatif)
- ▶ des annexes, si nécessaire.

Pour compléter ce dossier, le candidat dispose d'un site web en accès libre sur le site.



<http://travail-emploi.gouv.fr/titres-professionnels>

DOSSIER PROFESSIONNEL (DP)

Sommaire

Exemples de pratique professionnelle

| | | |
|---|----|----|
| Automatiser le déploiement d'une infrastructure dans le cloud | p. | 5 |
| ▶ Automatiser la création de serveurs à l'aide de scripts..... | p. | 5 |
| ▶ Automatiser le déploiement d'une infrastructure | p. | 11 |
| ▶ Sécuriser l'infrastructure..... | p. | 17 |
| ▶ Mettre l'infrastructure en production dans le cloud | p. | 21 |
| Déployer en continu une application | p. | 31 |
| ▶ Préparer un environnement de test..... | p. | 31 |
| ▶ Gérer le stockage des données | p. | 40 |
| ▶ Gérer des containers | p. | 45 |
| ▶ Automatiser la mise en production d'une application avec une plateforme | p. | 49 |
| Superviser les services déployés | p. | 54 |
| ▶ Définir et mettre en place des statistiques de services | p. | 54 |
| ▶ Exploiter une solution de supervision..... | p. | 57 |
| ▶ Echanger sur des réseaux professionnels éventuellement en anglais | p. | 66 |
| Titres, diplômes, CQP, attestations de formation (facultatif) | p. | 67 |
| Déclaration sur l'honneur | p. | 68 |
| Documents illustrant la pratique professionnelle (facultatif) | p. | 69 |
| Annexes (Si le RC le prévoit) | p. | 70 |

EXEMPLES DE PRATIQUE PROFESSIONNELLE

DOSSIER PROFESSIONNEL (DP)

Activité-type 1

Automatiser le déploiement d'une infrastructure dans le cloud

Exemple n°1 ▶

Automatiser la création de serveurs à l'aide de scripts

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

- Pour la premier compétence en administration de système DevOps , j'ai utilisé mon compte de la plateforme cloud azure pour la création des serveur cloud .
- Et sur ma virtuel Machine virtuel linux j'ai fait un code en scripts Shell avec des commande de Azur CLI (command line interface) pour la création automatique des serveur des machines virtuelles Linux
- On premier , j'ai installer Azure CLI et je me suis connecter a mon compte de la plateforme cloud Azure

dans la machine virtuelle debian on installe azure :

```
*****  
curl -sL https://aka.ms/InstallAzureCLIDeb | sudo bash  
*****  
# connecter au compte azure  
az login -u <username> -p <password>
```

```
zemni@debian2:~$ az login -u sarra.kassas@labom2iforformation.fr -p [REDACTED]  
[  
 {  
   "cloudName": "AzureCloud",  
   "homeTenantId": "373016f8-79a9-4eed-80d2-100ce948d960",  
   "id": "e0cace2d-3d81-4c4a-8787-30d6fc5412fe",  
   "isDefault": true,  
   "managedByTenants": [],  
   "name": "Azure subscription 1",  
   "state": "Enabled",  
   "tenantId": "373016f8-79a9-4eed-80d2-100ce948d960",  
   "user": {  
     "name": "sarrah.kassas@labom2iforformation.fr",  
     "type": "user"  
   }  
 }  
 ]
```

- Ce script va demander au utilisateur si il veut créer une machine , et grâce a une boucle while et if va vérifier si il n'a pas taper : « y/Y » pour « oui » ou «n/N » pour « non »
- Une fois l'utilisateur à taper : « y/Y » pour « oui » ou «n/N » pour « non » , le scripte va afficher la liste des groupe et les afficher (si il y a pas des groupe donc il va rien afficher)
Grace a la commande : `az group list --output table`

DOSSIER PROFESSIONNEL (DP)

```
zemni@debian2: ~                                     zemni@debian2: ~
GNU nano 5.4                                         script.sh *
#!/bin/bash
echo " Est ce que voulez vous ajouter ou créer une machine ??"
echo " Taper (y/Y) si voulez vous créer "
echo "Ou bien (n/N) si vous ne voulez pas "
read boolMachine
while [[ $boolMachine != "n" && $boolMachine != "N" && $boolMachine != "y" && $boolMachine != "Y" ]];
do
    echo "Vous avez mal taper la commande , je repete ma question ?? Est ce que voulez vous ajouter ou créer une machine ??? "
    echo "Taper (y/Y) si voulez vous créer "
    echo "Ou bien (n/N) si vous ne voulez pas "
    read boolMachine
done
if [[ $boolMachine=="y" || $boolMachine=="Y" ]];then
    while [[ $boolMachine == 'y' || $boolMachine == 'Y' ]];
    do
        echo "Vous trouvez la liste des groupe qu'elle existe déjà : "
        az group list --output table
        echo " Enter the Resource Group name: "
        read ResourceGroupName
        echo " Enter the location (i.e. centralus):"
        echo " ( eastus=East US , eastus2=East US 2 ,"
        echo " southcentralus=South Central US , westus2=West US 2 ,"
        echo " westus3=West US 3 , australiaeast=Australia East ,"
        echo " swedencentral=Sweden Central , uksouth=UK South ,"
        echo " westeurope=West Europe , centralus=Central US ,"
        echo " southafricanorth=South Africa North , centralindia=Central India ,"
        echo " eastasia=East Asia , japaneast=Japan East ,"
        echo " koreacentral=Korea Central , canadacentral=Canada Central ,"
        echo " francecentral=France Central , germanywestcentral=Germany West Central ,"
        echo " norwayeast=Norway East , polandcentral=Poland Central ,"
        echo " brazilsouth=Brazil South , qatarcentral=Qatar Central ,"
        echo " centralusstage=USA Centre , northcentralusstage=USA Centre Nord ,"
        echo " southcentralusstage=USA Centre Sud , eastusstage=USA Est ,"
        echo " jioindiawest=Jio India West , westcentralus=West Central US ,"
        echo " southafricanwest=South Africa West , australiacentral=Australia Central ,"
        echo " australiacentral2=Australia Central 2 , australiasoutheast=Australia Southeast ,"

```

- Ensuite il va lire le nom de groupe , va lister la localisation de serveur et il va lire la localisation , une fois c'est fait , il va vérifier si le non de groupe de ressources il n'existe pas donc il va le crée : `az group create --name $ResourceGroupName --$location`
sinon : il va afficher un message « Ce groupe existe déjà » et il va afficher les information de groupe : `az group show $ResourceGroupName`

```
zemni@debian2: ~                                     zemni@debian2: ~
GNU nano 5.4                                         script.sh *
echo " southcentralusstage=USA Centre Sud , eastusstage=USA Est ,"
echo " jioindiawest=Jio India West , westcentralus=West Central US ,"
echo " southafricanwest=South Africa West , australiacentral=Australia Central ,"
echo " australiacentral2=Australia Central 2 , australiasoutheast=Australia Southeast ,"
echo " japanwest=Japan West , jioindiacentral=Jio India Central ,"
echo " koresouth=Korea South , southindia=South India ,"
echo " westindia=West India , canadaeast=Canada East ,"
echo " norwaywest=Norway West , switzerlandwest=Switzerland West ,"
echo " centraluseup , eastus2eup,"
echo " eastusstag , northcentralus,"
echo " eastasia , westus,"
echo " japaneast , switzerlandnorth,"
read location
if [[ $(az group exists --name $ResourceGroupName) == "false" ]]; then
    az group create --name $ResourceGroupName --location $location
    echo " Ce groupe est etais bien créer"
else
    az group show --name $ResourceGroupName
    echo "Ce group Existe "
fi
echo " Enter the virtual machine name (used for generating resource names): "
read NameVM
echo " Enter the administrator username: "
read username
echo ""
echo "Vous trouvez ici la liste des images de la machine virtuel : "
az vm image list --output table
echo "Enter le nom de L'image que vous voulez crée pour votre machine virtuel: "
read image
echo "Vous trouvez toute la liste des réseaux "
az network vnet list
echo " Vous trouvez la liste de réseau de Groupe $ResourceGroupName "
az network vnet list $ResourceGroupName
```

- Apres il va lire le nom de la machine virtuelle , l'utilisateur , réseau , sous-réseau , clé ssh , adresse-IP :

DOSSIER PROFESSIONNEL (DP)

```
zemni@debian2: ~                                         script.sh *
```

```
GNU nano 5.4
echo "Vous trouvez ici la liste des images de la machine virtuel : "
az vm image list --output table
echo "Enter le nom de L'image que vous voulez créer pour votre machine virtuel: "
read image
echo "Vous trouvez toute la liste des réseaux "
az network vnet list
echo " Vous trouvez la liste de réseau de Groupe $ResourceGroupName "
az network vnet list $ResourceGroupName
echo " Entrer le nom de Reseau "
read VnetName
echo " Entrer le nom de sous-Reseau "
read subVnetName
echo " Donner le nom de l'adresse IP fournie "
read NameIp
echo "Donner la zone de l'adresse IP 1 , 2 ou 3 "
read zoneIp
while [[ $zoneIp -ne 1 && $zoneIp -ne 2 && $zoneIp -ne 3 ]];
do
    echo " il faut taper la zone de l'adresse IP : 1 - 2 ou 3 "
    read zoneIp
done
az network public-ip create --name $NameIp --resource-group $ResourceGroupName --zone $zoneIp --allocation-method Static
echo " Enter le nom de la clé ssh "
read KeyName
az sshkey create --location $location --public-key ~/.ssh/id_rsa.pub --resource-group $ResourceGroupName --name $keyName
az network vnet create --name $VnetName --resource-group $ResourceGroupName --subnet-name $subVnetName --location $location
az vm create --name $NameVm --resource-group $ResourceGroupName --image $image --admin-username $username --size Standard_B1s --vnet-name $subVnetName --ssh-key-name $keyName --generate-ssh-keys
echo " Est ce que vouslez ajouter ou créer une autre machine ?? "
echo " Taper (y/Y) si vouslez créer "
echo " Ou bien (n/N) si vous ne voulez pas "
read boolMachine
done
else
exit 3
fi
```

- Vous pouvez trouver ci-joint un exemple d'exécution de script sur le terminal de la machine virtuelle :

- Si le groupe sources existe donc il va afficher un message que le groupe existe déjà :

DOSSIER PROFESSIONNEL (DP)

```
{  
  "id": "/subscriptions/e0cace2d-3d81-4c4a-8787-30d6fc5412fe/resourceGroups/zemnigroup",  
  "location": "francecentral",  
  "managedBy": null,  
  "name": "zemnigroup",  
  "properties": {  
    "provisioningState": "Succeeded"  
  },  
  "tags": null,  
  "type": "Microsoft.Resources/resourceGroups"  
}  
Ce group Existe  
Enter the virtual machine name (used for generating resource names):
```

- Il va lister les réseaux disponible pour proposer à l'utilisateur à choisir un ou créer un autre :

```
zemni@debian2:~  
wincomputeSP1      latest  
Enter le nom de L'image que vous voulez créer pour votre machine virtuel:  
Debian  
Vous trouvez toute la liste des réseaux  
[  
  {"  
    "addressSpace": {  
      "addressPrefixes": [  
        "10.0.0.0/16"  
      ]  
    },  
    "enabledSubscription": false,  
    "etag": "0x8C86F0EFCB84B9-62D6910D14FA\\\"",  
    "id": "/subscriptions/e0cace2d-3d81-4c4a-8787-30d6fc5412fe/resourceGroups/dev_group/providers/Microsoft.Network/virtualNetworks/dev_net",  
    "name": "dev net",  
    "provisioningState": "Succeeded",  
    "resourceGroup": "dev_group",  
    "resourceGuid": "77e5b1a3-59dd-4394-8ec2-151e7a00479b",  
    "subnets": [  
      {"  
        "addressPrefix": "10.0.0.0/24",  
        "delegations": [  
          "0x8C86F0EFCB84B9-62D6910D14FA\\\""  
        ],  
        "etag": "0x8C86F0EFCB84B9-62D6910D14FA\\\"",  
        "id": "/subscriptions/e0cace2d-3d81-4c4a-8787-30d6fc5412fe/resourceGroups/dev_group/providers/Microsoft.Network/virtualNetworks/dev_net/subnets/dev_subnet",  
        "name": "dev subnet",  
        "privateLinkServiceNetworkPolicies": "Disabled",  
        "privateEndpointNetworkPolicies": "Enabled",  
        "resourceGroup": "dev_group",  
        "type": "Microsoft.Network/virtualNetworks/subnets"  
      }  
    ],  
    "type": "Microsoft.Network/virtualNetworks",  
    "virtualNetworkPeerings": []  
  },  
  {"  
    "addressSpace": {  
      "addressPrefixes": [  
        "10.0.0.0/16"  
      ]  
    },  
    "enabledSubscription": false,  
    "etag": "0x8C86F0EFCB84B9-62D6910D14FA\\\"",  
    "id": "/subscriptions/e0cace2d-3d81-4c4a-8787-30d6fc5412fe/resourceGroups/dev_group/providers/Microsoft.Network/virtualNetworks/dev_net/subnets/dev_subnet",  
    "name": "dev subnet",  
    "provisioningState": "Succeeded",  
    "resourceGroup": "dev_group",  
    "resourceGuid": "77e5b1a3-59dd-4394-8ec2-151e7a00479b",  
    "subnets": [  
      {"  
        "addressPrefix": "10.0.0.0/24",  
        "delegations": [  
          "0x8C86F0EFCB84B9-62D6910D14FA\\\""  
        ],  
        "etag": "0x8C86F0EFCB84B9-62D6910D14FA\\\"",  
        "id": "/subscriptions/e0cace2d-3d81-4c4a-8787-30d6fc5412fe/resourceGroups/dev_group/providers/Microsoft.Network/virtualNetworks/dev_net/subnets/dev_subnet",  
        "name": "dev subnet",  
        "privateLinkServiceNetworkPolicies": "Disabled",  
        "privateEndpointNetworkPolicies": "Enabled",  
        "resourceGroup": "dev_group",  
        "type": "Microsoft.Network/virtualNetworks/subnets"  
      }  
    ],  
    "type": "Microsoft.Network/virtualNetworks",  
    "virtualNetworkPeerings": []  
  }]
```

- Vérifier le réseaux et le sous réseaux et ajouter une clé ssh :
- Vérifier la zone de l'adresse IP

```
zemni@debian2:~  
x64      WindowsServer      MicrosoftWindowsServer_2008-R2-SP1      MicrosoftWindowsServer:WindowsServer:2008-R2-SP1:latest  
Win2008R2SP1      latest  
Enter le nom de L'image que vous voulez créer pour votre machine virtuel:  
Debian  
Vous trouvez toute la liste des réseaux  
[]  
Vous trouvez la liste de réseau de Groupe group-dev  
unrecognized arguments: group-dev  
Examples from AI knowledge base:  
az network vnet list  
List all virtual networks in a subscription.  
az network vnet list --resource-group MyResourceGroup  
List all virtual networks in a resource group.  
https://docs.microsoft.com/en-US/cli/azure/network/vnet#az\_network\_vnet\_list  
Read more about the command in reference docs  
Enter le nom de Réseau  
net-dev  
Entrer le nom de sous-Réseau  
sebnet-dev  
c'est le sous réseau sebnet-dev de Groupe group-dev :  
(ResourceNotFound) The Resource 'Microsoft.Network/virtualNetworks/sebnet-dev' under resource group 'group-dev' was not found. For more details please go to https://aka.ms/ARMResourceNotFoundFix  
Code: ResourceNotFound  
Message: The Resource 'Microsoft.Network/virtualNetworks/sebnet-dev' under resource group 'group-dev' was not found. For more details please go to https://aka.ms/ARMResourceNotFoundFix  
Donner le nom de l'adresse IP fournie  
ip-dev  
Donner la zone de l'adresse IP 1 , 2 ou 3  
3  
Please note that the default public IP used for creation will be changed from Basic to Standard in the future.  
(ZonesNotAllowedInBasicSkuResource) Request for resource /subscriptions/e0cace2d-3d81-4c4a-8787-30d6fc5412fe/resourceGroups/group-dev/providers/Microsoft.Network/publicIPAddresses/ip-dev is invalid as it is a basic sku and in an Availability Zone which is a deprecated configuration. Please change your resource to Standard sku for Availability Zone support.  
Code: ZonesNotAllowedInBasicSkuResource  
Message: Request for resource /subscriptions/e0cace2d-3d81-4c4a-8787-30d6fc5412fe/resourceGroups/group-dev/providers/Microsoft.Network/publicIPAddresses/ip-dev is invalid as it is a basic sku and in an Availability Zone which is a deprecated configuration. Please change your resource to Standard sku for Availability Zone support.  
Enter le nom de la clé ssh  
ssh-dev  
| Running ...
```

- La machine « vm-dev » est bien créée :

DOSSIER PROFESSIONNEL (DP)

```
zemni@debian2: ~
"addressPrefixes": [
    "10.0.0.0/16"
],
"enableDdosProtection": false,
"etag": "W\"5669fbcd-4f91-4af4-b7f0-7f60890b2f5c\"",
"id": "/subscriptions/e0cace2d-3d81-4c4a-8787-30d6fc5412fe/resourceGroups/group-dev/providers/Microsoft.Network/virtualNetworks/net-dev",
"location": "westeurope",
"name": "net-dev",
"provisioningState": "Succeeded",
"resourceGroup": "group-dev",
"resourceGuid": "d9591d53-d53a-49be-bf7f-44c2a273787d",
"subnets": [
    {
        "addressPrefix": "10.0.0.0/24",
        "delegations": [],
        "etag": "W\"5669fbcd-4f91-4af4-b7f0-7f60890b2f5c\"",
        "id": "/subscriptions/e0cace2d-3d81-4c4a-8787-30d6fc5412fe/resourceGroups/group-dev/providers/Microsoft.Network/virtualNetworks/net-dev/subnets/sebnet-dev",
        "name": "sebnet-dev",
        "privateEndpointNetworkPolicies": "Disabled",
        "privateLinkServiceNetworkPolicies": "Enabled",
        "provisioningState": "Succeeded",
        "resourceGroup": "group-dev",
        "type": "Microsoft.Network/virtualNetworks/subnets"
    }
],
"type": "Microsoft.Network/virtualNetworks",
"virtualNetworkPeerings": []
}
}

Selecting "uksouth" may reduce your costs. The region you've selected may cost more for the same services. You can disable this message in the future with the command "az config set core.display_region_identified=false". Learn more at https://go.microsoft.com/fwlink/?linkid=222571
Ignite (November) 2023 onwards "az vm/vmss create" command will deploy Gen2-Trusted Launch VM by default. To know more about the default change and Trusted Launch, please visit https://aka.ms/TLaD
It is recommended to use parameter "--public-ip-sku Standard" to create new VM with Standard public IP. Please note that the default public IP used for VM creation will be changed from Basic to Standard in the future.
Consider using the image alias including the version of the distribution you want to use. For example: please use Debian11 instead of Debian.
In Ignite (November) 2023, the aliases without version suffix (such as: 'UbuntuLTs', 'CentOS', 'Debian', 'Flatcar', 'SLES', 'openSUSE-Leap' and 'RHEL') will be removed.
{"error":{"code":"DeploymentNotFound","message":"Deployment 'vm deploy_JWVcYmbgJycBsoudNRNBKPIrM7XI0vUK' could not be found."}}
Est ce que voulez vous ajouter ou créer une autre machine ??
Taper (y/Y) si voulez vous créer
Ou bien (n/N) si vous ne voulez pas
```

- Dans ce travaux pratique j'ai cree 4 machine pour les utiliser dans les prochaine compétence :

| Nom | Type | Abonnement | Groupe de ressources | Emplacement | État | Système d'exploitation | Taille | Adresse IP publique |
|---------|-------------------|----------------------|----------------------|----------------|----------------------|------------------------|--------------|---------------------|
| vm-dev | Machine virtuelle | Azure subscription 1 | group-dev | West Europe | En cours d'exécution | Linux | Standard_B1s | 13.69.51.151 |
| vm-ops | Machine virtuelle | Azure subscription 1 | group-ops | Canada Central | En cours d'exécution | Linux | Standard_B1s | 20.63.39.76 |
| vm-prod | Machine virtuelle | Azure subscription 1 | group-prod | West US | En cours d'exécution | Linux | Standard_B1s | 168.62.22.99 |
| vm-test | Machine virtuelle | Azure subscription 1 | group-test | East US | En cours d'exécution | Linux | Standard_B1s | 52.188.71.129 |

DOSSIER PROFESSIONNEL (DP)

```
zemni@debian2:~$ ssh -i ~/.ssh/id_rsa admin-dev@vm-dev
ssh: connect to host vm-dev port 22: Connection refused
zemni@debian2:~$ ssh -i ~/.ssh/id_rsa admin-dev@20.111.47.60
The authenticity of host '20.111.47.60 (20.111.47.60)' can't be established.
ECDSA key fingerprint is SHA256:00rgeYVCKt/LLZ295oVZNDqxYzA+MniiH/4m0lDwuS40.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '20.111.47.60' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 6.2.0-1011-azure x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Wed Sep 13 13:38:46 UTC 2023

System load:  0.0      Processes:          99
Usage of /:   5.1% of 28.89GB  Users logged in:  0
Memory usage: 31%           IPv4 address for eth0: 10.0.0.4
Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

admin-dev@vm-dev:~$
```

2. Précisez les moyens utilisés :

- Microsoft Azure (La plateforme de cloud Azure).
- Virtuel machine : Ubuntu
- Shell Linux : Bash (Bourne Again Shell).
- Un émulateur de terminal MobaXterm.

