

# Projet 3 - Déploiement VM + Kubernetes

## Cybersécurité 1A

### Présentation du projet

Ce projet Python automatise :

1. Le déploiement d'une machine virtuelle via vCenter (PowerCLI)
2. L'installation d'un cluster Kubernetes simple (1 master + 2 workers)
3. Le déploiement d'une application NGINX de test

### Structure du projet

```
Projet3_FullAutomation/  
project3_deployer.py      # Script principal Python  
vm-config.json            # Configuration pour la VM à déployer  
hosts.txt                 # IPs des machines pour le cluster K8s  
README.txt                # (ce fichier)
```

### Prérequis

Sur la machine de gestion (ton PC) :

- Python 3.7+
- PowerShell installé (Windows) ou pwsh (Linux)
- VMware PowerCLI installé
- Clé SSH valide autorisée sur les 3 machines Linux
- Bibliothèque Python paramiko : `pip install paramiko`

### Fichier vm-config.json (exemple)

Modifie ce fichier pour adapter à ton infrastructure vCenter :

```
1 {  
2   "vcenter": "vcenter.entreprise.com",  
3   "username": "admin@vsphere.local",  
4   "password": "YourPasswordHere",  
5   "template": "Template-Ubuntu-20.04",  
6   "vm_name": "WebServer-01",  
7   "cpu": 2,  
8   "memory": 4096,  
9   "disk": 50,  
10  "network": "VLAN_100",  
11  "ip_address": "192.168.1.100",  
12  "datastore": "Datastore1",  
13  "cluster": "Cluster1"  
14 }
```

## Fichier `hosts.txt`

Liste les IPs des 3 machines Ubuntu :

```
192.168.1.10 master
192.168.1.11 worker1
192.168.1.12 worker2
```

## Lancer le script

1. Vérifie les chemins et le SSH dans le script (`project3_deployer.py`) Lance — le :

```
1 python project3_deployer.py
```

## Ce que le script fait

2. Connecte à vCenter, clone la VM, démarre
  - Se connecte à chaque hôte Ubuntu via SSH
  - Installe Kubernetes (kubeadm/kubectl/kubelet/docker)
  - Initie le master node
  - Rejoint les workers
  - Déploie une app de test (NGINX)

## Test final

Connecte-toi au master :

```
1 kubectl get nodes
2 kubectl get pods -A
3 kubectl get svc nginx
```

## Problèmes courants

- SSH non configuré → ajoute ta clé publique dans `~/.ssh/authorized_keys`
- vCenter inaccessible → vérifie l'adresse IP ou le nom DNS
- paramiko non installé → fais `pip install paramiko`

**Bon déploiement !**