Document de conception
SAE S4.Deploi.01

Tristan Petit, Nils Hubert, Toni Rey,
Majd El Sebeiti , Vianney Miquel

20 mars 2025

Table des matières

1 Introduction

En charge de l'IT pour l'entreprise PLETER, spécialisée dans l'animation 3D, un secteur fortement dépendant de l'informatique. Il est essentiel de mettre en place une infrastructure réseau robuste et sécurisée, capable de répondre à tous les besoins de l'entreprise. Cette infrastructure doit également être conçue pour supporter l'ensemble des logiciels et systèmes d'exploitation nécessaires aux équipes créatives et de développement, afin de garantir une compatibilité optimale avec les outils utilisés (ex : Blender, Adobe Creative Cloud, logiciel Autodesk, Unreal Engine, Cinema 4D, etc).
Lors de la phase de réflexion du projet, nous avons donc décidé d'utilser proxmox afin de virtualiser nos machines virtuelles. Le but initial était de prendre Qemu sur le serveur interne à notre IUT(assr) afin de faire tourner une machine virtuelle avec suffisament de ressource afin de pouvoir faire tourner notre infrastructure dedans. Avec proxmox, comme nous avons des hyperviseurs réunit dans un cluser nous amélioront les performances tout en complexifiant la tâche. En effet, les interface de VLAN de proxmox rendent l'approche de configurer des routeurs qui permettent de connecter des VMs à travers les hyperviseurs complexe. Nous avons les VxLan de l'interface proxmox qui doivent ainsi contenir les hyperviseurs mais pas relier les VM à part la vm routeur qui sera relié à un hyperviseur proxmox. D'autres approchent sont sans doute possible, mais c'est la piste sur laquelle nous cherchons. Afin de gagner du temps sur la partie réalisation, la plupart des tâches récurrentes comme la configuration des serveurs DHCP qui peupleront chaque sous réseaux, ont été crées. Les test de connectivité permettent donc de réaliser les tests sans problèmes particuliers.

2 Rappel et évolution sur l'architecture

L'architecture que nous avons imaginé n'ayant pas réellement commencé, nous n'avons pour le moment rien changé. Il nous faut trouver le moyen de configurer un routeur pour les machines virtuelles en les connectant entre elles à travers les hyperviseurs de manière transparente.

Nous avons fait pour le moment qu'une seule modification en la personne des routeurs. Nous n'avons désormais qu'un seul routeur qui permet d'interconnecter les réseaux qui ici sont des vlans, étant donné que nous avons un serveur DHCP par vlan celui ci devra rediriger les requêtes DHCP des machines vers le serveur en question et reciproquement. Il y aura également un service DNS tourant dans la DMZ pour assurer la résolutution de nom. Donc deux serveurs DNS, avec celui qui se trouve dans l'intranet des serveurs

3 Ressources utilisées

Pour notre SAE nous avons réunis les machines dans un cluster pour interconnecter nos hyperviseurs et de mutualiser les ressources qui sont répartis à travers nos hyperviseurs.