3. Avec qui avez-vous travaillé ?

- Travail individuel

4. Contexte

Nom de l'entreprise, organisme ou association ► *Cliquez ici pour taper du texte.*

Chantier, atelier, service ► *Cliquez ici pour taper du texte.*

Période d'exercice ► Du : **24/08/2023** au : **31/08/2023**

5. Informations complémentaires (*facultatif*)

DOSSIER PROFESSIONNEL (DP)

Activité-type 1

Automatiser le déploiement d'une infrastructure dans le cloud

Exemple n°2 ▶

Automatiser le déploiement d'une infrastructure

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

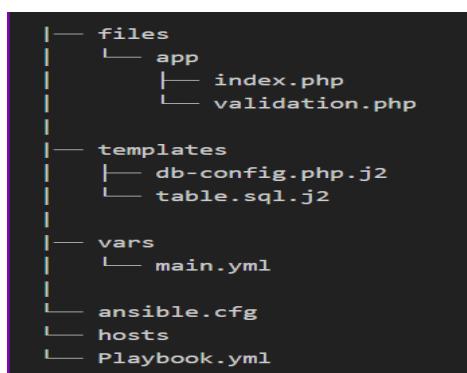
Après la création de 4 machine virtuelle sur le cloud de Microsoft Azure, nommé : « vm-ops (c'est une virtuel machine qui contient ansible et qu'elle va orchestrer les autre virtuel machine) , vm-dev et vm-test (c'est les machines de développeurs et qu'ils vont s'occuper de la partie web) , vm-prod (c'est la machine de la production de la base de données) .

J'ai créé un script Bash nommé install-ansible.sh sur la machine vm-ops pour l'installation d'ansible :

```
zemni@vm-ops:~$ ls  
install-ansible.sh  
zemni@vm-ops:~$ bash ./install-ansible.sh █
```

Ce Playbook Ansible nous fournira une alternative à l'exécution manuelle de la procédure d'installation générale d'un serveur LAMP (Linux, Apache, MySQL et PHP). L'exécution de ce Playbook automatisera donc les actions suivantes sur nos hôtes distants :

- Côté serveur web :
 - Installer les packages apache2, php et php-mysql
 - Déployer les sources de notre application dans notre serveur web distant
 - S'assurer que le service apache est bien démarré
- Côté serveur base de données :
 - Installer les packages mysql
 - Modifier le mot de passe root
 - Autoriser notre serveur web à communiquer avec la base de données
 - Configurer notre table mysql avec les bonnes colonnes et autorisations



- **Playbook.yml** : fichier Playbook contenant les tâches à exécuter sur le ou les serveurs distants.
- **vars/main.yml** : fichier pour nos variables afin de personnaliser les paramètres du Playbook (on peut aussi déclarer des variables dans le fichier Playbook).
- **hosts** : Fichier inventaire de notre Playbook.

DOSSIER PROFESSIONNEL (DP)

- `ansible.cfg` : par défaut ansible utilise le fichier de configuration `/etc/ansible/ansible.cfg` mais on peut surcharger la config en rajoutant un fichier nommé `ansible.cfg` à la racine du projet.
- `files/` : contient les sources de notre stack LAMP qui seront par la suite destinés à être traités par le module `copy`.
- `templates/` : contient des modèles de configurations dynamiques au format `jinja` qui sont destinés à être traités par le module `template`.

Le fichier inventaire

Pour ce projet, j'ai décidé de me séparer du fichier inventaire situé par défaut dans `/etc/ansible/hosts` et de **créer mon propre fichier `hosts`** à la racine du projet, où je me suis permis de séparer le serveur de base de données par rapport au serveur web, voici donc à quoi ressemble notre nouveau fichier inventaire :

```
GNU nano 6.2                                     hosts
[all]
vm-dev    ansible_host=192.168.163.147 ansible_user=zemni ansible_password=Reponse ansible_sudo_pass=Reponse ansible_sudo_password=Reponse
vm-prod   ansible_host=192.168.163.148 ansible_user=zemni ansible_password=Reponse ansible_sudo_pass=Reponse ansible_sudo_password=Reponse
[web]
vm-dev    ansible_host=192.168.163.147 ansible_user=zemni ansible_password=Reponse ansible_sudo_pass=Reponse ansible_sudo_password=Reponse
[db]
vm-prod   ansible_host=192.168.163.148 ansible_user=zemni ansible_password=Reponse ansible_sudo_pass=Reponse ansible_sudo_password=Reponse
[all:vars]
ansible_python_interpreter=/usr/bin/python3
```

Pour que notre nouveau fichier inventaire personnalisé soit pris en compte par votre Playbook, il faut au préalable modifier la valeur de la variable `inventory` située dans notre fichier de configuration ansible.

Par défaut ce fichier se situe dans le fichier `/etc/ansible/ansible.cfg`. Mais pour faire les choses dans les règles de l'art, nous allons laisser la configuration par défaut choisie par Ansible et créer notre propre fichier de configuration à la racine du projet. Dans notre nouveau fichier de config nous surchargerons uniquement la valeur de la variable `inventory`, ce qui nous donne le fichier `ansible.cfg` suivant :

```
GNU nano 6.2                                     ansible.cfg
[defaults]
inventory = ./hosts
```

Explication du Playbook

Pour commencer, voici déjà le contenu de notre Playbook :

```
---
# WEB SERVER

- hosts: web
become: true
vars_files: vars/main.yml
tasks:
- name: install apache and php last version
  apt:
    name:
```

DOSSIER PROFESSIONNEL (DP)

```
- apache2
- php
- php-mysql
state: present
update_cache: yes
- name: Give writable mode to http folder
file:
  path: /var/www/html
  state: directory
  mode: '0755'
- name: remove default index.html
file:
  path: /var/www/html/index.html
  state: absent
- name: upload web app source
copy:
  src: app/
  dest: /var/www/html/
- name: deploy php database config
template:
  src: "db-config.php.j2"
  dest: "/var/www/html/db-config.php"
- name: ensure apache service is start
service:
  name: apache2
  state: started
  enabled: yes
# DATABASE SERVER
- hosts: db
become: true
vars_files: vars/main.yml
vars:
  root_password: "my_secret_password"
tasks:
- name: install mysql
apt:
  name:
    - mysql-server
    - python-mysqldb # for mysql_db and mysql_user modules
  state: present
  update_cache: yes
- name: Create MySQL client config
copy:
  dest: "/root/.my.cnf"
  content: |
    [client]
    user=root
    password={{ root_password }}
  mode: 0400
- name: Allow external MySQL connexions (1/2)
lineinfile:
```

DOSSIER PROFESSIONNEL (DP)

```
path: /etc/mysql/mysql.conf.d/mysqld.cnf
regexp: '^skip-external-locking'
line: "# skip-external-locking"
notify: Restart mysql
- name: Allow external MySQL connexions (2/2)
  lineinfile:
    path: /etc/mysql/mysql.conf.d/mysqld.cnf
    regexp: '^bind-address'
    line: "# bind-address"
    notify: Restart mysql
- name: upload sql table config
  template:
    src: "table.sql.j2"
    dest: "/tmp/table.sql"
- name: add sql table to database
  mysql_db:
    name: "{{ mysql_dbname }}"
    state: present
    login_user: root
    login_password: '{{ root_password }}'
    state: import
    target: /tmp/table.sql
- name: "Create {{ mysql_user }} with all {{ mysql_dbname }} privileges"
  mysql_user:
    name: "{{ mysql_user }}"
    password: "{{ mysql_password }}"
    priv: "{{ mysql_dbname }}.*:ALL"
    host: "{{ webserver_host }}"
    state: present
    login_user: root
    login_password: '{{ root_password }}'
    login_unix_socket: /var/run/mysqld/mysqld.sock
handlers:
  - name: Restart mysql
    service:
      name: mysql
      state: restarted
```

Comme dit précédemment, nous avons choisi de séparer dans notre nouveau fichier inventaire le serveur de base de données par rapport à notre serveur web. Notre Playbook doit continuer dans cette voie en ciblant d'abord le serveur Web, puis le serveur de base de données (ou inversement).

Variables

Concernant les variables, vous avez le choix entre les placer directement depuis le mot-clé **vars**, ou vous pouvez les charger depuis un fichier en utilisant le mot-clé **vars_files** comme ceci :

```
vars_files: vars/main.yml
```

DOSSIER PROFESSIONNEL (DP)

```
GNU nano 6.2                               main.yml
---
mysql_user: "admin"
mysql_password: "secret"
mysql_dbname: "blog"
db_host: "192.168.163.148"
webserver_host: "192.168.163.147"
```

Voici le contenu du fichier de variables :

- **mysql_user** : l'utilisateur de notre base de données mysql qui exécutera nos requêtes SQL depuis notre application web.
- **mysql_password** : le mot de passe de l'utilisateur de notre base de données mysql.
- **mysql_dbname** : le nom de notre base de données.
- **db_host** : l'ip de notre machine mysql (utile pour la partie configuration mysql de notre application web).
- **webserver_host** : l'ip de la machine web (utile pour autoriser uniquement l'ip du serveur web à communiquer avec not

```
zemni@vm-ops:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/zemni/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/zemni/.ssh/id_rsa
Your public key has been saved in /home/zemni/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:zAhWjPbxeuSKGhhviPlz921Sg1p00Vx8LYyFGfdC0 zemni@vm-ops
The key's randomart image is:
+---[RSA 3072]----+
=..oo. o|
. + =+.BoEo.|_
. o . . +o.|
...+ + o
... o..So B
o o . . * o
o o . + o
..o + o
.o.o.o
+---[SHA256]----+
zemni@vm-ops:~$ cd .ssh
zemni@vm-ops:~/ssh$ ls
authorized_keys id_rsa id_rsa.pub
zemni@vm-ops:~/ssh$ ssh-copy-id zemni@192.168.163.147
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/zemni/.ssh/
id_rsa.pub"
The authenticity of host '192.168.163.147 (192.168.163.147)' can't be establishe
d.
ED25519 key fingerprint is SHA256:lRLI+aEBufIG4iynb845g5VojPL/DJdmYIMWmlo0fZs.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompt
ed now it is to install the new keys
zemni@192.168.163.147's password:

Number of key(s) added: 1

Now try logging into the machine, with:    "ssh 'zemni@192.168.163.147'"
```

Test :

Voici la commande pour lancer votre playbook . On visite ce URL <http://localhost>

```
zemni@vm-ops:~/projet$ ansible-playbook -i hosts playbook.yml
```

DOSSIER PROFESSIONNEL (DP)

```
zemni@vm-ops:~/projet1$ ansible-playbook -i hosts playbook.yml
PLAY [Deploy Infrastructure] ****
TASK [Gathering Facts] ****
ok: [vm-dev]
TASK [Update apt package cache] ****
changed: [vm-dev]
: TASK [Mise à jour des paquets] ****
changed: [vm-dev]
TASK [Installation de paquets de base] ****
ok: [vm-dev] => (item=nginx)
ok: [vm-dev] => (item=vim)
ok: [vm-dev] => (item=git)
TASK [Install required packages] ****
ok: [vm-dev] => (item=apache2)
ok: [vm-dev] => (item=mysql-client)
ok: [vm-dev] => (item=php)
ok: [vm-dev] => (item=python3)
ok: [vm-dev] => (item=python3-pip)
TASK [Configure Apache] ****
ok: [vm-dev]
PLAY [Installer MySQL sous Ubuntu] ****
TASK [Gathering Facts] ****
ok: [vm-prod]
TASK [Mise à jour des paquets] ****
changed: [vm-prod]
TASK [Installation de MySQL Server] ****
ok: [vm-prod]
TASK [Sécuriser l'installation MySQL] ****
```

2. Précisez les moyens utilisés :

- Microsoft Azure (La plateforme de cloud Azure).
- Virtuel Machine : Ubuntu
- Un émulateur de terminal MobaXterm.
- Ansible
- Apache – PHP - python
- Une base de données : Mysql

3. Avec qui avez-vous travaillé ?

- Travail individuel

4. Contexte

Nom de l'entreprise, organisme ou association ► *Cliquez ici pour taper du texte.*

Chantier, atelier, service ► *Cliquez ici pour taper du texte.*

Période d'exercice ► Du : *06/09/2023* au : *12/09/2023*

5. Informations complémentaires (*facultatif*)

DOSSIER PROFESSIONNEL (DP)

Activité-type 1 Automatiser le déploiement d'une infrastructure dans le cloud

Exemple n°3 ▶ Sécuriser l'infrastructure

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Sécuriser l'infrastructure avec Ansible implique plusieurs étapes pour garantir la protection des données, des systèmes et des communications. Pour sécuriser l'infrastructure à l'aide d'Ansible :

1. **Mises à jour de sécurité** : Assurer que tous les systèmes cibles sont à jour avec les derniers correctifs de sécurité. Utilisez le module apt d'Ansible pour automatiser les mises à jour.

```
- name : Mettre à jour les paquets  
become : yes  
apt :  
    upgrade : yes
```

2. **Configuration du pare-feu** : configurer le pare-feu sur les serveurs. Utiliser le module ufw, firewalld, ou tout autre module adapté à votre système d'exploitation.

```
---  
- name: Configurer le pare-feu avec UFW  
hosts: all  
become: yes  
tasks:  
  - name: Installer UFW  
    apt:  
      name: ufw  
      state: present  
  - name: Autoriser le trafic SSH  
    ufw:  
      rule: allow  
      port: 22  
      notify: Reload UFW  
  - name: Activer UFW  
    ufw:  
      state: enabled  
      notify: Reload UFW  
handlers:  
  - name: Reload UFW  
    ufw:  
      - state: reloaded
```

3. **Gestion des utilisateurs et des autorisations** : modifier ou supprimer des utilisateurs, ainsi que pour gérer leurs autorisations. Utilisez le module user d'Ansible pour cela.

```
- name: Créer un utilisateur  
user:  
  name: john  
  password: "$6$random_salt$hashed_password"  
  state: present
```

DOSSIER PROFESSIONNEL (DP)

4. **Gestion des clés SSH :** Utilisez Ansible pour distribuer les clés SSH et gérer l'accès des utilisateurs. Utilisez le module authorized_key pour cela. Gestion des clés d'accès, l'utilisation des clés SSH pour l'Authentification et la sécurisation de l'accès SSH :

- La sécurisation des accès ssh est essentielle pour protéger votre serveur contre des accès non autorisés. Utilisation de clés SSH :

Privilégiez l'authentification par clés ssh plutôt que par mot de passe.

Générez une paire de clés sur la machine avec la commande,

Générez une paire de clés sur chaque client : **ssh-keygen -t rsa** .

Après on tape la commande suivante :

Copier la clé publique vers le serveur : **ssh-copy-id -i ~/.ssh/id_rsa.pub**

Ou bien on copie la clé publique id_rsa.pub sur le serveur à l'emplacement
~/.ssh/authorized_keys pour chaque utilisateur autorisé.

```
- name: Set authorized key for remote user
  ansible.posix.authorized_key:
    user: "{{ created_username }}"
    state: present
    key: "{{ lookup('file', lookup('env','HOME') + '/.ssh/id_rsa.pub') }}"
```

5. **Sécurité des configurations système :** Automatisez la vérification et la configuration sécurisée des paramètres système, tels que les options de sécurité du noyau ou les configurations SSH. Utilisez le module lineinfile pour modifier des fichiers de configuration.

```
- name: Modifier la configuration du noyau
  become: yes
  lineinfile:
    path: /etc/sysctl.conf
    regexp: '^#?net\.ipv4\.icmp_echo_ignore_all'
    line: 'net.ipv4.icmp_echo_ignore_all = 1'
    notify: Reload sysctl
```

6. **Surveillance des journaux :** Utilisez Ansible pour configurer la surveillance des journaux sur les serveurs. Utiliser le module logrotate pour gérer la rotation des journaux.

```
- name: Configurer la rotation des journaux
  logrotate:
    name: myapp
    path: /var/log/myapp/*.log
```

7. **Chiffrement des données :** configurer le chiffrement des données au repos ou en transit, en utilisant par exemple le module openssl pour la gestion des certificats.

```
- name: Générer un certificat auto-signé
  openssl_certificate:
    path: /etc/ssl/myapp.crt
    privatekey_path: /etc/ssl/myapp.key
    common_name: myapp.example.com
```

8. **Tests de sécurité automatisés :** Intégrer des tests de sécurité automatisés dans le playbooks pour vérifier régulièrement la conformité de votre infrastructure.

DOSSIER PROFESSIONNEL (DP)

9. **Désactivation de l'accès Root** : Désactiver l'accès direct en tant que root via SSH. Utiliser plutôt des comptes d'utilisateurs normaux et utiliser « sudo » pour les priviléges d'administration. Et ajouter le sudo pour l'utilisateur de chaque machine en modifiant le fichier /etc/sudoers

En tapant dans le root la commande suivante : **Sudo visudo**

```
# et on ajoute cette ligne
root  ALL=(ALL:ALL) ALL
%admin  ALL=(ALL) ALL
%sudo  ALL=(ALL:ALL) ALL
```

```
- name: Setup passwordless sudo
  lineinfile:
    path: /etc/sudoers
    state: present
    dest: "/etc/sudoers.d/{{ new_user }}"
    regexp: '^%sudo'
    line: '%sudo ALL=(ALL) NOPASSWD: ALL'
    validate: '/usr/sbin/visudo -cf %s'
```

10. **Fail2Ban** : est installé pour détecter et prévenir les tentatives d'intrusion.

```
---
- name: Installer et configurer Fail2Ban
  hosts: all
  become: yes
  tasks:
    - name: Installer Fail2Ban
      apt:
        name: fail2ban
        state: present
      when: ansible_os_family == 'Ubuntu' # Pour les systèmes basés sur Debian/Ubuntu
    - name: Copier la configuration Fail2Ban
      template:
        src: templates/fail2ban/jail.conf.j2
        dest: /etc/fail2ban/jail.conf
      notify: Reload Fail2Ban

    - name: Activer le service Fail2Ban
      systemd:
        name: fail2ban
        state: started
        enabled: yes
    handlers:
      - name: Reload Fail2Ban
        systemd:
          name: fail2ban
          state: reloaded
```

11. **Créez le fichier de modèle pour la configuration de Fail2Ban** : Créez un répertoire templates/fail2ban et à l'intérieur, un fichier jail.conf.j2 avec le contenu suivant :

DOSSIER PROFESSIONNEL (DP)

```
[DEFAULT]
ignoreip = 127.0.0.1/8
bantime = 600
findtime = 600
maxretry = 3
[sshd]
enabled = true
```

12. Ajouter un nouvel utilisateur et groupe au système

Si l'utilisateur est déjà membre du groupe groupa et doit devenir membre de groupb:

`usermod -G groupa,groupb user-name`

Pour afficher les membres d'un groupe : `getent group group-name`

Comprenez comment configurer l'authentification classique avec le nom d'utilisateur et le mot de passe en veillant à utiliser des mots de passe forts.

Utiliser l'adduser pour ajouter un nouvel utilisateur à votre système : `adduser <nom_user>`

Vérifier un mot de passe pour l'utilisateur :

Utilisez la commande usermod pour ajouter l'utilisateur au groupe sudo :`usermod -aG sudo sammy`

2. Précisez les moyens utilisés :

- Microsoft Azure (La plateforme de cloud Azure).
- La machine Virtuel : Ubuntu
- L'émulateur de terminal MobaXterm.
- Installation des packages de sécurité
- Ansible

3. Avec qui avez-vous travaillé ?

- Travail Personnel

4. Contexte

Nom de l'entreprise, organisme ou association ► *Cliquez ici pour taper du texte.*

Chantier, atelier, service ► *Cliquez ici pour taper du texte.*

Période d'exercice ► Du : **25/09/2023** au : **29/09/2023**

5. Informations complémentaires (facultatif)

DOSSIER PROFESSIONNEL (DP)

Activité-type 1 Automatiser le déploiement d'une infrastructure dans le cloud

Exemple n°4 ▶ Mettre l'infrastructure en production dans le cloud

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Amazon Web Services (AWS) propose une large gamme de services que nous pouvons exploiter pour implémenter la haute disponibilité à toute application Web que nous déployons sur le Cloud AWS. Avant de commencer le déploiement de notre application, voyons voir déjà comment AWS organise son infrastructure à travers le monde depuis un schéma :



Module vpc : Voici les variables utilisées pour mon module vpc :

```
variable "vpc_cidr" {  
    default = "10.0.0.0/16"  
}  
variable "public_subnets_cidr" {  
    type  = list  
    default = ["10.0.1.0/24", "10.0.2.0/24"]  
}  
variable "private_subnets_cidr" {  
    type  = list  
    default = ["10.0.3.0/24", "10.0.4.0/24"]  
}  
variable "azs" {  
    type  = list  
    description = "AZs to use in your public and private subnet (make sure they are consistent with your AWS region)"  
    default  = ["us-west-2a", "us-west-2b"]  
}  
variable "prefix_name" {}
```

J'autorise ici l'utilisateur final le choix de choisir son propre bloc CIDR IPv4 pour son VPC, ses subnets privés et publics. Il peut également spécifier les zones de disponibilités à utiliser sur ces mêmes subnets. Vous verrez également tout au long du code de ce projet, la référence à la variable **prefix_name** qui comme son nom l'indique sera le préfixe à utiliser lors du nommage de nos différentes ressources AWS.

