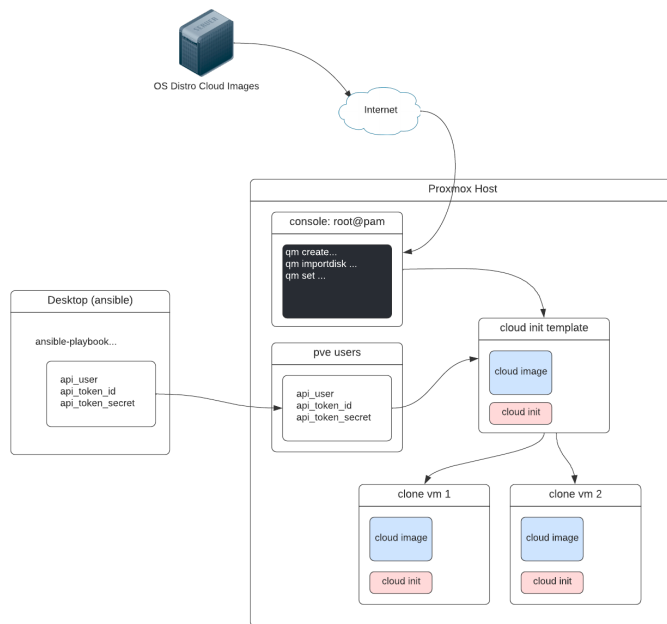
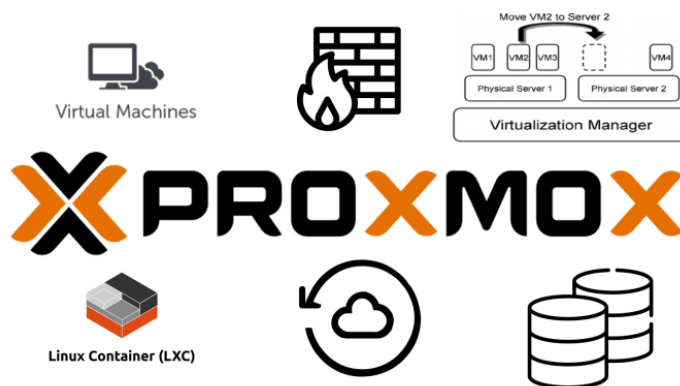


# Architecture de la baie de l'AFTEC Vannes

Cette Architecture consiste principalement en la combinaison de 3 logiciels **QEMU**, **Ansible** et **Proxmox**. Ces outils sont souvent utilisés ensemble dans des contextes de virtualisation et d'automatisation des infrastructures. Voici un aperçu de la façon dont ils pourraient interagir :



## 1. Proxmox



- **Proxmox** est un environnement de virtualisation de serveur open

source qui combine deux technologies de virtualisation : **KVM** (pour une virtualisation complète à l'aide de QEMU) et **LXC** (pour une virtualisation légère basée sur des conteneurs).

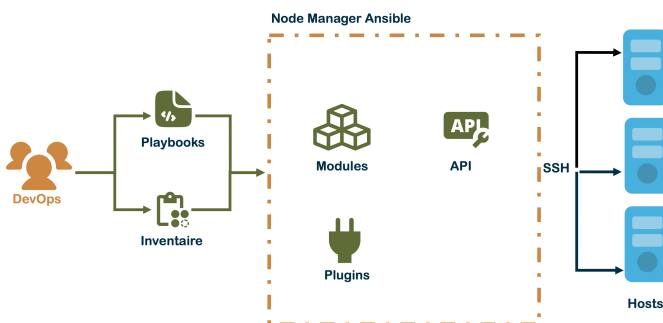
- **Proxmox VE** (Virtual Environment) est utilisé pour gérer les machines virtuelles (VM) et les conteneurs, fournissant une interface Web pour gérer les ressources, le stockage et la mise en réseau.

## 2. QEMU



- **QEMU** est un émulateur et un virtualiseur open source qui peut exécuter des machines virtuelles pour différentes architectures. Elle est souvent utilisée comme technologie sous-jacente à la virtualisation KVM.
- Dans un environnement **Proxmox**, **QEMU** est utilisé pour créer et gérer des VM. Proxmox agit comme une couche de gestion au-dessus de QEMU/KVM.

## 3. Ansible



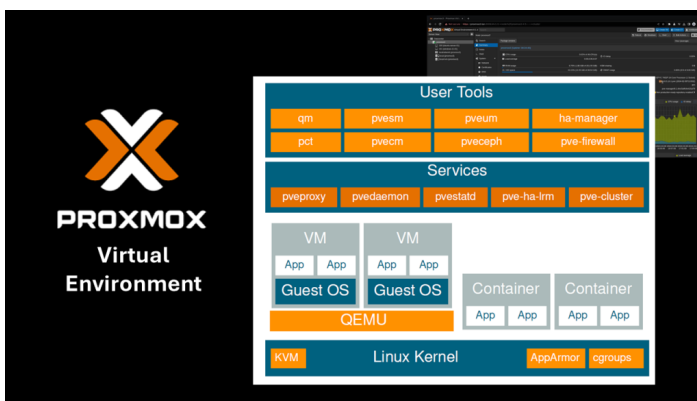
- **Ansible** est un outil d'automatisation open source utilisé pour

la gestion de la configuration, le déploiement d'applications et l'automatisation des tâches.