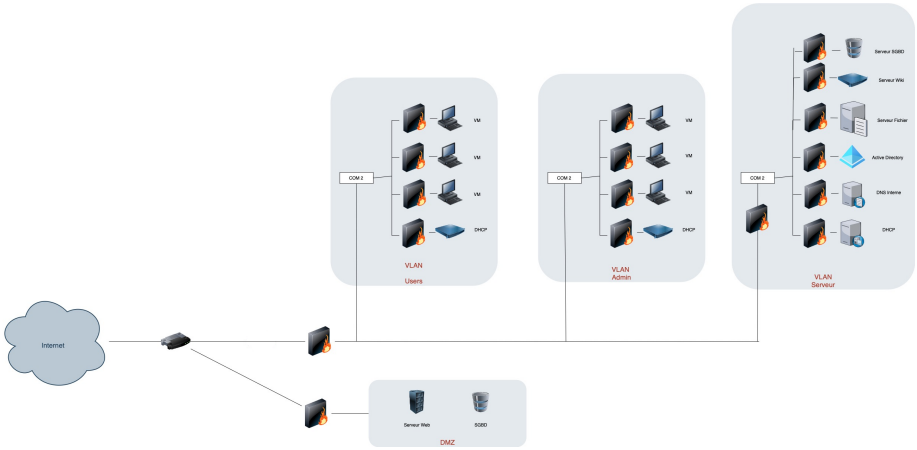

FIGURE 1 – Architecture de la première partie

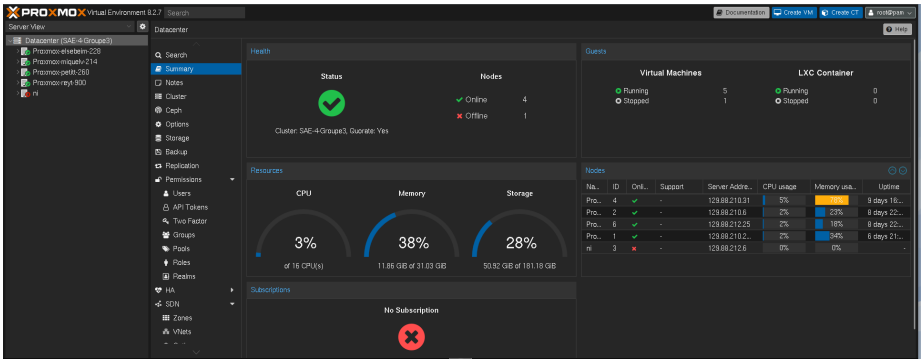

FIGURE 2 – Ressource utilisée pour la SAE sous proxmox

L'avantage de cette configuration par rapport à celle présente sur ASSR, c'est que nous n'avons pas machine virtuelle imbriqué cela permet d'améliorer de manière significative les performances.
Le CPU est la ressource qui pose le moins de problème dans notre cas, il en est tout autre pour la RAM. Celle ci va devoir être géré de manière équitable entre les hyperviseurs pour eviter une surcharge. Le stockage ne devrait pas poser problème outre mesure, mais pourrait le devenir si l'on était dans l'obligation de rajouter des machines.

4 Documentation Technique

4.1 DNS Interne

Ce réseau abrite un contrôleur de domaine Active Directory responsable de la gestion du domaine **local.pleter.ovh**. Afin d'assurer un fonctionnement optimal du domaine, il est recommandé que la gestion des domaines soit confiée au serveur DNS de Windows Server. Toutefois, pour éviter toute surcharge du serveur principal et garantir la redondance, un autre serveur DNS sera déployés pour l'ensemble de l'infrastructure. Un mécanisme de transfert de zone DNS sera mis en place via IXFR et sécurisé par TSIG. Ainsi, le serveur DNS intégré de Windows Server agira en tant que serveur principal, tandis qu'un serveur DNS Bind9 sera configuré en tant que serveur esclave. Le serveur DNS Bind9 assurera principalement la résolution DNS pour l'infrastructure, mais le serveur DNS Windows Server restera accessible comme serveur secondaire. Pour renforcer la sécurité des requêtes DNS, le protocole DoH (DNS over HTTPS) sera implémenté sur les deux serveurs DNS. Ces serveurs fonctionneront également en tant que résolveurs DNS récursifs et garantiront l'authenticité des réponses grâce à DNSSEC.

4.2 DNS Externe

L'infrastructure comprend une DMZ avec plusieurs serveurs de PLETER. Afin de garantir un accès fluide à ces serveurs depuis Internet via un nom de domaine, un serveur DNS Bind9 sera déployé dans la DMZ pour gérer la résolution des noms ***.pleter.ovh**. Il ne couvrira toutefois pas le sous-domaine interne **local.pleter.ovh**. Pour assurer l'intégrité des réponses, ce serveur mettra en œuvre DNSSEC sur sa zone, tout en proposant les services DoH et DoT (DNS over TLS) pour sécuriser les communications.

5 Introduction

Comme dit précédemment, la mise en place d'un routeur sur proxmox est complexe, c'est pour cela qu'on ne l'a pas encore mis en place. Ce document décrit donc le fonctionnement d'un script Bash à configurer un routeur qui segmentera le réseau en plusieurs VLAN. Le script met en place la transmission IPv4, installe les paquets nécessaires, configure plusieurs interfaces virtuelles associées aux VLAN, et initialise un pare-feu nftables.

6 Structure du Script

Le script est séparé en 5 étapes : sélection de l'interface, configuration du routage IPv4, installation du support VLAN, configuration des VLAN, et mise en place du pare-feu.

    - Sélection de l'interface réseau : On demande à l'utilisateur l'interface cible puis vérifie son existence.
- Activation du routage IPv4, en ajoutant la ligne 'net.ipv4.ip$_forward = 1$'à'$/etc/sysctl.conf$', $onactiveleroutage$
$Lescriptvérifieetinstallelepaquetpourles VLAN sinécessaire, ilsegmenteensuiteleréseaucommeprécisédans$

7 Tests et Validation

Après exécution du script, on pourra vérifier les points suivants avec les autres scripts créés :

- Connectivité entre les VLAN.
- Vérification du routage IPv4.

8 Conclusion

Pour l'instant, nous n'avons pas de routeur, mais on exécutera ce script qui permet de configurer un routeur pour un réseau segmenté en VLAN. Il faudra ensuite configurer un Pare-feu pour sécuriser ce routeur.