DOSSIER PROFESSIONNEL (DP)

Voici ensuite une partie de la configuration de notre module qui démontre comment j'ai **évité la duplication de code** grâce à l'utilisation du système de boucles :

```
...
...
# Public subnets
resource "aws_subnet" "main-public" {
  count      = length(var.public_subnets_cidr)
  vpc_id     = aws_vpc.main.id
  cidr_block = var.public_subnets_cidr[count.index]
  map_public_ip_on_launch = true
  availability_zone = var.azs[count.index]

  tags = {
    Name = "${var.prefix_name}-public-${count.index}"
  }
}

# Internet GW
resource "aws_internet_gateway" "main-gw" {
  vpc_id = aws_vpc.main.id

  tags = {
    Name = "${var.prefix_name}"
  }
}

# Public route tables
resource "aws_route_table" "main-public" {
  vpc_id = aws_vpc.main.id

  route {
    cidr_block = "0.0.0.0/0"
    gateway_id = aws_internet_gateway.main-gw.id
  }

  tags = {
    Name = "${var.prefix_name}-public"
  }
}

# Route associations public
resource "aws_route_table_association" "main-public" {
  count      = length(var.public_subnets_cidr)
  subnet_id  = aws_subnet.main-public[count.index].id
  route_table_id = aws_route_table.main-public.id
}
```

J'utilise le paramètre **count** pour boucler sur ma liste de blocs CIDR IPv4 de mes subnets publics en récupérant la longueur cette liste depuis la fonction **length(*ma_liste*)**, ensuite pour chaque itération je récupère l'élément de la

DOSSIER PROFESSIONNEL (DP)

liste `public_subnets_cidr` et `azs` depuis l'attribut `index` de mon paramètre `count`. J'utilise le même processus pour associer ma table de routage publique à mes subnets publics. Cette manipulation évite comme précisé plus haut, la duplication de code.

Passons maintenant aux variables de sortie de notre module vpc :

```
output "vpc_id" {
  value = aws_vpc.main.id
}

output "private_subnet_ids" {
  value = [for private_subnet in aws_subnet.main-private : private_subnet.id]
}

output "public_subnet_ids" {
  value = [for public_subnet in aws_subnet.main-public : public_subnet.id]
```

Je rajoute les attributs de mes ressources que je vais récupérer et utiliser sur mes autres modules, notamment l'id de mon vpc et ceux de mes subnets publics et privés. Autre astuce, J'utilise la boucle for pour créer une variable de type list des ids de mes subnets car c'est le type qui sera demandé plus tard dans mes autres modules de ressources.

Module s3

Voici les variables demandées du module S3 :

```
variable "bucket_name" {
  description = "bucket name must be globally unique"
}

variable "path_folder_content" {
  description = "add the contents of a folder to your S3 bucket (relatif or absolute path)"
}
```

D'un côté je demanderai à l'utilisateur de spécifier un nom de bucket qui doit être unique sur l'ensemble des noms de bucket mondiaux d'Amazon S3 et le chemin local des sources de l'application web qu'il souhaite envoyer dans son bucket S3.

Au jour d'aujourd'hui, il est compliqué depuis Terraform d'envoyer le contenu d'un dossier entier dans un bucket S3. J'ai dû donc trouver une astuce pour contourner ce problème. Pour ce faire, j'ai utilisé le provisionneur `local-exec` avec la commande `aws s3 sync` qui permet de copier et mettre à jour récursivement les nouveaux fichiers du répertoire source vers la destination, soit :

```
# S3 Bucket
resource "aws_s3_bucket" "my_bucket" {
  bucket = var.bucket_name
  acl   = "private"
  tags = {
    Name = var.bucket_name
  }
}

# Add content to bucket
resource "null_resource" "add_src_to_s3" {
  triggers = {
    build_number = "${timestamp()}" # run it all times
  }
  provisioner "local-exec" {
    command = "aws s3 sync ${var.path_folder_content} s3://${var.bucket_name}/"
```

DOSSIER PROFESSIONNEL (DP)

```
}
```

```
depends_on = [aws_s3_bucket.my_bucket]
```

```
}
```

Vous remarquerez, que j'ai rajouté l'argument **triggers** afin de forcer réexécution de l'ensemble du provisionneur en renvoyant la date et l'heure actuelles depuis la fonction **timestamp()** qui seront donc différents pour chaque nouvelle exécution de notre module de façon à s'assurer que notre bucket s3 contient toujours la dernière version de nos sources

Module ec2 role allow s3

Pour le bon fonctionnement de ce module j'ai besoin de récupérer le nom du bucket afin de créer la stratégie de mon rôle :

```
variable "bucket_name" {}
```

```
variable "prefix_name" {}
```

Ensuite dans ma configuration Terraform j'ai suivi la procédure exposée plus haut afin de créer mon profile d'instance que j'exporte dans mon fichier de variables de sortie **outputs.tf** :

```
output "name" {
  value = aws_iam_instance_profile.s3-mybucket-role-instanceprofile.name
}
```

Module alb_asg

Nous passons maintenant à la partie la plus complexe de notre projet, soit la création de notre ELB et ASG. On commence par récupérer l'id de notre VPC et de nos subnets ainsi que des informations basiques pour la création de notre ASG (ami, type d'instance, nombre min/max/désiré d'instances, profile d'instance, clé publique), le port et le protocole à utiliser dans notre groupe de sécurité et dans le Listener de notre ALB.

```
variable "prefix_name" {}
```

```
variable "vpc_id" {}
```

```
variable "private_subnet_ids" {
  type = list
}
```

```
variable "public_subnet_ids" {
  type = list
}
```

```
variable "webserver_port" {
  default = 80
}
```

```
variable "webserver_protocol" {
  default = "HTTP"
}
```

```
variable "user_data" {}
```

```
variable "instance_type" {
  default = "t2.micro"
}
```

```
variable "role_profile_name" {}
```

```
variable "min_instance" {
  description = "minimum number of instances for your ASG"
  default    = 2
}
```

```
variable "desired_instance" {
  description = "starting number of instances for your ASG"
  default    = 2
}
```

```
variable "max_instance" {
  description = "maximum number of instances for your ASG"
  default    = 4
}
```

```
variable "ami" {}
```

```
variable "path_to_public_key" {
```

DOSSIER PROFESSIONNEL (DP)

```
description = "relatif or absolute path of your public ssh key"
}
```

Pour la partie configuration de ce module, il était demandé d'autoriser uniquement le trafic provenant de notre ELB pour le groupe de sécurité de nos instances web. L'astuce pour satisfaire cette exigence, consiste non pas à utiliser l'adresse IP de notre ELB (c'est possible mais non recommandé) mais bel et bien le groupe de sécurité de notre ELB en tant que cible dans la règle ingress de notre groupe de sécurité de nos instances web. Ceci est possible depuis l'argument `security_groups` :

```
# Security group for ALB
```

```
resource "aws_security_group" "sg-alb" {
    vpc_id      = var.vpc_id
```

```
...
```

```
...
```

```
ingress {
```

```
    from_port  = var.webserver_port
    to_port    = var.webserver_port
    protocol   = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
```

```
}
```

```
...
```

```
...
```

```
# Security group for ASG instances
```

```
resource "aws_security_group" "sg-instances" {
```

```
    vpc_id      = var.vpc_id
```

```
...
```

```
...
```

```
ingress {
```

```
    from_port  = var.webserver_port
    to_port    = var.webserver_port
    protocol   = "tcp"
    security_groups = [aws_security_group.sg-alb.id]
```

```
}
```

```
...
```

```
...
```

Dans ma configuration de lancement de mon ASG j'utilise l'option `create_before_destroy`, cet indicateur est utilisé pour garantir que le remplacement d'une ressource est créé avant la destruction de l'instance d'origine :

```
# ASG launch configuration
```

```
resource "aws_launch_configuration" "my-launchconfig" {
```

```
...
```

```
...
```

```
lifecycle {
```

```
    create_before_destroy = true
```

```
}
```

```
}
```

Lors de la création de l'ASG, j'ai ajouté l'option `force_delete` qui permet de modifier le comportement par défaut de destruction de Terraform. Normalement, Terraform attend que tout votre pool instances EC2 soit terminé avant de supprimer votre ASG. Cependant, si vous valorisez cet attribut à `true`, cela forcera la suppression de votre ASG sans attendre la mise en fin de toutes les instances de votre ASG.

Ensuite, j'ai suivi la procédure énoncée plus haut pour notre ELB. Enfin, dans mes variables de sortie, j'exporte le nom DNS de mon ELB afin de pouvoir tester son bon fonctionnement directement depuis mon navigateur. J'exporte également le nom mon ASG

DOSSIER PROFESSIONNEL (DP)

qui sera utilisé plus tard dans nos alarmes CloudWatch, ainsi que l'id du groupe de sécurité de nos instances web afin de le rajouter en tant que cible dans la règle ingress du groupe de sécurité de notre base de données :

```
output "alb_dns_name" {
    value = aws_lb.my-alb.dns_name
}
output "webserver_sg_id" {
    value = [aws_security_group.sg-instances.id]
}
output "asg_name" {
    value = aws_autoscaling_group.my-autoscaling.name
}
```

Module cloudwatch_cpu_alarms

Pour notre dernier module, je propose à l'utilisateur final de choisir le seuil à atteindre sous forme de pourcentage pour déclencher notre alarme CloudWatch de scale up et scale down. Je lui demande également de renseigner le nom de l'ASG sur lequel sera déclencher le processus du scale up/down :

```
variable "prefix_name" {}

variable "max_cpu_percent_alarm" {
    description = "percentage of CPU consumption to achieve scale up"
    default     = 80
}

variable "min_cpu_percent_alarm" {
    description = "percentage of CPU consumption to achieve scale down"
    default     = 5
}

variable "asg_name" {
    description = "ASG name used to increase or decrease ec2 instances"
}
```

Dans la configuration de ce module, j'ai suivi à la lettre la procédure décrite plus haut pour nos alarmes CloudWatch. Cependant, je tiens à préciser que si jamais vous souhaitez être notifiés par mail avec le service SNS lors d'un dimensionnement dynamique de votre ASG. Sachez que Terraform vous donnera la possibilité de créer un topic dans le service SNS pour ce type de demande, mais malheureusement sur la version actuelle de cet article (v0.12.24), Terraform ne vous permet pas de renseigner une adresse mail d'abonnement à ce topic. C'est donc une opération qu'il faut effectuer depuis la console AWS en post création de votre infrastructure.

Néanmoins, je vous propose ci-dessous le code à utiliser pour créer votre topic SNS :

```
# Autoscaling notifications
resource "aws sns topic" "my-sns" {
    name      = "${var.prefix_name}-asg-sns"
    display_name = "my ASG SNS topic"
}

resource "aws autoscaling notification" "my-notify" {
    group_names = ["${var.asg_name}"]
    topic_arn   = "${aws sns topic.my-sns.arn}"
    notifications = [
        "autoscaling:EC2_INSTANCE_LAUNCH",
        "autoscaling:EC2_INSTANCE_TERMINATE",
        "autoscaling:EC2_INSTANCE_LAUNCH_ERROR"
    ]
}
```

DOSSIER PROFESSIONNEL (DP)

```
]
```

Module racine

Pour le module racine, voici à quoi va ressembler mon fichier de variables :

```
variable "region" {
  default = "us-west-2"
}

variable "prefix_name" {
  default = "devopssec"
}

variable "bucket_name" {
  default = "devopssec-bucket"
}

variable "db_password" {}

data "aws_ami" "ubuntu-ami" {
  most_recent = true
  filter {
    name  = "name"
    values = ["ubuntu/images/hvm-ssd/ubuntu-bionic-18.04-amd64-server-20200430"]
  }
  owners = ["099720109477"] # Canonical
}
```

Les variables **prefix_name** et **bucket_name** seront utilisées plusieurs fois en tant que module d'où leurs présences dans le fichier **variables.tf**. La variable **db_password** est une donnée secrète que je vais définir hors mon module racine d'où l'absence de la valeur par défaut. Enfin, grâce à ma Data Source **ubuntu-ami** je récupère automatiquement l'ami de la version 18.04 LTS bionic (amd64) de la région us-west-2 (plus d'informations sur le [chapitre consacré aux Data Sources](#)).

Dans le fichier de configuration Terraform de mon module racine, voici le user-data que j'utilise pour installer et configurer mon application web :

```
module "my_alb_asg" {
  source      = "./modules/alb_asg"
  ...
  ...
  user_data = <<<-EOF
#!/bin/bash
  sudo apt-get update -y
```

DOSSIER PROFESSIONNEL (DP)

```
sudo apt-get install -y apache2 awscli mysql-client php php-mysql
sudo systemctl start apache2
sudo systemctl enable apache2
sudo rm -f /var/www/html/index.html
sudo aws s3 sync s3://${var.bucket_name}/ /var/www/html/
mysql -h ${module.my_rds.host} -u ${module.my_rds.username} -p${var.db_password} </var/www/html/articles.sql
sudo sed -i 's/##DB_HOST##/${module.my_rds.host}' /var/www/html/db-config.php
sudo sed -i 's/##DB_USER##/${module.my_rds.username}' /var/www/html/db-config.php
sudo sed -i 's/##DB_PASSWORD##/${var.db_password}' /var/www/html/db-config.php
EOF
}
```

Dans ce user-data, les tâches suivantes seront exécutées :

- Mise à jour de la liste des packages de notre distribution.
- Installation du service web apache, l'interpréteur php avec la bibliothèque mysql et la cli (Command Line Interface) d'AWS.
- Activation permanente du service apache via la commande `systemctl`.
- Suppression de la page web d'accueil par défaut `index.html`.
- Téléchargement de nos sources depuis notre bucket S3.
- Création de l'architecture de la table de notre base de données mariadb.
- Configuration de la partie base de données de notre application web depuis la commande `sed`.

Pour finir, dans le fichier de variables de sortie de mon module racine, je récupère le nom DSN public de notre ELB :

```
output "alb_dns_name" {
  value = module.my_alb_asg.alb_dns_name
}
```

Lancement et test du projet

Pour utiliser ce projet, il faut d'abord commencer par indiquer le mot de passe root de notre base de données. Dans mon cas, j'ai choisi de le définir dans un fichier nommé `terraform.tfvars` (pour rappel ce nom de fichier est automatiquement pris en charge par Terraform lors de l'exécution de votre configuration) que je vais d'ailleurs ignorer dans mon fichier `.gitignore`. Voici à quoi ressemble ce fichier :

```
db_password = "votre-mot-de-passe"
```

Vous devez par la suite créer votre paire de clés ssh dans le dossier `keys`. Dans mon cas, j'utilise la commande `ssh-keygen` comme suit :

```
ssh-keygen -t rsa
```

Generating public/private rsa key pair.

Enter file in which to save the key (/home/hatim/.ssh/id_rsa): ./keys/terraform

Enter passphrase (empty for no passphrase):

Enter same passphrase again: Copier

DOSSIER PROFESSIONNEL (DP)

Rajoutez ensuite le chemin de votre clé publique dans le paramètre **path_to_public_key** du module **my_alb_asg**. Dans mon cas ça sera :

```
module "my_alb_asg" {  
  ...  
  ...  
  path_to_public_key = "/home/hatim/Documents/tmp/terraform/sources/keys/terraform.pub"  
  ...  
  ...  
}
```

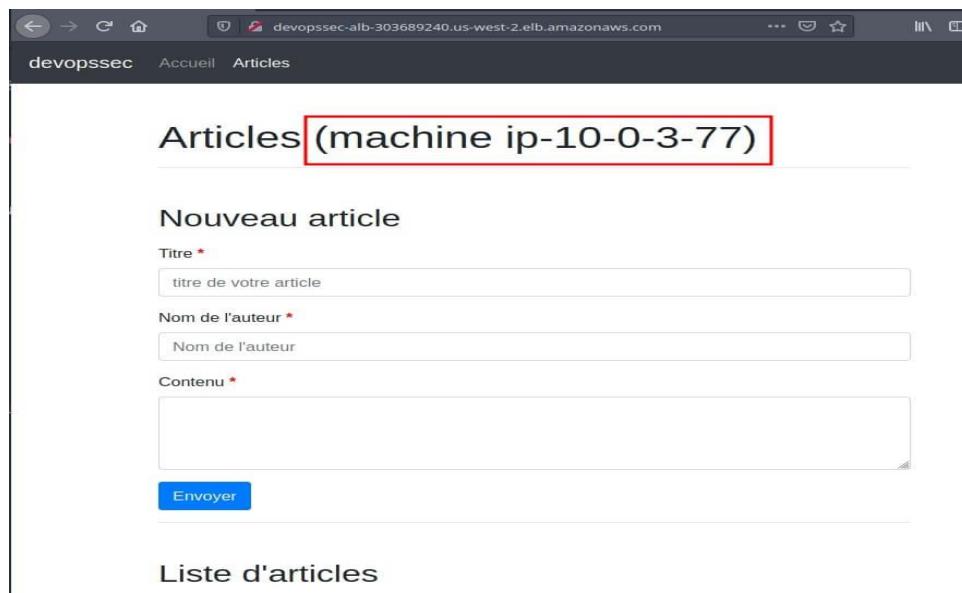
Lançons ensuite notre projet avec la commande suivante :

```
terraform init && terraform apply
```

Résultat :

```
...  
...  
Outputs:  
alb_dns_name = devopssec-alb-303689240.us-west-2.elb.amazonaws.com
```

Pour vérifier que notre ELB redirige le trafic sur les différentes instances web de notre ASG, j'ai rajouté le nom de la machine dans le code php. Ce qui nous donne la page web suivante sur notre navigateur :



DOSSIER PROFESSIONNEL (DP)

Articles (machine ip-10-0-4-157)

Nouveau article

Titre *

titre de votre article

Nom de l'auteur *

Nom de l'auteur

Contenu *

Envoyer

Liste d'articles

2. Précisez les moyens utilisés :

- Microsoft Azure (La plateforme de cloud Azure).
- Machine virtuel : Ubuntu (Système d'exploitation)
- Un émulateur de terminal MobaXterm.
- Base de données : MariaDB.
- Language : PHP
- Terraform

3. Avec qui avez-vous travaillé ?

- Travail individuel.

4. Contexte

Nom de l'entreprise, organisme ou association ► *Cliquez ici pour taper du texte.*

Chantier, atelier, service ► *Cliquez ici pour taper du texte.*

Période d'exercice ► Du : 09/10/2023 au : 13/10/2023

5. Informations complémentaires (facultatif)

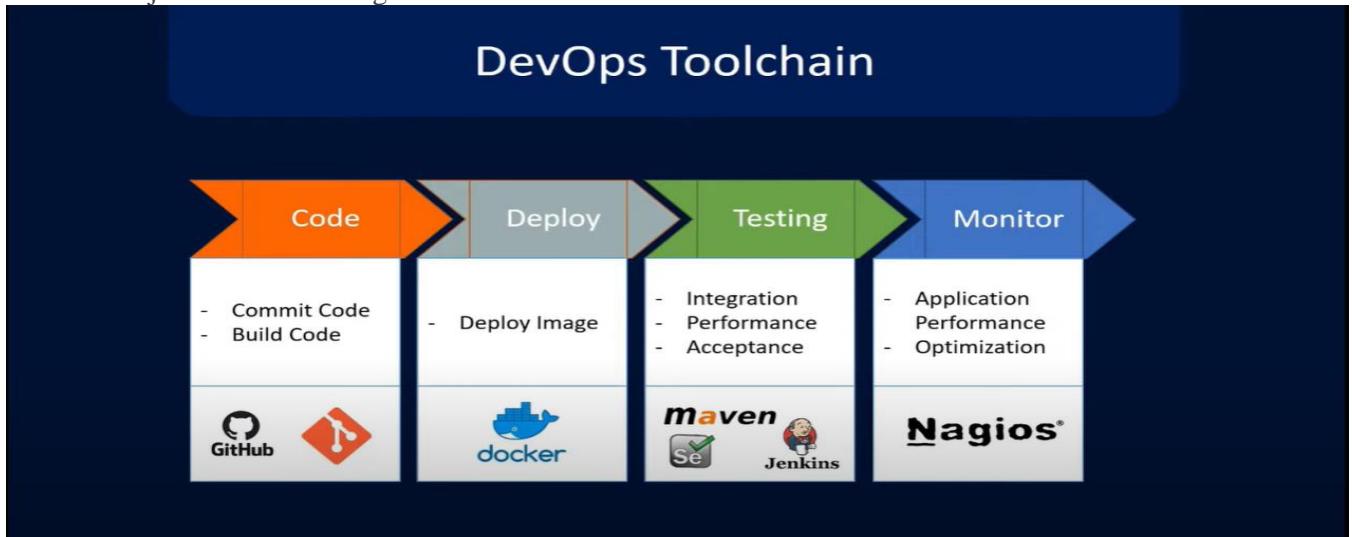
DOSSIER PROFESSIONNEL (DP)

Activité-type 2 Déployer en continu une application

Exemple n°1 ▶ Préparer un environnement de test

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Dans ce TP je vais utiliser le logiciel Jenkins et GIT.