- Il est sans agent, ce qui signifie qu'il ne nécessite l'installation d'aucun logiciel sur les machines cibles. Il utilise SSH (ou d'autres méthodes) pour communiquer et effectuer des tâches.

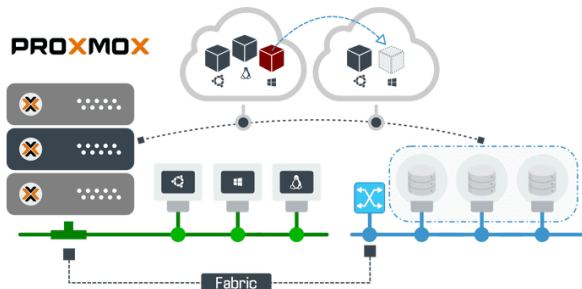
## Comment ils travaillent ensemble

### 1. Déploiement de VM avec QEMU sur Proxmox :



- Proxmox utilise QEMU/KVM pour exécuter des machines virtuelles. Lorsque vous créez une nouvelle VM dans Proxmox, elle s'exécute essentiellement sur QEMU.
- Vous pouvez gérer les machines virtuelles, les instantanés, le stockage et la mise en réseau via l'interface Web Proxmox ou via des outils de ligne de commande.

### 2. Utiliser Ansible pour automatiser Proxmox :

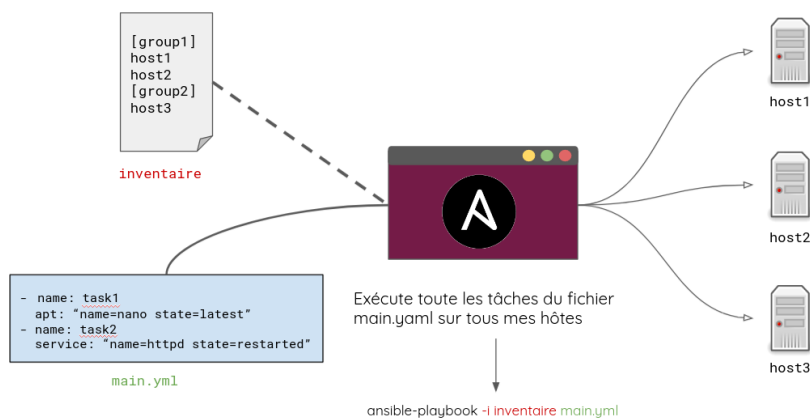


- Ansible peut être utilisé pour automatiser la gestion de Proxmox et

de ses machines virtuelles.

- Il existe des modules et des collections Ansible spécifiques comme « `community.general.proxmox` » qui vous permettent d'interagir avec l'API de Proxmox. Vous pouvez automatiser des tâches telles que la création, la configuration, le démarrage et l'arrêt de machines virtuelles ou de conteneurs.
- Vous pouvez également utiliser Ansible pour configurer les machines virtuelles après leur provisionnement (par exemple, installer un logiciel, configurer des utilisateurs, configurer des réseaux).

### Exemple de flux de travail



- **Provisionnement** : utilisez Proxmox pour provisionner une nouvelle VM basée sur un modèle QEMU.
- **Automatisation avec Ansible** :
  - Écrivez un playbook Ansible pour automatiser le déploiement d'applications, de services ou de configuration sur la VM nouvellement créée.
  - Vous pouvez également utiliser Ansible pour gérer directement les ressources Proxmox, comme la création de nouvelles machines virtuelles, la gestion du stockage et la gestion des sauvegardes.

## Exemple de Playbook Ansible pour créer une VM sur Proxmox

Voici un exemple de base de ce à quoi pourrait ressembler un playbook Ansible pour créer une VM sur Proxmox :

```
---
- name: Create a new VM on Proxmox
  hosts: localhost
  gather_facts: no

  tasks:
    - name: Create VM
      community.general.proxmox:
        node: pve
        api_host: "https://proxmox-server.example.com:8006/api2/json"
        api_user: "root@pam"
        api_password: "your-password"
        validate_certs: false
        vmid: 101
        name: "example-vm"
        cpu: 2
        memory: 2048
        storage: "local-lvm"
        disk_size: "10G"
        network:
          - model: virtio
            bridge: vmbr0
        state: present
```

Ce playbook créera une nouvelle VM avec des ressources spécifiées sur le serveur Proxmox. Une fois la VM créée, vous pouvez écrire des tâches supplémentaires dans Ansible pour configurer la VM.

## Conclusion

La combinaison de QEMU, Ansible et Proxmox permet de créer un environnement de virtualisation et d'automatisation puissant et flexible. Vous pouvez utiliser Proxmox pour gérer les VM et Ansible pour automatiser à la fois la gestion de Proxmox et la configuration des VM elles-mêmes. Il s'agit d'une

configuration courante dans les environnements DevOps où l'infrastructure doit être à la fois évolutive et facile à gérer.