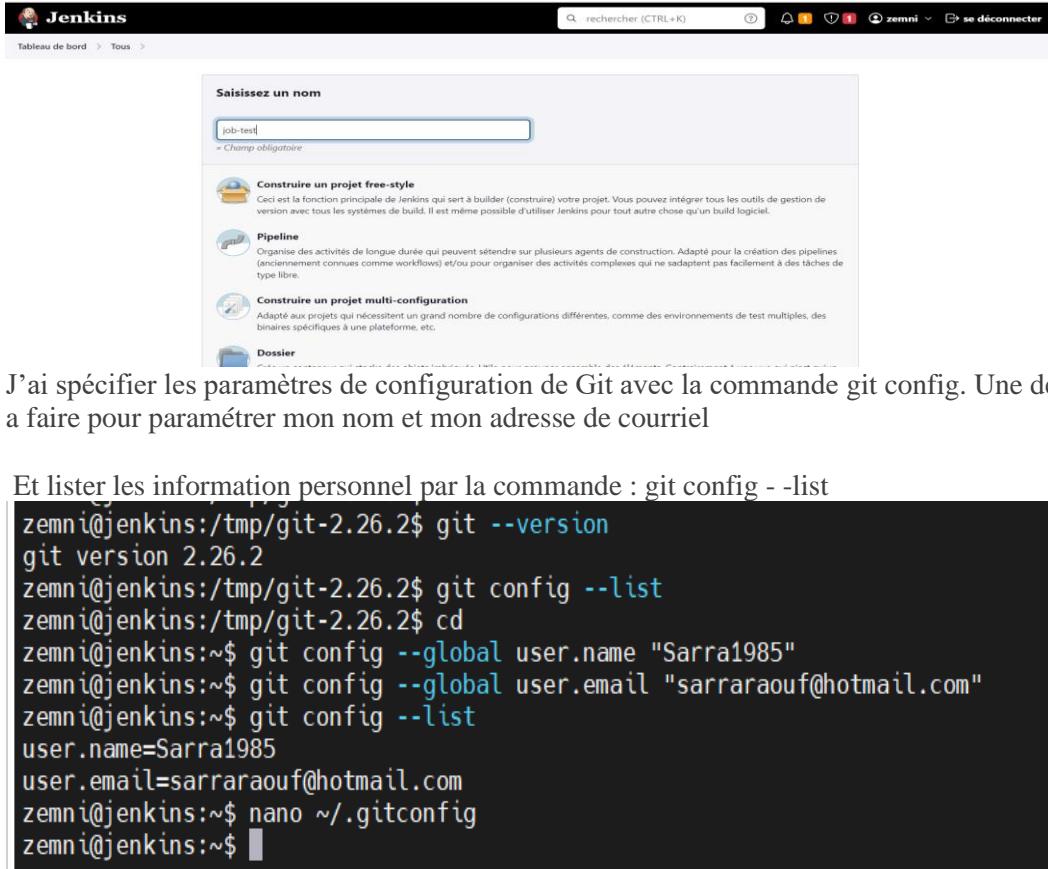
- Jenkins est un outil open source de serveur d'automatisation. Il aide à automatiser les parties du développement logiciel liées au build, aux tests et au déploiement. Il facilite l'intégration continue et va me permettre de faire l'intégration et la livraison continue. Il est écrit en Java c'est pour ça je vais débuter par l'installation de Java. Jenkins fonctionne dans un conteneur de servlets tel qu'Apache Tomcat, ou en mode autonome avec son propre serveur web embarqué. Il s'interface avec des systèmes de gestion de versions tels que CVS, GIT et Subversion (je vais utiliser GIT). Et exécute des projets basés sur Apache Ant et Apache Maven.

Jenkins va limiter les risques des erreurs d'une application, il va pousser le code sur les serveurs de production plus facilement et il va déployer l'application automatiquement et en continu.

- GIT est un logiciel de gestion de versions décentralisé. C'est le logiciel de gestion de versions le plus populaire dans le développement logiciel et web. Git indexe les fichiers d'après leur somme de contrôle calculée avec la fonction de hachage SHA-1. Quand un fichier n'est pas modifié, la somme de contrôle ne change pas et le fichier n'est stocké qu'une seule fois. En revanche, si le fichier est modifié, les deux versions sont stockées sur le disque.
- Après la création du compte Jenkins et GitHub, j'ai mis mon projet sur le compte GitHub. Pour faire les modifications et la création d'un environnement de test, j'ai cloné le projet sur ma machine.

J'ai créé une pipeline sous Jenkins, ce qui va permettre d'automatiser le test et l'intégration continue (CI/CD) de l'application. Je crée une liaison entre le Projet Jenkins avec l'application dans le compte GitHub.

DOSSIER PROFESSIONNEL (DP)



J'ai spécifier les paramètres de configuration de Git avec la commande git config. Une des premières choses à faire pour paramétrer mon nom et mon adresse de courriel

```
zemni@jenkins:/tmp/git-2.26.2$ git --version
git version 2.26.2
zemni@jenkins:/tmp/git-2.26.2$ git config --list
zemni@jenkins:/tmp/git-2.26.2$ cd
zemni@jenkins:~$ git config --global user.name "Sarra1985"
zemni@jenkins:~$ git config --global user.email "sarraraouf@hotmail.com"
zemni@jenkins:~$ git config --list
user.name=Sarra1985
user.email=sarraraouf@hotmail.com
zemni@jenkins:~$ nano ~/.gitconfig
zemni@jenkins:~$
```

1. Créer une nouvelle branche nommée « master »

```
$ git branch feature_x
```

2.Basculer vers une branche existante

```
$ git checkout feature_x
```

- Enregistrer des modifications dans un dépôt

- Pousser son travail sur un dépôt distant

```
$ git push origin master
```

La commande «push» sert à envoyer tout les «commits» effectués se trouvant dans le répertoire Git/dépôt (HEAD) de la copie du dépôt local vers le dépôt distant.

- Récupérer et tirer depuis des dépôts distants

```
$ git pull
```

La commande «pull» permet de mettre à jour votre dépôt local des dernières validations (modifications des fichiers). La commande est faite avant l'indexation des modifications.

DOSSIER PROFESSIONNEL (DP)

```
zemni@jenkins:~$ mkdir projetfilm
zemni@jenkins:~$ cd projetfilm
zemni@jenkins:~/projetfilm$ git clone https://github.com/Sarra1985/filmAPI.git
Clonage dans 'filmAPI'...
remote: Enumerating objects: 16, done.
remote: Counting objects: 100% (16/16), done.
remote: Compressing objects: 100% (14/14), done.
remote: Total 16 (delta 4), reused 0 (delta 0), pack-reused 0
Réception d'objets: 100% (16/16), 4.59 Kio | 587.00 Kio/s, fait.
Résolution des deltas: 100% (4/4), fait.
zemni@jenkins:~/projetfilm$ ls
filmAPI
zemni@jenkins:~/projetfilm$ cd filmAPI/
zemni@jenkins:~/projetfilm/filmAPI$ ls
index.html script1.js script.js styles.css
zemni@jenkins:~/projetfilm/filmAPI$ git init
Dépôt Git existant réinitialisé dans /home/zemni/projetfilm/filmAPI/.git/
zemni@jenkins:~/projetfilm/filmAPI$ ls -la
total 32
drwxrwxr-x 4 zemni zemni 4096 déc. 24 19:03 .
drwxrwxr-x 3 zemni zemni 4096 déc. 24 19:03 ..
drwxrwxr-x 8 zemni zemni 4096 déc. 24 19:06 .git
drwxrwxr-x 3 zemni zemni 4096 déc. 24 19:03 .github
-rw-rw-r-- 1 zemni zemni 655 déc. 24 19:03 index.html
-rw-rw-r-- 1 zemni zemni 1722 déc. 24 19:03 script1.js
-rw-rw-r-- 1 zemni zemni 1261 déc. 24 19:03 script.js
-rw-rw-r-- 1 zemni zemni 1585 déc. 24 19:03 styles.css
zemni@jenkins:~/projetfilm/filmAPI$ git status
Sur la branche main
Votre branche est à jour avec 'origin/main'.
rien à valider, la copie de travail est propre
zemni@jenkins:~/projetfilm/filmAPI$ git branch master
zemni@jenkins:~/projetfilm/filmAPI$ git checkout master
Basculement sur la branche 'master'
zemni@jenkins:~/projetfilm/filmAPI$ git status
Sur la branche master
rien à valider, la copie de travail est propre
zemni@jenkins:~/projetfilm/filmAPI$ git push --force origin master
Username for 'https://github.com': Sarra1985
Password for 'https://Sarra1985@github.com':
Everything up-to-date
zemni@jenkins:~/projetfilm/filmAPI$
```

The screenshot shows the GitHub repository page for 'filmAPI'. At the top, there's a navigation bar with links for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. A search bar is also present. Below the navigation, the repository name 'filmAPI' is shown along with its status as 'Public'. There are buttons for Pin and Unwatch. The main content area shows the 'master' branch details, including a message stating 'This branch is up to date with main.' Below this, a list of commits is displayed, all made by 'Sarra1985'. The commits are: 'Add or update the Azure App Service build and deployment workflow config' (8f87516, 3 months ago), '.github/workflows' (3 months ago), 'index.html' (10 months ago), 'script.js' (10 months ago), 'script1.js' (10 months ago), and 'styles.css' (10 months ago). At the bottom of the commit list, there's a link to 'README'.

Pour voir quels serveurs distants j'ai configurés, j'ai exécuter la git remotecommande. Il répertorie les noms abrégés de chaque handle distant que j'ai spécifié. J'ai cloné mon référentiel, je peu au moins voir l'origin : c'est le nom par défaut que Git donne au serveur à partir duquel vous avez cloné : **\$ git remote -v**

DOSSIER PROFESSIONNEL (DP)

```
zemni@jenkins:~/projetfilm/filmAPI$ git remote -v
origin  https://github.com/Sarra1985/filmAPI.git (fetch)
origin  https://github.com/Sarra1985/filmAPI.git (push)
zemni@jenkins:~/projetfilm/filmAPI$ █
```

\$ git log => va lister la liste des étapes des modifications .

Et on peut modifier le nom de commit par la étape suivante :

- Indexer l'ajout ou les changements d'un fichier avant de soumettre (commit) les modifications
\$ git add -A
- Enregistrer des modifications dans un dépôt : Valider les modifications
\$ git commit -m « Mon premier commit »

La commande «commit» est faite pour valider ceux qui ont été indexés avec «git add». Pas d'index pas de validation. Après l'option -m est suivi d'un commentaire de l'utilisateur décrivant ce qui a été accompli et le fichier est ajouté au répertoire Git/dépôt (local) mais pas encore sur le dépôt distant.

```
zemni@jenkins:~/projetfilm/filmAPI$ git log
commit 8f8751641014b8037489132b139aa1fca4d99be7 (HEAD -> master, origin/master, origin/main
, origin/HEAD, main)
Author: Sarra1985 <110898069+Sarra1985@users.noreply.github.com>
Date:   Wed Sep 6 16:39:05 2023 +0200

    Add or update the Azure App Service build and deployment workflow config

commit fbdb15afd8216146b0d36b3293cdad5b2f55a102
Author: Sarra1985 <110898069+Sarra1985@users.noreply.github.com>
Date:   Wed Sep 6 16:07:16 2023 +0200

    ci: add Azure Static Web Apps workflow file
    on-behalf-of: @Azure opensource@microsoft.com

commit 6a2b99c3374aaaae258555bee5d2d129026f47e1e
Author: Sarra1985 <110898069+Sarra1985@users.noreply.github.com>
Date:   Sun Feb 12 22:46:02 2023 +0100

    Add files via upload
zemni@jenkins:~/projetfilm/filmAPI$ █
```

J'ai modifié le fichier index.html et j'ai taper la commande \$ git status

```
zemni@jenkins:~/projetfilm/filmAPI$ git status
Sur la branche master
Modifications qui ne seront pas validées :
  (utilisez "git add <fichier>..." pour mettre à jour ce qui sera validé)
  (utilisez "git restore <fichier>..." pour annuler les modifications dans le répertoire de travail)
    modifié :          index.html

aucune modification n'a été ajoutée à la validation (utilisez "git add" ou "git commit -a")
zemni@jenkins:~/projetfilm/filmAPI$ █
```

Ses deux commandes permettent d'ajouter un contrôle de version, en les préparant à être validés lors de cette commit.

```
zemni@jenkins:~/projetfilm/filmAPI$ git add .
zemni@jenkins:~/projetfilm/filmAPI$ git commit -m "modifie n=1 index.html"
[master 01a8a4f] modifier n=1 index.html
  1 file changed, 2 insertions(+), 2 deletions(-)
zemni@jenkins:~/projetfilm/filmAPI$ git log
zemni@jenkins:~/projetfilm/filmAPI$ █
```

DOSSIER PROFESSIONNEL (DP)

```
zemni@jenkins:~/projetfilm/filmAPI$ git log
commit 01a8a4fc33fe61999b3687ea3323a54188b0148e (HEAD -> master)
Author: Sarra1985 <sarraraouf@hotmail.com>
Date:   Wed Dec 27 23:29:13 2023 +0000

    modifier n=1 index.html

commit 8f8751641014b8037489132b139aa1fca4d99be7 (origin/master, origin/main, origin/HEAD, main)
Author: Sarra1985 <110898069+Sarra1985@users.noreply.github.com>
Date:   Wed Sep 6 16:39:05 2023 +0200

    Add or update the Azure App Service build and deployment workflow config

commit fbdb15af8216146b0d36b3293cdad5b2f55a102
Author: Sarra1985 <110898069+Sarra1985@users.noreply.github.com>
Date:   Wed Sep 6 16:07:16 2023 +0200

    ci: add Azure Static Web Apps workflow file
    on-behalf-of: @Azure opensource@microsoft.com

commit 6a2b99c3374aaaae258555bee5d2d129926f47e1e
Author: Sarra1985 <110898069+Sarra1985@users.noreply.github.com>
Date:   Sun Feb 12 22:46:02 2023 +0100

    Add files via upload
zemni@jenkins:~/projetfilm/filmAPI$
```

Tableau de bord > job-test > Configuration

Configure

General

Gestion de code source

Ce qui déclenche le build

Environnements de Build

Étapes du build

Actions à la suite du build

Repository URL: https://github.com/Sarra1985/filmAPI.git

Credentials: aucun

Avancé

Add Repository

Branches to build:

Branch Specifier (blank for 'any'): */master

Add Branch

Sauver Apply

Étapes du build

Exécuter un script shell

Commande

Voir la liste des variables d'environnement disponibles

```
git config -global user.email "sarraraouf@hotmail.com"
git config -global user.name "Sarra1985"
```

Avancé

Ajouter une étape au build

Actions à la suite du build

Sauver Apply

DOSSIER PROFESSIONNEL (DP)

Configure

Actions à la suite du build

Git Publisher

Push Only If Build Succeeds

Merge Results
If pre-build merging is configured, push the result back to the origin

Force Push
Add force option to git push

Tags

Tags to push to remote repositories

Tag to push
JENKINS-\$BUILD_ID

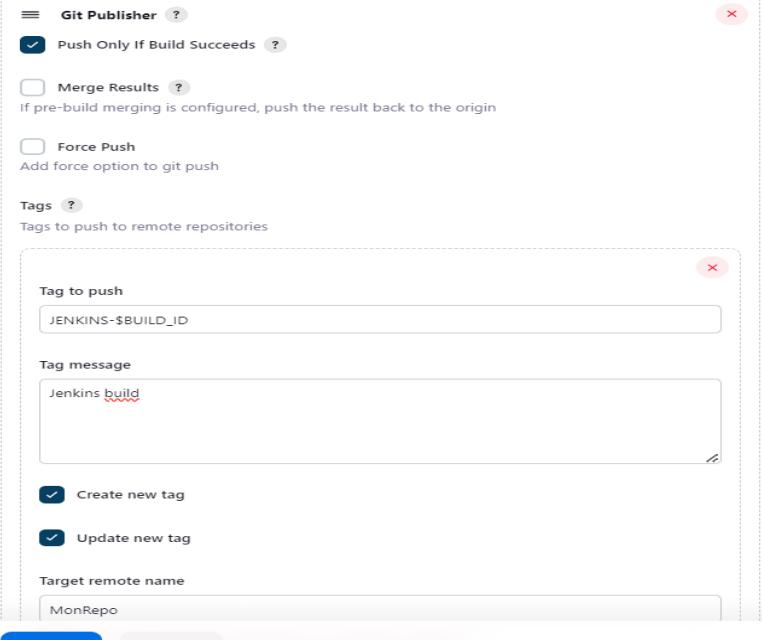
Tag message
Jenkins build

Create new tag

Update new tag

Target remote name
MonRepo

Sauver **Apply**



Jenkins

Tableau de bord >

+ Nouveau Item Ajouter une descrip

Utilisateurs Tous +

Historique des constructions

Administrer Jenkins

Mes vues

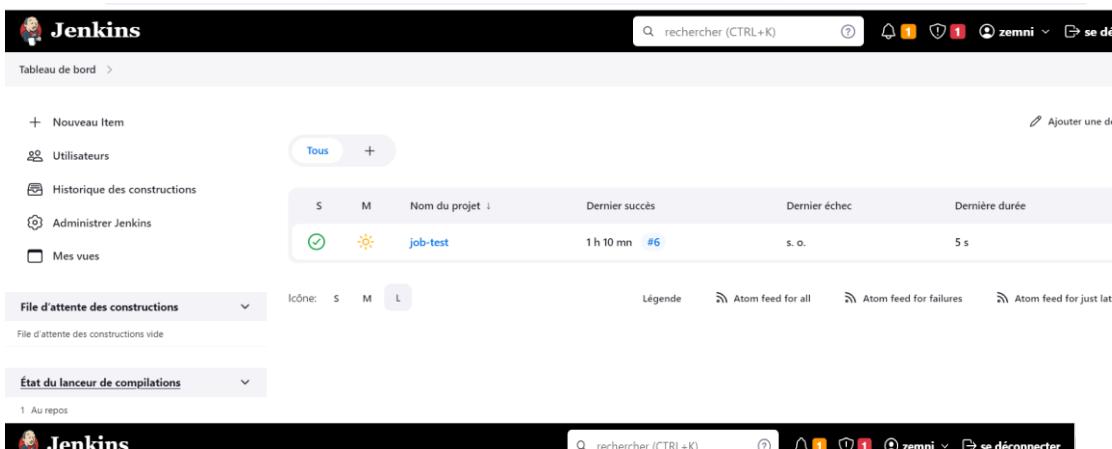
| S | M | Nom du projet | Dernier succès | Dernier échec | Dernière durée |
|---|---|---------------|----------------|---------------|----------------|
| | | job-test | 1 h 10 mn #6 | s. o. | 5 s |

File d'attente des constructions Icone: S M L

Atom feed for all Atom feed for failures Atom feed for just latest build

État du lanceur de compilations

1 Au repos



Jenkins

Tableau de bord > job-test > Workspace

État

Répertoire de travail job-test / index.html

Effacer l'espace de travail

Lancer un build

Configurer

Supprimer Projet

GitHub

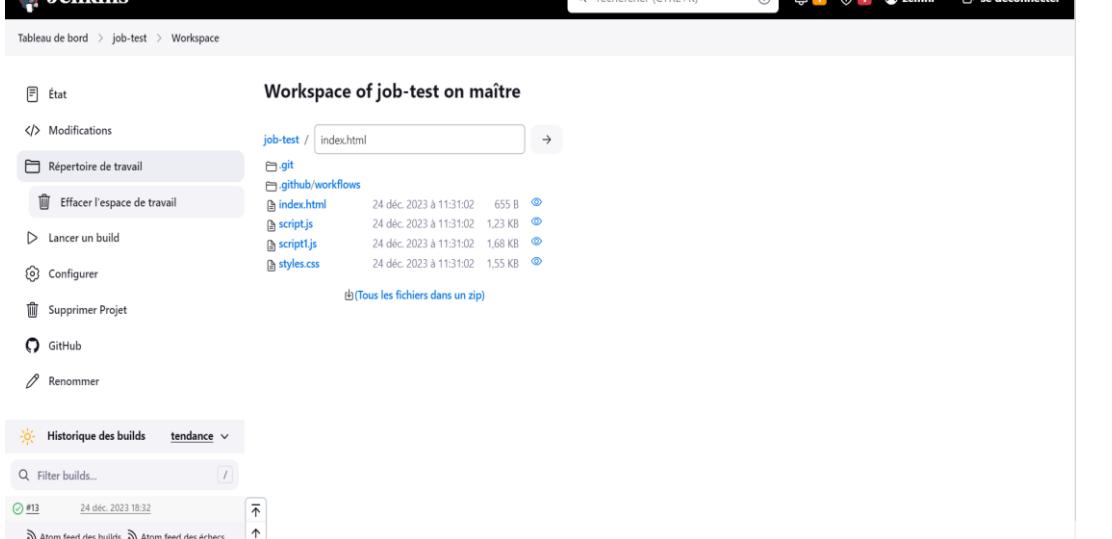
Renommer

Historique des builds tendance

Filter builds... /

#13 24 déc. 2023 18:32 ↑

Atom feed des builds Atom feed des échecs



DOSSIER PROFESSIONNEL (DP)

The screenshot shows the GitHub Actions CI/CD interface for a repository named 'filmAPI'. The pipeline is defined in a YAML file named 'azure-static-web-apps-thankful-tree-038457410.yml'. The pipeline consists of four stages: 'Cloning Git', 'Build', 'Test', and 'Deploy'. The 'Cloning Git' stage uses the 'git' step to clone the repository from 'https://github.com/Sarra1985/filmAPI.git'. The 'Build' stage uses the 'sh' step with the command 'make'. The 'Test' stage uses the 'sh' step with the command 'make check' and the 'junit' step to process XML reports. The 'Deploy' stage uses the 'sh' step with the command 'make publish'. There are two workflow runs listed: one for commit '8f87516' (labeled 'Add or update the Azure App Service build and deployment...') and another for commit 'fbdb15a' (labeled 'ci: add Azure Static Web Apps workflow file'). Both runs were triggered by pushes from the 'main' branch.

```
pipeline {
  agent any
  stages {
    stage('Cloning Git') {
      steps {
        git 'https://github.com/Sarra1985/filmAPI.git'
      }
    }
    stage('Build') {
      steps {
        sh 'make'
      }
    }
    stage('Test'){
      steps {
        sh 'make check'
        junit 'reports/**/*.xml'
      }
    }
    stage('Deploy') {
      steps {
        sh 'make publish'
      }
    }
  }
}
```

DOSSIER PROFESSIONNEL (DP)

The screenshot shows the Jenkins interface for the 'projet-test' pipeline. On the left, there's a sidebar with various options like Status, Changes, Lancer un build, Configurer, Supprimer Pipeline, Full Stage View, GitHub, Favoris, Open Blue Ocean, Stages, Renommer, and Pipeline Syntax. The main area is titled 'Stage View' and displays a timeline of stages: Cloning Git (3s), Build (471ms), Test (86ms), and Deploy (93ms). Below this, a table shows a single build (#12) from December 26 at 00:41, which had no changes. A section titled 'Liens permanents' lists recent builds: 'Dernier build en échec (#11), il y a 1 mn 38 s', 'Dernier build non réussi (#11), il y a 1 mn 38 s', and 'Last completed build (#11), il y a 1 mn 38 s'. At the bottom, tabs for 'Historique des builds' and 'tendance' are visible.

Il est possible de créer une nouvelle branche à partir de cette version grâce à la commande git checkout

```
git checkout -b old-state 3u38t0qn00p1sr1o8012r16kp2nsxcmul1w426w5
```

Pour revenir à la version actuelle du répertoire de travail, il faudra refaire un check-out.

Il est possible de revenir en arrière et de supprimer toutes les modifications qui ont été enregistrées depuis la révision mais la commande ne sera pas la même si on a publié les commits ou non.

Si tous les commits n'ont pas été publiés, il suffit de revenir à la version antérieure du répertoire de travail grâce à la commande git reset :

```
git reset --hard 3u38t0qn00p1sr1o8012r16kp2nsxcmul1w426w5
```

Si on a des modifications non enregistrées dans le répertoire de travail. Pour les conserver, il faut exécuter en plus la commande get stash.

```
git stash #sauvegarde les modifications
```

```
git reset --hard 3u38t0qn00p1sr1o8012r16kp2nsxcmul1w426w5
```

```
#remet à zéro le répertoire de travail par rapport à la révision spécifiée
```

```
git stash pop #applique les modifications sauvegardées
```

Si un fichier est concerné à la fois par les modifications en cours et le retour à la version antérieure du répertoire de travail, il y aura un conflit à résoudre.

Si les commits ont été publiés, il est possible de créer un patch qui correspondra aux modifications à effectuer pour revenir en arrière, puis enregistrer les modifications avec un nouveau commit :

```
git revert 3u38t0qn00p1sr1o8012r16kp2nsxcmul1w426w5
```

DOSSIER PROFESSIONNEL (DP)

2. Précisez les moyens utilisés :

- Microsoft Azure (La plateforme de cloud Azure).
- La machine virtuel chez Azure Cloud : Ubuntu (Système d'exploitation) .
- Un émulateur de terminal MobaXterm.
- Jenkins
- GIT

3. Avec qui avez-vous travaillé ?

- Travail personnel

4. Contexte

Nom de l'entreprise, organisme ou association ► *Cliquez ici pour taper du texte.*

Chantier, atelier, service ► *Cliquez ici pour taper du texte.*

Période d'exercice ► Du : 23/10/2023 au : 27/10/2023

5. Informations complémentaires (*facultatif*)

DOSSIER PROFESSIONNEL (DP)

Activité-type 2 Déployer en continu une application

Exemple n°2 ▶ Gérer le stockage des données

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

MySQL est un système de gestion de base de données open source, généralement installé dans le cadre de la populaire pile LAMP (Linux, Apache, MySQL, PHP/Python/Perl). Il met en œuvre le modèle relationnel et utilise le langage de requête structuré (mieux connu sous le nom de SQL – Structured Query Language) pour gérer ses données.

- **Installation de MySQL** :
\$ sudo apt update

Installer le paquet mysql-server :
\$ sudo apt install mysql-server

Cela installera MySQL, mais ne demandera pas de définir un mot de passe ou de faire d'autres changements de configuration. Comme cela rend l'installation de MySQL non sécurisée, je vais aborder ce point.

- **Configuration de MySQL**

Pour les nouvelles installations de MySQL, vous devrez exécuter le script de sécurité inclus dans le SGBD. Ce script modifie certaines des options par défaut les moins sûres pour des choses comme les connexions root distantes et les sample users.

Exécutez le script de sécurité avec sudo :
\$ sudo mysql_secure_installation

Cette étape permet d'apporter une modification aux options de sécurité de l'installation MySQL. La première, demandera la configuration du plugin **Validate Password**, pour tester la solidité de votre mot de passe MySQL.

Mettre en place le plugin **Validate Password**, le script demandera de choisir un niveau de validation du mot de passe. Le niveau le plus fort – sélectionner en entrant 2 - exigera que le mot de passe comporte au moins huit caractères, dont un mélange de majuscules, de minuscules, de chiffres et de caractères spéciaux :

```
zemni@test2:~$ sudo mysql_secure_installation
[sudo] password for zemni:
Securing the MySQL server deployment.

Connecting to MySQL using a blank password.

VALIDATE PASSWORD COMPONENT can be used to test passwords
and improve security. It checks the strength of password
and allows the users to set only those passwords which are
secure enough. Would you like to setup VALIDATE PASSWORD component?

Press y|Y for Yes, any other key for No: y

There are three levels of password validation policy:

LOW    Length >= 8
MEDIUM Length >= 8, numeric, mixed case, and special characters
STRONG Length >= 8, numeric, mixed case, special characters and dictionary file

Please enter 0 = LOW, 1 = MEDIUM and 2 = STRONG: 2

Skipping password set for root as authentication with auth_socket is used by default.
If you would like to use password authentication instead, this can be done with the "ALTER USER" command.
See https://dev.mysql.com/doc/refman/8.0/en/alter-user.html#alter-user-password-management for more information.

By default, a MySQL installation has an anonymous user,
allowing anyone to log into MySQL without having to have
a user account created for them. This is intended only for
testing, and to make the installation go a bit smoother.
You should remove them before moving into a production
environment.

Remove anonymous users? (Press y|Y for Yes, any other key for No) : y
Success.
```

DOSSIER PROFESSIONNEL (DP)

Choisir ou non de configurer le plugin `Validate Password`, l'invite suivante demandera de définir un mot de passe pour l'utilisateur `root` de MySQL. Entrer et confirmer le mot de passe sécurisé aux choix : Utiliser le plugin `Validate Password`, permet de recevoir des commentaires sur la force du nouveau mot de passe.

Ensuite, le script demandera si je veux continuer avec le mot de passe que je viens de saisir ou si je veux en saisir un nouveau.

En supposant que je suis satisfais de la force du mot de passe que je vient d'entrer, saisir Y pour poursuivre le script : À partir de là, je peux appuyer sur Y puis sur ENTER pour accepter les valeurs par défaut pour toutes les questions suivantes.

Cela supprimera les utilisateurs anonymes et la base de données de test, désactivera les connexions root à distance, et chargera ces nouvelles règles afin que MySQL respecte immédiatement les modifications que j'ai apporté.

Même si j'ai défini un mot de passe pour l'utilisateur MySQL `root`, cet utilisateur n'est pas configuré pour s'authentifier avec un mot de passe lors de la connexion au shell MySQL. Si je le souhaite, je peux ajuster ce paramètre en suivant l'étape 3.

- **Ajustement de l'authentification et des priviléges de l'utilisateur :**

L'utilisateur MySQL `root` est configuré pour s'authentifier en utilisant le plugin `auth_socket` par défaut plutôt qu'avec un mot de passe. Cela permet d'améliorer la sécurité et la convivialité dans de nombreux cas, mais cela peut aussi compliquer les choses si je devais autoriser un programme externe (par exemple, phpMyAdmin) à accéder à l'utilisateur.

Afin d'utiliser un mot de passe pour se connecter à MySQL en tant que `root`, je devrais changer sa méthode d'authentification de `auth_socket` à un autre plugin, tel que `caching_sha2_password` ou `mysql_native_password`. Pour ce faire, ouvrez l'invite MySQL depuis mon terminal :

```
root@test2:~# sudo mysql
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 12
Server version: 8.0.35-0ubuntu0.22.04.1 (Ubuntu)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> SELECT user,authentication_string,plugin,host FROM mysql.user;
+-----+-----+-----+
| user | authentication_string | plugin | host |
+-----+-----+-----+
| debian-sys-maint | $A$005$&] r
(..4{Ga)P
X$6q2vD7stKG9Y3H33jzzq0EIyRR9UuBzDfzK6JbCmV/ | caching_sha2_password |
| localhost |
| mysql.infoschema | $A$005$THISISACOMBINATIONOFINVALIDSALTANDPASSWORD
THATMUSTNEVERBEUSED | caching_sha2_password | localhost |
| mysql.session | $A$005$THISISACOMBINATIONOFINVALIDSALTANDPASSWORD
THATMUSTNEVERBEUSED | caching_sha2_password | localhost |
| mysql.sys | $A$005$THISISACOMBINATIONOFINVALIDSALTANDPASSWORD
THATMUSTNEVERBEUSED | caching_sha2_password | localhost |
| root | auth_socket | localhost |
+-----+-----+-----+
5 rows in set (0,00 sec)
```

Ensuite, vérifier la méthode d'authentification utilisée par chacun de vos comptes utilisateurs MySQL à l'aide de la commande suivante : Dans cet exemple, je pouvais voir que l'utilisateur `root` s'authentifie effectivement en utilisant le plugin `auth_socket`. Pour configurer le compte `root` afin qu'il s'authentifie avec un mot de passe, lancez une instruction `ALTER USER` pour modifier le plugin d'authentification utilisé et définir un nouveau mot de passe. Je remplace `password` par un mot de passe fort, et souvenez-vous que cette

DOSSIER PROFESSIONNEL (DP)

commande modifiera le mot de passe **root** que vous avez défini à l'étape 2 :

```
mysql> ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_pass
      word BY 'Sarra22919758&';
Query OK, 0 rows affected (0,19 sec)

mysql> ALTER USER 'root'@'localhost' IDENTIFIED WITH caching_sha2_password
      word BY 'Sarra22919758&';
Query OK, 0 rows affected (0,04 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0,03 sec)

mysql> SELECT user,authentication_string,plugin,host FROM mysql.user;
+-----+-----+-----+
| user | authentication_string | plugin | host |
+-----+-----+-----+
| debian-sys-maint | $A$005$&] r
| (.4{Ga)P
| XS6q2vD7stKG9Y3H33jzzq0EiyRR9UuBzDfzK6JbCmV/ | caching_sha2_password | localhost |
| mysql.infoschema | $A$005$THISISACOMBINATIONOFPASSWORDTHATMUSTNEVERBEUSED | caching_sha2_password | localhost |
| mysql.session | $A$005$THISISACOMBINATIONOFPASSWORDTHATMUSTNEVERBEUSED | caching_sha2_password | localhost |
| mysql.sys | $A$005$THISISACOMBINATIONOFPASSWORDTHATMUSTNEVERBEUSED | caching_sha2_password | localhost |
| root | $A$005$4ON3!p9A\p28ZMhz0ArjhBNWruwDgQG8i.UhxxLjdnSHGMlqmfpKD | caching_sha2_password | localhost |
+-----+-----+-----+
5 rows in set (0,00 sec)
```

Après l'authentification par le mot de passe pour **root**, pour accéder au shell MySQL, s'identifier et connecter il faut taper cette commande :

```
zemni@test2:~$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 30
Server version: 8.0.35-0ubuntu0.22.04.1 (Ubuntu)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> █
```

J'exécute ensuite FLUSH PRIVILEGES qui ordonne au serveur de recharger les tableaux de subventions et de mettre vos nouvelles modifications en vigueur :

```
mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (1,93 sec)

mysql> █
```

Je vérifie à nouveau les méthodes d'authentification utilisées par chacun de mes utilisateurs pour confirmer que **root** ne s'authentifie plus à l'aide du plugin **auth_socket** :

```
mysql> SELECT user,authentication_string,plugin,host FROM mysql.user;
+-----+-----+-----+
| user | authentication_string | plugin | host |
+-----+-----+-----+
| debian-sys-maint | $A$005$&] r
| (.4{Ga)P
| XS6q2vD7stKG9Y3H33jzzq0EiyRR9UuBzDfzK6JbCmV/ | caching_sha2_password | localhost |
| mysql.infoschema | $A$005$THISISACOMBINATIONOFPASSWORDTHATMUSTNEVERBEUSED | caching_sha2_password | localhost |
| mysql.session | $A$005$THISISACOMBINATIONOFPASSWORDTHATMUSTNEVERBEUSED | caching_sha2_password | localhost |
| mysql.sys | $A$005$THISISACOMBINATIONOFPASSWORDTHATMUSTNEVERBEUSED | caching_sha2_password | localhost |
| root | $A$005$4ON3!p9A\p28ZMhz0ArjhBNWruwDgQG8i.UhxxLjdnSHGMlqmfpKD | caching_sha2_password | localhost |
+-----+-----+-----+
5 rows in set (0,01 sec)

mysql> █
```

Je peux voir dans cet exemple que l'utilisateur MySQL **root** s'authentifie maintenant en utilisant le plugin **caching_sha2_password**.

De là, crée un nouvel utilisateur et attribuez-lui un mot de passe fort :

```
mysql> CREATE USER 'zemni'@'localhost' IDENTIFIED BY 'Sarra22919758&';
Query OK, 0 rows affected (4,08 sec)

mysql> █
```

DOSSIER PROFESSIONNEL (DP)

Ensuite, accorder à mon nouvel utilisateur les priviléges appropriés. Je peux accorder les priviléges d'utilisateur à tous les tableaux de la base de données, ainsi que le pouvoir d'ajouter, de modifier et de supprimer des priviléges d'utilisateur, avec cette commande :

```
mysql> GRANT ALL PRIVILEGES ON *.* TO 'zemni'@'localhost' WITH GRANT OPTION;  
Query OK, 0 rows affected (0,79 sec)
```

A ce stade, j'ai pas besoin d'exécuter à nouveau la commande FLUSH PRIVILEGES. Cette commande n'est nécessaire que lorsque je modifie les tableaux de subvention en utilisant des instructions comme INSERT, UPDATE ou DELETE. Comme j'ai créé un nouvel utilisateur, au lieu de modifier un utilisateur existant, le FLUSH PRIVILEGES est inutile ici.

▪ Test de MySQL

Quelle que soit la manière dont vous l'avez installé, MySQL aurait dû commencer à fonctionner automatiquement. Pour le tester, vérifiez son état.

```
zemni@test2:~$ systemctl status mysql.service  
● mysql.service - MySQL Community Server  
  Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vendor  
  Active: active (running) since Thu 2023-12-28 00:58:46 UTC; 12h >  
    Process: 2743 ExecStartPre=/usr/share/mysql/mysql-systemd-start p  
  Main PID: 2759 (mysqld)  
    Status: "Server is operational"  
      Tasks: 39 (limit: 2100)  
     Memory: 363.6M  
        CPU: 2min 54.291s  
       CGroup: /system.slice/mysql.service  
                 └─2759 /usr/sbin/mysqld  
  
déc. 28 00:58:44 test2 systemd[1]: Starting MySQL Community Server...  
déc. 28 00:58:46 test2 systemd[1]: Started MySQL Community Server.  
lines 1-14/14 (END)
```

▪ Création d'un exemple de base de données .

Tout d'abord, je connecte à mon serveur MySQL en tant que zemni. Après exécutez la commande suivante pour créer une base de données devops ,

```
zemni@test2:~$ mysql -u zemni -p  
Enter password:  
Welcome to the MySQL monitor. Commands end with ; or \g.  
Your MySQL connection id is 31  
Server version: 8.0.35-0ubuntu0.22.04.1 (Ubuntu)  
  
Copyright (c) 2000, 2023, Oracle and/or its affiliates.  
  
Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective  
owners.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
  
mysql> SHOW TABLES;  
ERROR 1046 (3D000): No database selected  
mysql> Create database devops;  
Query OK, 1 row affected (0,73 sec)
```

après accéder à devops créer une table de clients customers. Cette table contiendra les enregistrements des clients (customers), y compris les customer_id, customer_name, et level. Il y aura deux niveaux de clients : BASIC et VIP

```
mysql> Use devops;  
Database changed  
mysql> Create table customers(customer_id BIGINT PRIMARY KEY, customer  
_name VARCHAR(50), level VARCHAR(50) ) ENGINE=INNODB;  
Query OK, 0 rows affected (1,88 sec)
```

Ajoutez quelques enregistrements à la table customers.

DOSSIER PROFESSIONNEL (DP)

```
mysql> Insert into customers (customer_id, customer_name, level )value  
s('1','JOHN DOE','BASIC');  
Query OK, 1 row affected (1,35 sec)  
  
mysql> Insert into customers (customer_id, customer_name, level )value  
s('2','MARY ROE','BASIC');  
Query OK, 1 row affected (0,40 sec)  
  
mysql> Insert into customers (customer_id, customer_name, level )value  
s('3','JOHN DOE','VIP');  
Query OK, 1 row affected (0,09 sec)
```

Les échantillons d'enregistrements ont été insérés avec succès, exécutez la commande `SELECT` :

```
mysql> SHOW TABLES;  
+-----+  
| Tables_in_devops |  
+-----+  
| customers |  
+-----+  
1 row in set (0,03 sec)  
  
mysql> Select * from customers;  
+-----+-----+-----+  
| customer_id | customer_name | level |  
+-----+-----+-----+  
| 1 | JOHN DOE | BASIC |  
| 2 | MARY ROE | BASIC |  
| 3 | JOHN DOE | VIP |  
+-----+-----+-----+  
3 rows in set (0,00 sec)
```

2. Précisez les moyens utilisés :

- Microsoft Azure (La plateforme de cloud Azure).
- Machine virtuelle chez Azure Cloud : Ubuntu (Système d'exploitation)
- Un émulateur de terminal MobaXterm
- Base de Données :MySql

3. Avec qui avez-vous travaillé ?

- Travail individuel

4. Contexte

Nom de l'entreprise, organisme ou association ► *Cliquez ici pour taper du texte.*

Chantier, atelier, service ► *Cliquez ici pour taper du texte.*

Période d'exercice ► Du : *06/11/2023* au : *10/11/2023*

5. Informations complémentaires (*facultatif*)

DOSSIER PROFESSIONNEL (DP)

Activité-type 2 Déployer en continu une application

Exemple n°3 ▶ Gérer des containers

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

En cherchant dans le Hub Docker, j'ai pu dénicher les images adéquates, notamment :

- Une image officielle php avec le tag 7-apache
- Une image officielle mysql

Une fois que j'ai trouvé les bonnes images, je peux alors m'attaquer à la modification du Dockerfile.

Pour le moment, nous utiliserons ce Dockerfile seulement pour construire une image avec une couche OS, Apache et Php sans implémenter aucun service de base de données. Cette image se basera sur l'image officielle php avec le tag 7-apache qui vient déjà avec un OS (distribution Ubuntu). Concernant l'image mysql nous l'utiliserons plus tard dans notre docker-compose.yml.

Dans le même dossier que vous avez désarchivé, créez un fichier Dockerfile et mettez dedans le contenu suivant :

Buildez ensuite votre image avec la commande suivante :

```
GNU nano 6.2                                            Dockerfile
FROM php:7-apache
LABEL version="1.0" maintainer="ZEMNI Sarra"
# Activation des modules php
RUN docker-php-ext-install pdo pdo_mysql
WORKDIR /var/www/html
```

```
zemni@test1:~/projetdocker$ sudo docker build -t devops .
[sudo] password for zemni:
[+] Building 145.9s (7/7) FINISHED
  => [internal] load .dockerignore                               docker:default
  => => transferring context: 2B                                0.2s
  => [internal] load build definition from Dockerfile        0.1s
  => => transferring dockerfile: 194B                          0.2s
  => [internal] load metadata for docker.io/library/php:7-apache   0.1s
  => [1/3] FROM docker.io/library/php:7-apache                2.1s
  => [1/3]  FROM docker.io/library/php:7-apache@sha256:c9d7e608f7383267347977ad  0.7 6s
  => => exporting layers                                       0.1s
  => => writing image sha256:64fe2c0b4172b53136e5f13f64579c383b005e37b4fafa39f0  0.0s
  => => naming to docker.io/library/devops                      0.0s
zemni@test1:~/projetdocker$
```

```
zemni@test1:~/projetdocker$ sudo docker images
REPOSITORY      TAG      IMAGE ID      CREATED          SIZE
devops          latest   ffd3fa8d5111  34 seconds ago  453MB
mysql           5.7     5107333e08a8  2 weeks ago    501MB
zemni@test1:~/projetdocker$
```

Les besoins pour notre Docker Compose

Avant de créer notre fichier docker-compose.yml, il faut auparavant **définir les comportements de nos conteneurs**.
Nos besoins pour le conteneur de la base de données

On va débuter par la récolte des besoins du conteneur de la base de données. Pour celle-ci, il nous faudra :

- Un fichier sql pour créer l'architecture de notre base de données.
- Un volume pour stocker les données.

DOSSIER PROFESSIONNEL (DP)

Avant de foncer tête baissée dans la création/modification de notre fichier sql, il toujours important de vérifier avant ce que nous propose la page Docker Hub de l'image mysql. En lisant sa description, les informations qui m'ont le plus captivé sont ses variables d'environnements qu'on peut surcharger, notamment :

- **MYSQL_ROOT_PASSWORD**: spécifie le mot de passe qui sera défini pour le compte MySQL root (**c'est une variable obligatoire**).
- **MYSQL_DATABASE**: spécifie le nom de la base de données à créer au démarrage de l'image.
- **MYSQL_USER** et **MYSQL_PASSWORD** : utilisées conjointement pour créer un nouvel utilisateur avec son mot de passe. Cet utilisateur se verra accorder des autorisations de super-utilisateur pour la base de données **MYSQL_DATABASE**.

Ces variables d'environnements vont nous aider à créer une partie de l'architecture de notre base de données.

Dans la description de l'image mysql, il existe une autre information très utile. Lorsqu'un conteneur mysql est démarré, il exécutera des fichiers avec des extensions **.sh**, **.sql** et **.sql.gz** qui se trouvent dans **/docker-entrypoint-initdb.d**. Nous allons profiter de cette information pour déposer le fichier **articles.sql** (disponible dans les sources téléchargées) dans le dossier **/docker-entrypoint-initdb.d** afin de créer automatiquement notre table SQL.

Nos besoins pour le conteneur de l'application web

Concernant le conteneur de l'application web, nous aurons besoin de :

- Une communication avec le conteneur de la base de données.
- Un volume pour stocker les sources de l'application web.

Me concernant la seule information utile dans la description de la page Docker Hub de l'image php, est qu'il est possible d'installer et d'activer les modules php dans le conteneur php avec la commande **docker-php-ext-install** (C'est la commande utilisée dans notre Dockerfile afin d'activer le module pdo et pdo_mysql).

Création du docker-compose

Contenu du docker-compose

Commencez d'abord par créer un fichier et nommez le **docker-compose.yml**, ensuite copiez collez le contenu ci-dessous. Par la suite, plus bas dans l'article, je vais vous fournir les explications des différentes lignes de ce fichier :

```
GNU nano 6.2                                            docker-compose.yml *
```

```
version: '3.7'

services:
  db:
    image: mysql:5.7
    container_name: mysql_c
    restart: always
    volumes:
      - db-volume:/var/lib/mysql
      - ./articles.sql:/docker-entrypoint-initdb.d/articles.sql
    environment:
      MYSQL_ROOT_PASSWORD: test
      MYSQL_DATABASE: test
      MYSQL_USER: test
      MYSQL_PASSWORD: test

  app:
    image: devops
    container_name: devops_c
    restart: always
    volumes:
      - ./app:/var/www/html
    ports:
      - 8080:80
    depends_on:
      - db

volumes:
  db-volume:
```

DOSSIER PROFESSIONNEL (DP)

Lancer l'application depuis docker-compose.yml

Pour être sur le même pied d'estale, voici à quoi doit ressembler votre arborescence :

```
zemni@test1:~/projetdocker$ tree
.
└── app
    ├── db-config.php
    ├── index.php
    └── validation.php
    ├── articles.sql
    └── docker-compose.yml
    Dockerfile

1 directory, 6 files
zemni@test1:~/projetdocker$
```

A un niveau du dossier qui contient le fichier docker-compose.yml. Ensuite lancez la commande suivante pour **exécuter les services du docker-compose.yml** :

```
zemni@test1:~/projetdocker$ sudo docker-compose up -d
Creating network "projetdocker_default" with the default driver
Creating volume "projetdocker_db-volume" with default driver
Creating mysql_c ... done
Creating devops_c ... done
```

Ici l'option **-d** permet d'**exécuter les conteneurs du Docker compose en arrière-plan**.

Si vous le souhaitez, vous pouvez **vérifier le démarrage des conteneurs issus du docker-compose.yml** :

```
zemni@test1:~/projetdocker$ sudo docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED             STATUS              NAMES
           PORTS
7c394e83dec1      devops            "docker-php-entrypoi..."   6 minutes ago     Up 6 minutes       devops_c
b0ee95f0e0c0      mysql:5.7        "docker-entrypoint.s..."   6 minutes ago     Restarting (1) 37 se
conds ago          mysql_c
zemni@test1:~/projetdocker$
```

Pour seulement **lister les conteneurs du docker-compose.yml**, il suffit d'exécuter la commande suivante :

```
zemni@test1:~/projetdocker$ sudo docker-compose ps
  Name            Command           State           Ports
-----
devops_c         docker-php-entrypoint apac ...   Up      0.0.0.0:8080->80/tcp,:::8080->80/tcp
mysql_c          docker-entrypoint.sh mysqld   Restarting
zemni@test1:~/projetdocker$
```

Visiter la page suivante <http://localhost:8080/>, et vous obtiendrez le résultat suivant :

DOSSIER PROFESSIONNEL (DP)

Ma propre image Docker Accueil Articles

Articles

Nouveau article

Titre *

Nom de l'auteur *

Contenu *

Envoyer

Liste d'articles

test2

06/07/19 18:38

je test l'article test 2

— test2

2. Précisez les moyens utilisés :

- Docker
- Microsoft Azure (La plateforme de cloud Azure).
- Machine Virtuel chez Cloud Azure : Ubuntu Système d'exploitation
- Un émulateur de terminal MobaXterm.
- Base de données : MariaDB
- Language : PHP

3. Avec qui avez-vous travaillé ?

- Travail Individuel

4. Contexte

Nom de l'entreprise, organisme ou association ► *Cliquez ici pour taper du texte.*

Chantier, atelier, service ► *Cliquez ici pour taper du texte.*

Période d'exercice ► Du : **13/11/2023** au : **17/11/2023**

5. Informations complémentaires (*facultatif*)

DOSSIER PROFESSIONNEL (DP)

Activité-type 2 Déployer en continu une application

Exemple n°4 ▶ Automatiser la mise en production d'une application avec une plateforme

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Nous devons permettre à Kubernetes de lire notre repository local de docker : `eval $(minikube docker-env)`
Verifier le statut des images docker : `$ docker images`

```
mohamedsirajachabbak@MBP-de-Siraj ~ % eval $(minikube docker-env)
mohamedsirajachabbak@MBP-de-Siraj ~ % docker images
REPOSITORY          TAG      IMAGE ID      CREATED       SIZE
<none>              <none>   f918ea0f4c97  24 hours ago  343MB
<none>              <none>   c65a3d2c194a  24 hours ago  559MB
<none>              <none>   30d2ef10a9fc  2 weeks ago   135MB
registry.k8s.io/kube-apiserver   v1.25.3   12dd70322f97  4 weeks ago   123MB
registry.k8s.io/kube-proxy        v1.25.3   bcc74496abfd  4 weeks ago   58MB
registry.k8s.io/kube-scheduler    v1.25.3   13b15fc3e093  4 weeks ago   49.3MB
registry.k8s.io/kube-controller-manager v1.25.3   cf6c9e4e18a3  4 weeks ago   113MB
kubernetesui/dashboard          <none>   20b332c9a70d  8 weeks ago   244MB
registry.k8s.io/pause            3.8      4e42fb3c9d90  5 months ago  514kB
registry.k8s.io/etcfd            3.5.4-0   8e041a3b0ba8  5 months ago  179MB
kubernetesui/metrics-scrapers   <none>   a422e0e98235  5 months ago  42.3MB
registry.k8s.io/coredns/coredns  v1.9.3   b19406328e70  5 months ago  47.7MB
k8s.gcr.io/pause                3.6      7d46a07936af  14 months ago  484kB
gcr.io/k8s-minikube/storage-provisioner v5      ba04bb24b957  19 months ago  29MB
<none>              <none>   365ec60129c5  4 years ago   95.4MB
mohamedsirajachabbak@MBP-de-Siraj ~ %
```

Nous allons créer l'image docker de notre application spring boot.

Changez votre repertoire courant et mettez vous dans le repertoire contenant votre application spring boot, ensuite lancer un build de votre image docker : `$ docker build -t backend:1.0` .

```
[INFO] Replacing main artifact with repackaged archive
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 23.707 s
[INFO] Finished at: 2022-11-14T21:37:45Z
[INFO] -----
Removing intermediate container da131faee224
-> 09e45a0d5af3
Step 7/10 : FROM openjdk:11-jre
11-jre: Pulling from library/openjdk
114ba63dd73a: Pull complete
bc0b8a8acead: Pull complete
a4ea641ee679: Pull complete
f3b7f28ea285: Pull complete
3d45ceb6cdd3: Pull complete
40af590bb61e: Pull complete
Digest: sha256:356949c3125c4f08104745e7ea92bd995da4567634e6599b470d2f972d13e0e2
Status: Downloaded newer image for openjdk:11-jre
-> f22f9e14b908
Step 8/10 : COPY --from=build /workspace/target/*.jar backend.jar
--> 3f9aae26f277
Step 9/10 : EXPOSE 8081
--> Running in 63e1988640a9
Removing intermediate container 63e1988640a9
-> 072bf297c946
Step 10/10 : ENTRYPOINT ["java", "-jar", "backend.jar"]
--> Running in 438a31d2bef1
Removing intermediate container 438a31d2bef1
-> 25dfe1e46205
Successfully built 25dfe1e46205
Successfully tagged backend:1.0
mohamedsirajachabbak@MBP-de-Siraj Flabon-backend %
```

Verifier la creation de l'image :`$ docker images`

DOSSIER PROFESSIONNEL (DP)

```
mohamedsirajachabbak@MBP-de-Siraj Fiabon-backend % docker images
REPOSITORY          TAG      IMAGE ID      CREATED       SIZE
backend            1.0      25dfe1e46205  6 minutes ago  343MB
<none>              <none>   09e45a0d5af3   7 minutes ago  559MB
registry.k8s.io/kube-apiserver    v1.25.3   12dd70322f97  4 weeks ago   123MB
registry.k8s.io/kube-proxy        v1.25.3   bcc74496abfd  4 weeks ago   58MB
registry.k8s.io/kube-controller-manager v1.25.3   cf6c9e4e18a3  4 weeks ago   113MB
registry.k8s.io/kube-scheduler     v1.25.3   13b15fc3e093  4 weeks ago   49.3MB
kubernetesui/dashboard           <none>   20b332c9a70d  8 weeks ago   244MB
openjdk               11-jre   f22f9e14b908  3 months ago  294MB
registry.k8s.io/pause            3.8      4e42fb3c9d90  5 months ago  514kB
registry.k8s.io/etcld            3.5.4-0   8e041a3b0ba8  5 months ago  179MB
kubernetesui/metrics-scrapers    <none>   a422e0e98235  5 months ago  42.3MB
registry.k8s.io/coredns/coredns  v1.9.3   b19406328e70  5 months ago  47.7MB
k8s.gcr.io/pause                3.6      7d46a07936af  14 months ago 484kB
maven                  3.6.3-jdk-11-slim 204896d0cfdc  19 months ago  431MB
gcr.io/k8s-minikube/storage-provisioner v5      ba04bb24b957  19 months ago  29MB
mohamedsirajachabbak@MBP-de-Siraj Fiabon-backend %
```

Maintenant que l'image Docker est créée, nous pouvons la déployer sur le cluster Kubernetes

Déploiement avec kubernetes

Création du fichier de déploiement Kubernetes :

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: fiabon-backend
spec:
  selector:
    matchLabels:
      app: fiabon-backend
  replicas: 2
  template:
    metadata:
      labels:
        app: fiabon-backend
    spec:
      containers:
        - name: fiabon-backend
          image: backend.jar:latest
          imagePullPolicy: IfNotPresent
          ports:
            - containerPort: 8081
```

Explications :

apiVersion : décrit la version du serveur API de K8S *kind* : le type d'objet K8s que nous utiliserons pour cette spécification *metadata* : décrit les informations sur l'application *replicas* : décrit le nombre de pods que nous devons exécuter pour la même application. *containers* : décrit les spécifications du container comme le nom, l'image et le port exposé.

Maintenant que nous avons créé le fichier de déploiement Kubernetes, nous pouvons le déployer sur le cluster : *kubectl apply -f deployment.yaml*

Verifier l'état de déploiement : \$ *kubectl get deployments*

DOSSIER PROFESSIONNEL (DP)

```
registry.k8s.io/kube-controller-manager v1.25.3 cf6c9e4e18a3 4 weeks ago 113MB
registry.k8s.io/kube-scheduler v1.25.3 13b15fc3e093 4 weeks ago 49.3MB
kubernetesui/dashboard <none> 20b332c9a70d 8 weeks ago 244MB
openjdk 11-jre f22f9e14b908 3 months ago 294MB
registry.k8s.io/pause 3.8 4e42fb3c9d90 5 months ago 514kB
registry.k8s.io/etc 3.5.4-0 8e041a3b0ba8 5 months ago 179MB
kubernetesui/metrics-scrapers <none> a422e0e98235 5 months ago 42.3MB
registry.k8s.io/coredns/coredns v1.9.3 b19406328e70 5 months ago 47.7MB
k8s.gcr.io/pause 3.6 7d46a07936af 14 months ago 484kB
maven 3.6.3-jdk-11-slim 204896d0cfdc 19 months ago 431MB
gcr.io/k8s-minikube/storage-provisioner v5 ba04bb24b957 19 months ago 29MB
mohamedsirajachabbak@MBP-de-Siraj Fiabon-backend % kubectl apply -f deployment.yaml

deployment.apps/fiabon-backend created
mohamedsirajachabbak@MBP-de-Siraj Fiabon-backend % kubectl get deployments
NAME READY UP-TO-DATE AVAILABLE AGE
fiabon-backend 2/2 2 2 7s
mohamedsirajachabbak@MBP-de-Siraj Fiabon-backend %
```

Ensuite, puisque nous avons mentionné deux répliques, Kubernetes va créer deux pods (instances) pour notre application.

Verifier le status de nos deux pods : \$ kubectl get pods

```
registry.k8s.io/pause 3.8 4e42fb3c9d90 5 months ago 514kB
registry.k8s.io/etc 3.5.4-0 8e041a3b0ba8 5 months ago 179MB
kubernetesui/metrics-scrapers <none> a422e0e98235 5 months ago 42.3MB
registry.k8s.io/coredns/coredns v1.9.3 b19406328e70 5 months ago 47.7MB
k8s.gcr.io/pause 3.6 7d46a07936af 14 months ago 484kB
maven 3.6.3-jdk-11-slim 204896d0cfdc 19 months ago 431MB
gcr.io/k8s-minikube/storage-provisioner v5 ba04bb24b957 19 months ago 29MB
mohamedsirajachabbak@MBP-de-Siraj Fiabon-backend % kubectl apply -f deployment.yaml

deployment.apps/fiabon-backend created
mohamedsirajachabbak@MBP-de-Siraj Fiabon-backend % kubectl get deployments
NAME READY UP-TO-DATE AVAILABLE AGE
fiabon-backend 2/2 2 2 7s
mohamedsirajachabbak@MBP-de-Siraj Fiabon-backend % kubectl get pods
NAME READY STATUS RESTARTS AGE
fiabon-backend-7cf85c9f4b-4k5ls 1/1 Running 0 5m3s
fiabon-backend-7cf85c9f4b-5rplk 1/1 Running 0 5m3s
mohamedsirajachabbak@MBP-de-Siraj Fiabon-backend %
```

Nous pouvons voir ici que le statut est en “Running”. Verifier les logs de chaque pods :

\$ kubectl logs fiabon-backend-7cf85c9f4b-4k5ls kubectl logs fiabon-backend-7cf85c9f4b-5rplk

```
Fiabon-backend-7cf85c9f4b-5rplk 1/1 Running 0 5m3s
mohamedsirajachabbak@MBP-de-Siraj Fiabon-backend % kubectl logs fiabon-backend-7cf85c9f4b-4k5ls
:
:
:: Spring Boot ::   (\v2.1.8.RELEASE)
:
:
2022-11-14 21:46:39.195 INFO 1 --- [       PID 1 </backend.jar started by root in /]      main] Authentication.MainApplication      : Starting MainApplication v0.0.1-SNAPSHOT on fiabon-backend-7cf85c9f4b-4k5ls with PID 1
2022-11-14 21:46:39.201 INFO 1 --- [       PID 1 </backend.jar started by root in /]      main] Authentication.MainApplication      : No active profile set, falling back to default profiles: default
2022-11-14 21:46:42.486 INFO 1 --- [       PID 1 </backend.jar started by root in /]      main] org.apache.catalina.core.StandardService : Starting service [Tomcat]
2022-11-14 21:46:42.486 INFO 1 --- [       PID 1 </backend.jar started by root in /]      main] org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.0.45]
2022-11-14 21:46:42.487 INFO 1 --- [       PID 1 </backend.jar started by root in /]      main] o.a.catalina.core.StandardHost : Initializing Spring embedded WebApplicationContext
2022-11-14 21:46:42.487 INFO 1 --- [       PID 1 </backend.jar started by root in /]      main] o.s.web.context.ContextLoader      : Root WebApplicationContext: starting
2022-11-14 21:46:44.131 INFO 1 --- [       PID 1 </backend.jar started by root in /]      main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Start completed.
2022-11-14 21:46:44.168 INFO 1 --- [       PID 1 </backend.jar started by root in /]      main] o.hibernate.jpa.internal.util.LogHelper : HHH000204: Processing PersistenceUnitInfo [
    name: default
    ...]
2022-11-14 21:46:44.268 INFO 1 --- [       PID 1 </backend.jar started by root in /]      main] org.hibernate.Version      : HHH000412: Hibernate Core {5.3.11.Final}
2022-11-14 21:46:44.276 INFO 1 --- [       PID 1 </backend.jar started by root in /]      main] org.hibernate.cfg.Environment : HHH000206: hibernate.properties not found
2022-11-14 21:46:44.498 INFO 1 --- [       PID 1 </backend.jar started by root in /]      main] o.hibernate.annotations.common.Version : HCANN000001: Hibernate Commons Annotations {5.0.4.Final}
2022-11-14 21:46:44.908 INFO 1 --- [       PID 1 </backend.jar started by root in /]      main] org.hibernate.dialect.Dialect : HHH000404: Using dialect: org.hibernate.dialect.PostgreSQLDialect
2022-11-14 21:46:44.908 INFO 1 --- [       PID 1 </backend.jar started by root in /]      main] o.h.e.j.e.i.LobCreatorBuilderImpl : HHH000421: Disabling contextual LOB creation as hibernate.jdbc.lob.non_contextual_creation is true
2022-11-14 21:46:50.242 INFO 1 --- [       PID 1 </backend.jar started by root in /]      main] org.hibernate.type.BasicTypeRegistry : HHH000270: Type registration [java.util.UUID] overrides previous: org.hibernate.type.UUIDType
2022-11-14 21:46:53.382 INFO 1 --- [       PID 1 </backend.jar started by root in /]      main] Authentication.MainApplication      : Started MainApplication in 15.008 seconds (JVM running for 15.531)
mohamedsirajachabbak@MBP-de-Siraj Fiabon-backend % kubectl logs fiabon-backend-7cf85c9f4b-5rplk
:
:
:: Spring Boot ::   (\v2.1.8.RELEASE)
:
:
2022-11-14 21:46:39.267 INFO 1 --- [       PID 1 </backend.jar started by root in /]      main] Authentication.MainApplication      : Starting MainApplication v0.0.1-SNAPSHOT on fiabon-backend-7cf85c9f4b-5rplk with PID 1
mohamedsirajachabbak@MBP-de-Siraj Fiabon-backend %
```

DOSSIER PROFESSIONNEL (DP)

L'application est déployée dans notre cluster K8S, mais comme vous pouvez le constater l'application tourne sur deux pods différents et deux instances différentes.

Donc la requête va tomber sur quelle instance ? le pod1 (fiabon-backend-7cf85c9f4b-4k5ls) ou le pod 2 (fiabon-backend-7cf85c9f4b-5rplk) 😱 ?

Pour résoudre ce problème, nous avons besoin d'un mécanisme de répartition de la charge un **load balancer**. Configuration de notre load balancer *loadBalancer.yaml* :

```
apiVersion: v1
kind: Service
metadata:
  name: fiabon-backend
spec:
  selector:
    app: fiabon-backend
  ports:
    - protocol: "TCP"
      port: 8081 # Le port sur lequel le service est exécuté dans le cluster.
      targetPort: 8081 # Le port exposé par le service
      type: NodePort # type de service
```

Maintenant que nous avons créé le fichier de service, exposons notre application à l'extérieur du cluster k8s en utilisant la commande ci-dessous : \$ *kubectl apply -f loadbalancer.yaml*

Vérifier le statut de notre loadbalancer : \$ *kubectl get service* *kubectl get nodes -o wide minikube ip*

```
mohamedsirajachabbak@MBP-de-Siraj Fiabon-backend % kubectl apply -f loadbalancer.yaml
service/fiabon-backend created
mohamedsirajachabbak@MBP-de-Siraj Fiabon-backend % kubectl get service
NAME        TYPE        CLUSTER-IP      EXTERNAL-IP      PORT(S)        AGE
fiabon-backend   NodePort    10.105.20.159 <none>        8081:31877/TCP  11s
kubernetes     ClusterIP   10.96.0.1     <none>        443/TCP       31m
mohamedsirajachabbak@MBP-de-Siraj Fiabon-backend % kubectl get nodes -o wide
NAME      STATUS   ROLES      AGE      VERSION   INTERNAL-IP      EXTERNAL-IP      OS-IMAGE      KERNEL-VERSION      CONTAINER-RUNTIME
minikube  Ready    control-plane  31m    v1.25.3   192.168.49.2   <none>        Ubuntu 20.04.5 LTS  5.15.49-linuxkit  docker://20.10.20
mohamedsirajachabbak@MBP-de-Siraj Fiabon-backend %
```

Vérifier le statut de notre kubernetes : \$Minikube dashboard

The screenshot shows the Kubernetes dashboard interface. On the left, a sidebar lists various resources: Workloads (Cron Jobs, Daemon Sets, Deployments, Jobs, Pods, Replica Sets, Replication Controllers, Stateful Sets), Service (Ingresses, Ingress Classes, Services), and Config and Storage (Config Maps, Persistent Volume Claims, Secrets). The main area displays the status of workloads, specifically Deployments, Pods, and Replica Sets. Each category has a summary chart and a detailed table below it. For Deployments, there are three green circles labeled 'Running: 1'. For Pods, there are two green circles labeled 'Running: 2'. For Replica Sets, there is one green circle labeled 'Running: 1'. Below the charts, the 'Deployments' table shows a single entry: 'fiabon-backend' with 'backend:1.0' as the image, 2 / 2 pods, and a creation date of 26 minutes ago. The 'Pods' table shows two entries: 'fiabon-backend-7cf85c9f4b-4k5ls' and 'fiabon-backend-7cf85c9f4b-5rplk'.

DOSSIER PROFESSIONNEL (DP)

Ça y est, notre application est sur kubernetes

Il est important de noter que ce processus de déploiement de base ne couvre que les aspects les plus simples de Kubernetes. Il existe de nombreuses autres fonctionnalités et options que vous pouvez utiliser pour configurer et gérer votre application sur Kubernetes, y compris la mise à l'échelle automatique, la gestion des dépendances, la création de déploiements rollback et bien plus encore.

2. Précisez les moyens utilisés :

- Microsoft Azure (La plateforme de cloud Azure).
- Machine Virtuel chez Cloud Azure :Ubuntu (Système d'exploitation)
- Un émulateur de terminal MobaXterm.
- Docker
- Kubernetes

3. Avec qui avez-vous travaillé ?

- Travail personnel

4. Contexte

Nom de l'entreprise, organisme ou association ► *Cliquez ici pour taper du texte.*

Chantier, atelier, service ► *Cliquez ici pour taper du texte.*

Période d'exercice ► Du : **20/11/2023** au : **24/11/2023**

5. Informations complémentaires (*facultatif*)

DOSSIER PROFESSIONNEL (DP)

Activité-type 3 Superviser les services déployés

Exemple n°1 ▶ Définir et mettre en place des statistiques de services

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Définition et Mise en Place de Statistiques de Services pour la Surveillance avec Nagios

Introduction : Dans le paysage informatique dynamique d'aujourd'hui, la surveillance efficace des services est cruciale pour garantir la fiabilité et les performances de l'infrastructure. Ce document décrit les étapes pour définir et mettre en place des statistiques de services à l'aide de Nagios, une solution de surveillance open source largement utilisée.

1. Définition des Statistiques de Services : Les statistiques de services englobent un ensemble de mesures qui fournissent des informations sur la santé et les performances de services spécifiques au sein d'un environnement informatique. Ces mesures peuvent inclure la disponibilité, le temps de réponse, le débit et les taux d'erreur, entre autres.

2. Identification des Services Clés : Commencez par identifier les services critiques au sein de votre infrastructure qui nécessitent une surveillance. Il peut s'agir de serveurs web, de bases de données ou d'applications personnalisées. Définissez clairement la portée de chaque service et les indicateurs clés de performance (KPI) pertinents pour leur bon fonctionnement.

3. Configuration de Nagios :

- Installation : Commencez par installer Nagios sur un serveur dédié. Suivez le guide d'installation officiel de Nagios pour votre système d'exploitation spécifique.
- Définition de Service : Utilisez les configurations de Nagios pour définir chaque service surveillé. Cela implique la création de définitions de service spécifiant les vérifications de service, les paramètres de notification et les données de performance à collecter.

Exemple de Définition de Service (services.cfg) :

```
define service {
    host_name      votre_serveur
    service_description HTTP
    check_command   check_http
    use            generic-service
    notification_interval 30
}
```

4. Collecte de Données de Performance : Activez la collecte de données de performance pour chaque service. Nagios prend en charge la sortie de données de performance à partir des vérifications de service, vous permettant de stocker et de visualiser les mesures dans le temps. Ces données peuvent être utilisées pour l'analyse des tendances et la planification des capacités.

DOSSIER PROFESSIONNEL (DP)

Exemple de Configuration de Données de Performance (nagios.cfg) :

```
enable_performance_data=1
```

5. Visualisation et Rapports : Intégrez Nagios avec des outils de visualisation ou des solutions de reporting pour présenter les statistiques de services de manière claire et exploitable. Des outils tels que Grafana ou Nagiosgraph peuvent améliorer la représentation graphique des tendances de performance.

6. Notifications et Alertes : Configurez Nagios pour envoyer des notifications et des alertes en fonction de seuils prédéfinis pour chaque service. Cela garantit que l'équipe opérationnelle est informée rapidement de toute déviation par rapport aux niveaux de performance attendus.

Conclusion : La définition et la mise en place de statistiques de services avec Nagios constituent une approche proactive pour maintenir un environnement informatique robuste. Revoyez régulièrement et mettez à jour votre stratégie de surveillance pour vous adapter aux besoins et aux évolutions technologiques.

Voici une liste de services que on peut mettre en place pour la supervision avec Nagios :

1. HTTP (Web Server) :

- Vérifiez la disponibilité du serveur web.
- Surveillez le temps de réponse des requêtes HTTP.
- Vérifiez les certificats SSL.

2. SMTP (Email Server) :

- Contrôlez la disponibilité du serveur de messagerie.
- Vérifiez la réception des emails.

3. DNS (Domain Name System) :

- Surveillez la résolution de noms de domaine.
- Vérifiez la disponibilité des serveurs DNS.

4. FTP (File Transfer Protocol) :

- Contrôlez la disponibilité du serveur FTP.
- Surveillez les transferts de fichiers.

5. SSH (Secure Shell) :

- Vérifiez la disponibilité des services SSH.
- Contrôlez l'accès aux serveurs via SSH.

6. Database Servers (MySQL, PostgreSQL, etc.) :

- Surveillez la disponibilité des serveurs de bases de données.
- Contrôlez les requêtes SQL et les performances.

7. SNMP (Simple Network Management Protocol) :

- Surveillez les périphériques réseau compatibles SNMP.
- Collectez des informations sur la charge du réseau.

8. Custom Applications :

- Adaptez Nagios pour surveiller des applications spécifiques à votre environnement.
- Contrôlez les performances et la disponibilité des applications métier.

9. System Resources (CPU, Memory, Disk) :

- Surveillez l'utilisation du processeur, de la mémoire et de l'espace disque.
- Contrôlez les performances générales du système.

10. Network Services (DHCP, LDAP, NTP) :

- Vérifiez la disponibilité des services réseau essentiels.
- Surveillez les temps de réponse et la synchronisation.

11. Security Services (Intrusion Detection Systems, Log Analysis) :

DOSSIER PROFESSIONNEL (DP)

- Surveillez les événements de sécurité.
- Contrôlez les journaux d'activité et les alertes d'intrusion.

12. Virtualization Platforms (VMware, KVM, Hyper-V) :

- Surveillez la disponibilité des serveurs de virtualisation.
- Contrôlez les performances des machines virtuelles.

13. Backup Services :

- Vérifiez l'état des sauvegardes.
- Surveillez les processus de sauvegarde.

14. VoIP (Voice over IP) :

- Contrôlez la disponibilité des serveurs VoIP.
- Surveillez la qualité des appels.

15. Containers (Docker, Kubernetes) :

- Surveillez l'état des conteneurs.
- Contrôlez les performances des clusters de conteneurs.

2. Précisez les moyens utilisés :

- Microsoft Azure (La plateforme de cloud Azure).
- MobaXterm.
- Superviseur : Nagios

3. Avec qui avez-vous travaillé ?

- Travail individuel

4. Contexte

Nom de l'entreprise, organisme ou association ► *Cliquez ici pour taper du texte.*

Chantier, atelier, service ► *Cliquez ici pour taper du texte.*

Période d'exercice ► Du : **28/11/2023** au : **05/12/2023**

5. Informations complémentaires (facultatif)

Activité-type 3 Superviser les services déployés

Exemple n° 2 ▶ Exploiter une solution de supervision

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Nagios est un logiciel qui permet de superviser un système d'information. Il est considéré comme étant la référence des solutions de supervision open source. Il dispose de nombreuses fonctions telles que l'héritage multiple, les dépendances, l'escalade de notifications, les Template de services et d'hôtes, le support des surveillances actives et passives, etc. L'interface web est la partie graphique, via un serveur web tel que Apache, et qui va permettre à l'administrateur d'avoir une vue d'ensemble de son réseau, de visualiser la supervision des équipements et de produire des rapports d'activités.

J'ai créé deux serveur Ubuntu. Un serveur nommé « nagios1 », c'est le serveur Nagios core , c'est le localhost et le deuxième serveur nommé « nagios5 » est le serveur client qui contient le Nagios nrpe , le serveur Nagios core va surveiller le deuxième serveur Nagios nrpe

Télécharger Nagios Core depuis leur répertoire Github sur le premier serveur :

- sudo apt-get update
- sudo apt-get install -y autoconf gcc libc6 make wget unzip apache2 php libapache2-mod-php7.4 libgd-dev
- sudo apt-get install openssl libssl-dev
- cd /tmp
- wget -O nagioscore.tar.gz <https://github.com/NagiosEnterprises/nagioscore/archive/nagios-4.4.14.tar.gz>
- tar xzf nagioscore.tar.gz
- cd /tmp/nagioscore-nagios-4.4.14/
- sudo ./configure --with-httpd-conf=/etc/apache2/sites-enabled
- sudo make all

Créer un utilisateur et un groupe . Cela crée l'utilisateur et le groupe Nagios. L'utilisateur www-data est également ajouté au groupe nagios.

-
- sudo make install-groups-users
- sudo usermod -a -G nagios www-data

Installer les binaires : Cette étape installe les fichiers binaires, les CGI et les fichiers HTML.

- sudo make install

Installer le service/ Daemon : Cela installe les fichiers de service ou de démon et les configure également pour qu'ils démarrent au démarrage.

- sudo make install-daemoninit

Les informations sur le démarrage et l'arrêt des services seront expliquées plus loin. Installer le mode de commande. Ceci installe et configure le fichier de commande externe.

- sudo make install-commandmode
- sudo make install-config

Installer les fichiers de configuration Apache. Cela installe les fichiers de configuration du serveur Web Apache et configure les paramètres Apache.

DOSSIER PROFESSIONNEL (DP)

- sudo make install-webconf
- sudo a2enmod rewrite
- sudo a2enmod cgi

Configurer le pare-feu. Autoriser le trafic entrant sur le port 80 sur le pare-feu local afin de pouvoir accéder à l'interface Web Nagios Core.

- sudo ufw allow Apache
- sudo ufw reload

Créer un compte utilisateur nagiosadmin. Créeira un compte utilisateur appelé nagiosadmin et vous serez invité à fournir un mot de passe pour le compte.

- sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
- sudo systemctl restart apache2.service
- sudo systemctl start nagios.service

```
zemni@nagios1:~$ sudo systemctl status nagios
● nagios.service - Nagios Core 4.4.6
   Loaded: loaded (/lib/systemd/system/nagios.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2023-11-30 19:17:18 UTC; 3min 11s ago
     Docs: https://www.nagios.org/documentation
 Process: 2825 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
 Process: 2826 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
 Main PID: 2827 (nagios)
   Tasks: 6 (limit: 970)
  Memory: 2.0M
    CPU: 832ms
   CGroup: /system.slice/nagios.service
           └─2827 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
               ├─2828 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
               ├─2829 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
               ├─2830 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
               ├─2831 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
               └─2832 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

nov. 30 19:17:18 nagios1 nagios[2827]: qh: Socket '/usr/local/nagios/var/rw/nagios.qh' successfully initialized
nov. 30 19:17:18 nagios1 nagios[2827]: qh: core query handler registered
nov. 30 19:17:18 nagios1 nagios[2827]: qh: echo service query handler registered
nov. 30 19:17:18 nagios1 nagios[2827]: qh: help for the query handler registered
nov. 30 19:17:18 nagios1 nagios[2827]: wproc: Successfully registered manager as @wproc with query handler
nov. 30 19:17:18 nagios1 nagios[2827]: wproc: Registry request: name=Core Worker 2830;pid=2830
nov. 30 19:17:18 nagios1 nagios[2827]: wproc: Registry request: name=Core Worker 2829;pid=2829
nov. 30 19:17:18 nagios1 nagios[2827]: wproc: Registry request: name=Core Worker 2828;pid=2828
nov. 30 19:17:18 nagios1 nagios[2827]: wproc: Registry request: name=Core Worker 2831;pid=2831
nov. 30 19:17:19 nagios1 nagios[2827]: Successfully launched command file worker with pid 2832
```

① 192.168.163.156/nagios

Se connecter

http://192.168.163.156

Votre connexion à ce site n'est pas privée

Nom d'utilisateur

Mot de passe

Se connecter

[Annuler](#)

DOSSIER PROFESSIONNEL (DP)

The screenshot shows the Nagios Core 4.4.6 web interface. At the top, it displays "Non sécurisé | 192.168.163.156/nagios/". Below this, there are two main status summary boxes: "Host Status Totals" and "Service Status Totals".

Host Status Totals:
Last Updated: Thu Nov 30 19:22:48 UTC 2023
Updated every 90 seconds
Nagios® Core™ 4.4.6 - www.nagios.org
Logged in as nagiosadmin

Service Status Totals:
Ok Warning Unknown Critical Pending
All Problems All Types
8 8 0

Service Status Details For All Hosts:
Limit Results: 100

| Host | Service | Status | Last Check | Duration | Attempt | Status Information |
|-----------------|----------------|----------|---------------------|---------------|---------|--|
| localhost | Current Load | CRITICAL | 11-30-2023 19:20:05 | 0d 4h 37m 43s | 4/4 | (No output on stdout) stderr: execvp(/usr/local/nagios/libexec/check_load...) failed: errno is 2: No such file or directory |
| localhost | Current Users | CRITICAL | 11-30-2023 19:20:43 | 0d 4h 37m 5s | 4/4 | (No output on stdout) stderr: execvp(/usr/local/nagios/libexec/check_users...) failed: errno is 2: No such file or directory |
| HTTP | PING | CRITICAL | 11-30-2023 19:21:20 | 0d 4h 36m 28s | 4/4 | (No output on stdout) stderr: execvp(/usr/local/nagios/libexec/check_http...) failed: errno is 2: No such file or directory |
| HTTP | Root Partition | CRITICAL | 11-30-2023 19:22:35 | 0d 4h 35m 13s | 4/4 | (No output on stdout) stderr: execvp(/usr/local/nagios/libexec/check_disk...) failed: errno is 2: No such file or directory |
| SSH | Swap Usage | CRITICAL | 11-30-2023 19:18:13 | 0d 4h 34m 35s | 4/4 | (No output on stdout) stderr: execvp(/usr/local/nagios/libexec/check_swap...) failed: errno is 2: No such file or directory |
| Total Processes | | CRITICAL | 11-30-2023 19:18:50 | 0d 4h 33m 58s | 4/4 | (No output on stdout) stderr: execvp(/usr/local/nagios/libexec/check_procs...) failed: errno is 2: No such file or directory |
| Uptime | | CRITICAL | 11-30-2023 19:19:30 | 0d 4h 33m 20s | 4/4 | (No output on stdout) stderr: execvp(/usr/local/nagios/libexec/check_procs...) failed: errno is 2: No such file or directory |

Results 1 - 8 of 8 Matching Services

On est toujours sur le serveur 1 « nagios1 ». Nagios est maintenant en cours d'exécution, pour confirmer cela, je connecte à l'interface Web de Nagios. Je connecte avec l'adresse ip « <https://192.168.163.149/nagios> » ou bien <http://core-013.domain.local/nagios>

Un nom d'utilisateur et un mot de passe vous seront demandés. Le nom d'utilisateur est nagiosadmin.

Une fois connecté, l'interface Nagios vous est présentée. Félicitations, vous avez installé Nagios Core.

MAIS ATTENDEZ ...

Actuellement, J'ai uniquement installé le moteur Nagios Core.

Nagios Core a besoin de plugins pour fonctionner correctement. Les étapes suivantes vous guideront dans l'installation des plugins Nagios.

installent la plupart des plugins fournis dans le package Nagios Plugins. Cependant, certains plugins nécessitent d'autres bibliothèques qui ne sont pas incluses dans ces instructions.

Installer les plugins Nagios à partir des sources

- sudo apt-get install -y autoconf gcc libc6 libmcrypt-dev make libssl-dev wget bc gawk dc build-essential snmp libnet-snmp-perl gettext
- cd /tmp
- wget --no-check-certificate -O nagios-plugins.tar.gz <https://github.com/nagios-plugins/nagios-plugins/archive/release-2.4.6.tar.gz>
- tar zxf nagios-plugins.tar.gz
- cd /tmp/nagios-plugins-release-2.4.6/
- sudo ./tools/setup
- sudo ./configure
- sudo make
- sudo make install
- sudo systemctl start nagios.service
- sudo systemctl stop nagios.service
- sudo systemctl restart nagios.service
- sudo systemctl status nagios.service

DOSSIER PROFESSIONNEL (DP)

The screenshot shows the Nagios web interface. On the left, there's a navigation menu with links like General, Home, Documentation, Current Status, and Problems. The main area displays 'Current Network Status' with last updated information and a log-in message. It includes two summary tables: 'Host Status Totals' and 'Service Status Totals'. Below these are detailed tables for 'Service Status Details For All Hosts' and 'Service Status Details For All Services'. A search bar at the bottom is set to 'Quick Search'.

```
zemni@nagios1:~$ cd /usr/local/nagios/ ; ./nagios -v
zemni@nagios1:/usr/local/nagios$ ls -la
total 104
drwxrwxr-x 3 nagios nagios 4096 déc. 4 15:58 .
drwxr-xr-x 8 root root 4096 déc. 2 09:59 ..
-rw-rw-r-- 1 nagios nagios 13719 déc. 4 12:37 cgi.cfg
-rw-r--r-- 1 root root 50 déc. 2 10:03 htpasswd.users
-rw-rw-r-- 1 nagios nagios 46010 déc. 4 14:38 nagios.cfg
-rw-r--r-- 1 nagios nagios 12851 déc. 4 15:13 nrpe.cfg
-rw-r--r-- 1 root root 1024 déc. 4 15:58 .nrpe.cfg.swp
drwxrwxr-x 2 nagios nagios 4096 déc. 4 15:56 objects
-rw-rw---- 1 nagios nagios 1313 déc. 2 09:59 resource.cfg
zemni@nagios1:/usr/local/nagios/etc$
```

```
zemni@nagios1:~$ sudo nano /usr/local/nagios/etc/objects/localhost.cfg
```

```
#####
# Define a host for the local machine
define host {
    use          linux-server      ; Name of host
    host_name    localhost         ; This host definition
    alias        localhost         ; is inherited from
    address     192.168.163.149
}

define host {
    use          linux-server      ; Name of host
    host_name    nagios5          ; This host definition
    alias        MX server         ; is inherited from
    address     192.168.163.155
}
```

DOSSIER PROFESSIONNEL (DP)

```
# Define a service to "ping" the local machine
define service {
    use          local-service      ; Name of
    host_name   localhost,nagios5
    service_description PING
    check_command  check_ping!100.0,20%!500.0,60%
}

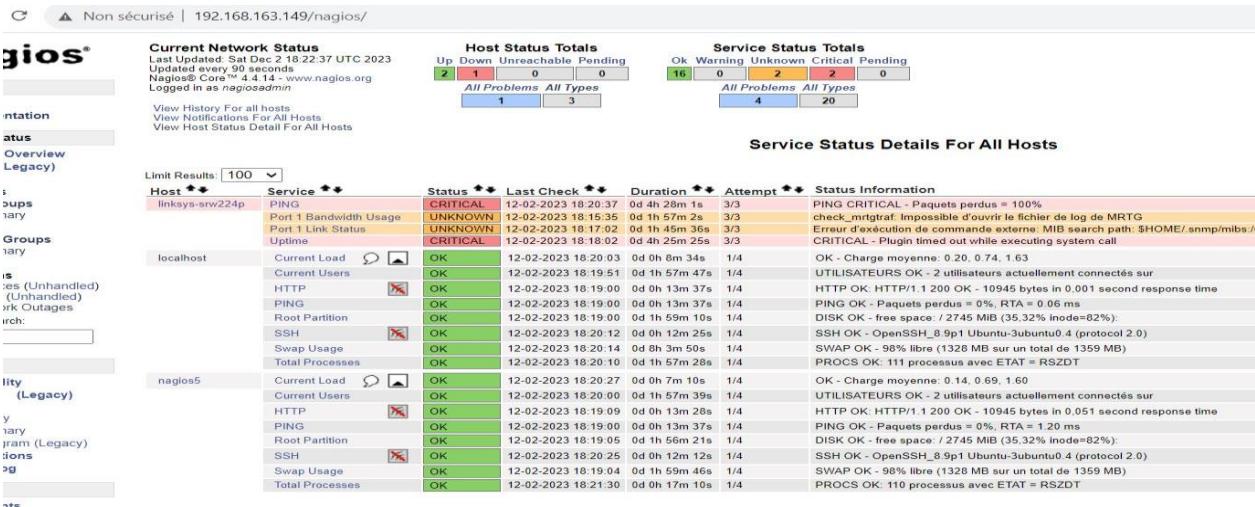
# Define a service to check the disk space of the root partition
# on the local machine. Warning if < 20% free, critical if
# < 10% free space on partition.
define service {
    use          local-service      ; Name of
    host_name   localhost,nagios5
    service_description Root Partition
    check_command  check_local_disk!20%!10%!
}

# Define a service to check the number of currently logged in
# users on the local machine. Warning if > 20 users, critical
# if > 50 users.
define service {
    use          local-service      ; Name of
    host_name   localhost,nagios5
    service_description Current Users
    check_command  check_local_users!20!50
}

# Define a service to check the number of currently running processes
# on the local machine. Warning if > 250 processes, critical if
# > 400 processes.
define service {
    use          local-service      ; Name of service template
    host_name   localhost,nagios5
    service_description Total Processes
    check_command  check_local_procs!250!400!RSZDT
}

# Define a service to check the load on the local machine.
define service {
    use          local-service      ; Name of service template
    host_name   localhost,nagios5
    service_description Current Load
    check_command  check_local_load!5.0,4.0,3.0!10.0,6.0,4.0
}

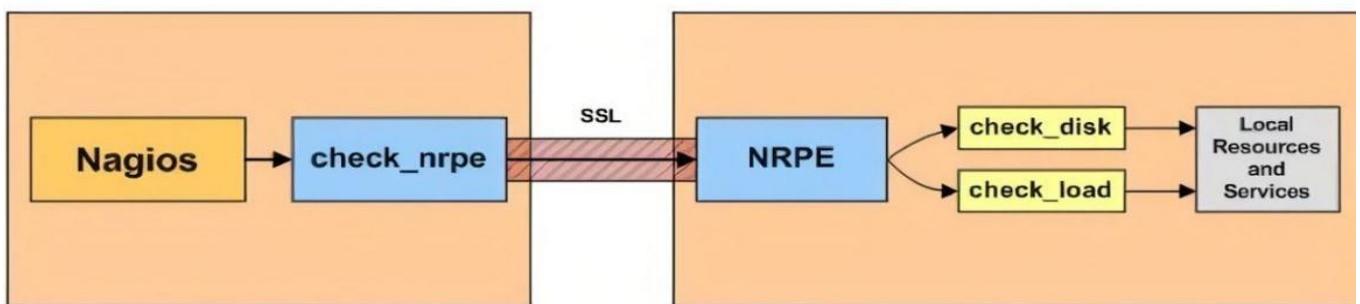
# Define a service to check the swap usage the local machine.
# Critical if less than 10% of swap is free, warning if less than 20%
define service {
    use          local-service      ; Name of service template
    host_name   localhost,nagios5
    service_description Swap Usage
    check_command  check_local_swap!20%!10%
```



The screenshot shows the Nagios web interface with the following sections:

- Current Network Status:** Last Updated: Sat Dec 2 18:22:37 UTC 2023. Shows linksys-srw224p as the only host.
- Host Status Totals:** Up: 2, Down: 1, Unreachable: 0, Pending: 0. All Problems: 1, All Types: 3.
- Service Status Totals:** Ok: 16, Warning: 0, Unknown: 2, Critical: 2, Pending: 0. All Problems: 4, All Types: 20.
- Service Status Details For All Hosts:** A table listing services for hosts linksys-srw224p, localhost, and nagios5. Services include PING, Port 1 Bandwidth Usage, Port 1 Link Status, Uptime, Current Load, Current Users, HTTP, SSH, Swap Usage, and Total Processes. Status information includes error messages like "PING CRITICAL - Paquets perdus = 100%" and "SSH OK - OpenSSH_8.9p1 Ubuntu-3ubuntu0.4 (protocol 2.0)".

Nagios NRPE est l'acronyme pour Remote Plugin Executor. NRPE sera installé sur le serveur Nagios Core ainsi que sur les serveurs que vous surveillerez. Il est important de faire la différence entre ses deux installations. Celle installé sur le serveur Nagios Core est un plugin "check_nrpe".



Monitoring Host

Remote Linux/Unix Host

Apres la création de deux serveur sur le cloud : on a le serveur nommé « nagios4 » avec l'adresse ip : 192.168.163.162 et le serveur client nommé « nagios5 » avec l'adresse ip : 192.168.163.163

DOSSIER PROFESSIONNEL (DP)

Je fais ses étape pour mettre on place l'installation de nagios nrpe sur un serveur distant .

- sudo apt-get update
- sudo apt-get install -y autoconf automake gcc libc6 libmcrypt-dev make libssl-dev wget openssl
- cd /tmp
- wget --no-check-certificate -O nrpe.tar.gz https://github.com/NagiosEnterprises/nrpe/archive/nrpe-4.1.0.tar.gz

```
zemni@nagios4:/tmp$ ls
Gm5gcx0g
GmhH9IwT
nagioscore-nagios-4.4.14
nagioscore.tar.gz
nagios-plugins-release-2.4.6
nagios-plugins.tar.gz
nrpe-4.1.0.tar.gz
snap-private-tmp
systemd-private-eb60a62395734c3a95a5071b6cc4c278-apache2.service-Xjrpa7
systemd-private-eb60a62395734c3a95a5071b6cc4c278-ModemManager.service-vFjSSU
systemd-private-eb60a62395734c3a95a5071b6cc4c278-systemd-logind.service-BP1sBw
systemd-private-eb60a62395734c3a95a5071b6cc4c278-systemd-resolved.service-oCpuBs
systemd-private-eb60a62395734c3a95a5071b6cc4c278-systemd-timesyncd.service-hdgAJl
vmware-root_10353-1949248498
vmware-root_724-2965906890
vmware-root_9862-962248483
zemni@nagios4:/tmp$
```

On va décompresser le fichier zip nrpe-4.1.0.tar.gz

- cd /tmp
- tar xzf nrpe-4.1.0.tar.gz
- cd /tmp/nrpe-nrpe-4.1.0/
- sudo ./configure --enable-command-args --with-ssl-lib=/usr/lib/x86_64-linux-gnu/
- sudo make all
- sudo make install
- sudo make install-config

Cette configuration que dans la machine serveur

- sudo sh -c "echo >> /etc/services"
- sudo sh -c "sudo echo '# Nagios services' >> /etc/services"
- sudo sh -c "sudo echo 'nrpe 5666/tcp' >> /etc/services"
- sudo cp startup/default-service /etc/systemd/system/nrpe.service
- sudo chmod 644 /etc/systemd/system/nrpe.service

```
zemni@nagios4:/usr/local/nagios/etc/objects$ sudo nano commands.cfg
```

```
define command{
    command_name check_nrpe
    command_line $USER1$/check_nrpe -H $HOSTADDRESS$ -c $ARG1$}
```

Passant au serveur client « nagios5 » :

On fait les même étape d'installation de nrpe et on installe le plugin de nagios

- sudo apt-get install -y autoconf gcc libc6 libmcrypt-dev make libssl-dev wget bc gawk dc build-essential

DOSSIER PROFESSIONNEL (DP)

```
snmp libnet-snmp-perl gettext
- cd /tmp
- wget --no-check-certificate -O nagios-plugins.tar.gz https://github.com/nagios-plugins/nagios-plugins/archive/release-2.2.1.tar.gz
- tar zxf nagios-plugins.tar.gz
- cd /tmp/nagios-plugins-release-2.2.1/
- sudo ./tools/setup
- sudo ./configure
- sudo make
- sudo make install
```

```
GNU nano 6.2                               /usr/local/nagios/etc/nrpe.cfg

allowed_hosts=127.0.0.1,::1,192.168.163.162
server_address=0.0.0.0

# COMMAND ARGUMENT PROCESSING
# This option determines whether or not the NRPE daemon will allow
# to specify arguments to commands that are executed. This option
# if the daemon was configured with the --enable-command-args con
# option.
#
# *** ENABLING THIS OPTION IS A SECURITY RISK! ***
# Read the SECURITY file for information on some of the security i
# of enabling this variable.
#
# Values: 0=do not allow arguments, 1=allow command arguments
dont_blame_nrpe=1
```

```
GNU nano 6.2                               /usr/local/nagios/etc/nrpe.cfg

command[check_users]=/usr/local/nagios/libexec/check_users -w 5 -c 10
command[check_load]=/usr/local/nagios/libexec/check_load -r -w .15,.10,.05 -c .30,.25,.20
command[check_sda1]=/usr/local/nagios/libexec/check_disk -w 20% -c 10% -p /dev/sda1
command[check_zombie_procs]=/usr/local/nagios/libexec/check_procs -w 5 -c 10 -s Z
command[check_total_procs]=/usr/local/nagios/libexec/check_procs -w 150 -c 200
```

On est sur le serveur 1 : on tape cette commande pour vérifier si il y a une connexion avec le client

```
zemni@nagios4:~$ /usr/local/nagios/libexec/check_nrpe -H 192.168.163.163
NRPE v4.1.0
```

Et dans le fichier /usr/local/nagios/etc/nagios.cfg on ajoute le fichier de configuration :

```
# You can specify individual object config files as shown below:
cfg_file=/usr/local/nagios/etc/objects/commands.cfg
cfg_file=/usr/local/nagios/etc/objects/contacts.cfg
cfg_file=/usr/local/nagios/etc/objects/timeperiods.cfg
cfg_file=/usr/local/nagios/etc/objects/templates.cfg
cfg_file=/usr/local/nagios/etc/objects/nagios5.cfg
# Definitions for monitoring the local (Linux) host
cfg_file=/usr/local/nagios/etc/objects/localhost.cfg
```

Dans le fichier /usr/local/nagios/etc/objects/nagios5.cfg

DOSSIER PROFESSIONNEL (DP)

```

define host{
    use          linux-server
    host_name    nagios5
    alias        nagios-client
    address      192.168.163.163
}

define service {
    use          generic-service
    host_name    nagios5
    service_description  Users
    check_command  check_nrpe!check_users
}

define service {
    use          generic-service
    host_name    nagios5
    service_description  CPU load
    check_command  check_nrpe!check_load
}

define service {
    use          generic-service
    host_name    nagios5
    service_description  Disk /
    check_command  check_nrpe!check_sda1
}

define service {
    use          generic-service
    host_name    nagios5
    service_description  Zombie Processes
    check_command  check_nrpe!check_zombie_procs
}

define service {
    use          generic-service
    host_name    nagios5
}

```

⚠ Non sécurisé | 192.168.163.162/nagios/

Current Network Status
 Last Updated: Wed Dec 6 17:26:41 UTC 2023
 Updated every 90 seconds
 Nagios® Core™ 4.4.14 - www.nagios.org
 Logged in as nagiosadmin

Host Status Totals

| Up | Down | Unreachable | Pending |
|----|------|-------------|---------|
| 2 | 0 | 0 | 0 |

| All Problems | All Types |
|--------------|-----------|
| 0 | 2 |

Service Status Totals

| Ok | Warning | Unknown | Critical | Pending |
|----|---------|---------|----------|---------|
| 11 | 1 | 0 | 1 | 0 |

| All Problems | All Types |
|--------------|-----------|
| 2 | 13 |

Service Status Details For All Hosts

| Host | Service | Status | Last Check | Duration | Attempt | Status Information |
|-----------|------------------|----------|---------------------|--------------|---------|---|
| localhost | Current Load | OK | 12-06-2023 17:21:37 | 0d 3h 2m 44s | 1/4 | OK - Charge moyenne: 0.00, 0.00, 0.00 |
| | Current Users | OK | 12-06-2023 17:22:41 | 0d 3h 2m 6s | 1/4 | UTILISATEURS OK - 3 utilisateurs actuellement connectés sur |
| | HTTP | OK | 12-06-2023 17:23:45 | 0d 3h 6m 29s | 1/4 | HTTP OK: HTTP/1.1 200 OK - 10945 bytes in 0.001 second res |
| | PING | OK | 12-06-2023 17:25:50 | 0d 3h 5m 51s | 1/4 | PING OK - Paquets perdus = 0%, RTA = 0.13 ms |
| | Root Partition | OK | 12-06-2023 17:24:49 | 0d 3h 5m 14s | 1/4 | DISK OK - free space: / 2892 MB (37.21% inode=82%) |
| | SSH | OK | 12-06-2023 17:22:03 | 0d 3h 4m 36s | 1/4 | SSH OK - OpenSSH_8.9p1 Ubuntu-3ubuntu0.3 (protocol 2.0) |
| nagios5 | Swap Usage | OK | 12-06-2023 17:25:10 | 0d 3h 3m 59s | 1/4 | SWAP OK - 100% libre (1378 MB sur un total de 1389 MB) |
| | Total Processes | OK | 12-06-2023 17:25:50 | 0d 3h 3m 21s | 1/4 | PROCS OK: 108 processus avec ETAT = RSZDT |
| | CPU load | WARNING | 12-06-2023 17:26:09 | 0d 0h 0m 32s | 3/3 | WARNING - load average: 0.09, 0.13, 0.15 |
| localhost | Disk / | OK | 12-06-2023 17:23:13 | 0d 0h 6m 7s+ | 1/3 | DISK OK - free space: /dev 404 MB (100.00% inode=100%) |
| | Total Processes | CRITICAL | 12-06-2023 17:26:17 | 0d 0h 2m 24s | 2/3 | PROCS CRITICAL: 207 processus |
| | Users | OK | 12-06-2023 17:25:21 | 0d 0h 6m 7s+ | 1/3 | USERS OK - 3 users currently logged in |
| | Zombie Processes | OK | 12-06-2023 17:26:25 | 0d 0h 6m 7s+ | 1/3 | PROCS OK: 0 processus avec STATE = Z |



DOSSIER PROFESSIONNEL (DP)

2. Précisez les moyens utilisés :

- Microsoft Azure (La plateforme de cloud Azure).
- MobaXterm.
- Superviseur : Nagios Core et Nagios nrpe

3. Avec qui avez-vous travaillé ?

- Travail individuel

4. Contexte

Nom de l'entreprise, organisme ou association ► *Cliquez ici pour taper du texte.*

Chantier, atelier, service ► *Cliquez ici pour taper du texte.*

Période d'exercice ► Du : **28/11/2023** au : **05/12/2023**

5. Informations complémentaires (*facultatif*)

DOSSIER PROFESSIONNEL (DP)

Activité-type 3 Superviser les services déployés

Exemple n° 3 ▶ Echanger sur des réseaux professionnels éventuellement en anglais

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

How to Engage in Professional DevOps Networking:

Engaging in professional DevOps networks can be an excellent opportunity to expand your knowledge, share experiences, and establish valuable contacts in the field. Here are some steps you can take to initiate conversations on professional DevOps networks:

1. **Identify DevOps Networking Platforms:** Look for online platforms specifically dedicated to DevOps, such as LinkedIn, Twitter, Reddit, or specialized forums like Stack Overflow. Sign up on these platforms if you haven't already.
2. **Optimize Your Profile:** Update your profile on these platforms to reflect your experience, skills, and interest in DevOps. Use relevant keywords to make your profile more accessible to people searching for DevOps professionals.
3. **Find DevOps communities and networks**
With many online and offline platforms available, it's easy to find DevOps communities and networks that match your interests, goals and needs. Popular online platforms include DevOps Stack Exchange, a Q&A site for practitioners; DevOps Reddit, a Reddit community for topics; DevOps Chat, a Slack channel for conversations; and DevOps Meetup, a platform for finding and organizing local events. You can also search for other online platforms such as blogs, forums, newsletters, podcasts and social media groups focused on DevOps topics. Offline platforms such as coworking spaces, hackathons and industry associations can also offer opportunities to connect with peers and experts
4. **Join Groups and Communities:** On LinkedIn and other platforms, join groups and communities related to DevOps. This will allow you to follow relevant discussions and connect with other professionals.
5. **Participate in Conversations:** Be active in discussions by sharing your thoughts, asking questions, and commenting on other members' posts. Feel free to share your own experiences or seek advice on specific challenges you may be facing.
6. **Use Relevant Hashtags:** On platforms like Twitter, use popular hashtags related to DevOps to make your posts more visible to the community. This can include hashtags such as #DevOps, #ContinuousIntegration, #ContinuousDelivery, etc.
7. **Attend Virtual Events:** Look for webinars, virtual conferences, and other online events related to DevOps. These events often provide networking opportunities through live chats and discussion forums.
8. **Contribute to Open Source Projects:** If possible, consider contributing to open-source projects related to DevOps. This can allow you to interact with other contributors and build relationships within the community.
9. **Directly Connect with Professionals:** Feel free to send connection invitations or direct messages to DevOps professionals whose work you admire. Express your interest in the field and inquire if they would be open to discussing or sharing insights.
10. **Share your knowledge and feedback**

DOSSIER PROFESSIONNEL (DP)

Sharing your knowledge and feedback is a great way to contribute to DevOps communities and networks. By doing so, you can demonstrate your expertise, showcase your work, inspire others and learn from others. Here are some tips to help you share your knowledge and feedback effectively. Consider writing articles, creating videos, podcasts, webinars or courses on DevOps topics to share your ideas on platforms that reach a wide audience such as Medium or YouTube. In addition, you can upload your code, scripts, configurations or templates to platforms such as GitHub or GitLab for others to view, download or collaborate. You can also comment on other members' contributions and share your comments on platforms that enable interaction and discussion, such as DevOps Stack Exchange or DevOps Chat. What's more, you can rate and review contributions on platforms such as Medium or LinkedIn.

11. Benefits of contributing to DevOps communities and networks

Contributing to DevOps communities and networks can bring many benefits, both personally and professionally, including the opportunity to learn new skills and techniques that can help you improve your DevOps practice and performance. You can also build your reputation as a DevOps professional and leader, expanding your network of contacts and connections that can help you advance your career and opportunities. What's more, contributing to DevOps communities and networks can be fun, as you share your passion and enthusiasm for DevOps with other like-minded people, while feeling the sense of belonging and community that comes from being part of a group of like-minded people.

2. Précisez les moyens utilisés :

3. Avec qui avez-vous travaillé ?

- Travail individuel

4. Contexte

Nom de l'entreprise, organisme ou association ► *Cliquez ici pour taper du texte.*

Chantier, atelier, service ► *Cliquez ici pour taper du texte.*

Période d'exercice ► Du : **11/12/2023** au : **16/12/2023**

5. Informations complémentaires (facultatif)

DOSSIER PROFESSIONNEL (DP)

Titres, diplômes, CQP, attestations de formation

(facultatif)

| Intitulé | Autorité ou organisme | Date |
|--------------|----------------------------------|---|
| Cliquez ici. | Cliquez ici pour taper du texte. | Cliquez ici pour sélectionner une date. |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |

DOSSIER PROFESSIONNEL (DP)

Déclaration sur l'honneur

Je soussigné(e) [prénom et nom] Zemni Kassas Sarra,

déclare sur l'honneur que les renseignements fournis dans ce dossier sont exacts et que je suis l'auteur(e) des réalisations jointes.

Fait à Clamart le 23 Aout 2023

pour faire valoir ce que de droit.

Signature :

DOSSIER PROFESSIONNEL (DP)

Documents illustrant la pratique professionnelle

(*facultatif*)

Intitulé

Cliquez ici pour taper du texte.

ANNEXES

(*Si le RC le prévoit*)