Présentation du projet d'auto-hébergement du club Kr[HACK]en

Pierre Coimbra

Contents

1 Préface

Ce document est une documentation non technique qui n'abordera pas ce qu'il y a derrière l'infrastructure (Redondance des services clé, communication entre les deux PC, configuration du cluster...). Une documentation beaucoup plus technique qui part de zéro et détaille la mise en place de l'infrastructure est disponible à cette adresse.

`https://github.com/coimbrap/serveur_proxmox_dev_krkn`

Nous présentons ici les objectifs du projet, l'infrastructure matérielle et logicielle et l'articulation globale des services.

2 Motivation du projet

D'abord, l'idée est d'héberger tous les outils utilisés par le club (Web, NextCloud,Git...) afin d'avoir un contrôle total sur les services que nous utilisons. Ensuite, nous voudrions mettre en place une structure capable d'accueillir des environnements de CTF correctement cloisonnés par rapport aux services permanents du club.

3 Gestion du serveur en interne

Nous sommes conscients que, dans un tel projet le plus dur n'est pas de monter l'infrastructure mais de la maintenir au fil des années. Les responsabilités seront donc gérées de manière extrêmement strictes, avec plusieurs niveaux d'accès. Il faudra en effet différencier le poste de webmestre, qui ne pourra agir que sur la partie applicative, de celui de l'administrateur système qui aura l'accès global. De grands pouvoirs appelant de grandes responsabilités, les adminsys en poste auront la charge de former leur successeurs.

Pour la gestion en interne du serveur, nous nous organiserions de la manière suivante :

- Seules deux personnes du bureau auront le rôle d'administrateur système, soit tous les droits sur le serveur.
- Le responsable technique du club aura le rôle de webmestre, il pourra intervenir sur les services comme le site web, le cloud... Cependant, il ne pourra pas toucher à l'infrastructure autour.
- Tous les membres actifs du club auront accès aux services web.

4 Présentation de l'infrastructure

4.1 Infrastructure matérielle

Du côté infrastructure, nous disposons d'un rack 1U avec deux PC à l'intérieur possédant chacun 24Go de DDR3-ECC et un Xeon x5670 6 Coeurs cadencé à 2.93 GHz. Côté stockage, nous allons mettre en place un RAID1 ZFS avec deux disques par PC (les données du premier disque seront aussi présentes sur le second) ainsi le stockage sera répliqué pour éviter toute perte de données.

4.2 Infrastucture logicielle

Les deux PC accueilleront Proxmox comme hyperviseur ; un hyperviseur permet à plusieurs systèmes virtualisés de travailler sur une seule machine physique en même temps en se partageant les ressources disponibles.

Pour mieux comprendre comment les services seront disposés sur le serveur, il faut se dire que chaque service aura un contenant uniquement pour lui (VM ou container) et qu'il ne pourra communiquer que selon des règles bien précises avec l'hyperviseur (Proxmox) et les autres contenants.

4.3 Les différents services

Nous allons présenter rapidement les services que nous envisageons mais n'aborderons pas ici ce qui permet d'accéder à ces services (Firewall, Proxy...)

Pour rappel : **1 service = 1 contenant**

4.3.1 Les services

Il y aura deux types de services,

- Ceux qui sont directement accessibles depuis Internet derrière le pare-feu, ce sont les services frontend.
- Ceux qui sont accessibles uniquement à travers une frontend, ce sont les services backend.

4.3.2 Services "sensibles"

L'infrastructure du club s'articulait de la manière suivante :

- Le site web du club.
- Le Wiki du club.
- Un serveur mail pour remplacer le service fourni par OVH.

Avec en plus,

- Un annuaire LDAP (openldap), qui permettra d'avoir un compte unique pour chaque utilisateur et de définir différents groupes d'utilisateurs.
- Un cloud (NextCloud) pour mettre en commun des fichiers au sein du club et l'ordre du jour des réunions.
- Un serveur Git (Gitea) sur lequel toutes les sources des challenges du club seront stockées ainsi que la documentation du club.
- Un service de messagerie instantanée du type Mattermost.
- Et d'autres services...

Ce qui permettrait d'auto-héberger tous les services du club.

4.3.3 Services CTF

L'objectif est de remplacer la banque de challenge du club stockée actuellement sur un poste en B141. Celui-ci n'est pas documenté, ce qui réduit les modifications que nous pouvons y apporter.

A partir des sources des challenges actuels, une nouvelle infrastructure CTF prendra forme. Elle s'organisera de la manière suivante :

- Un premier CTFd avec tous les challenges du club utilisés pour les OpenCTF.
- Un autre CTFd que nous utiliserons pour les sessions en externe, comme par exemple pour la session 0.
- Une VM avec différents environnements Docker temporaires pour les challenges système.
- Une VM avec différents environnements Docker pour les challenges Web.

4.4 Articulation globale

Il y aura trois switchs virtuels afin de séparer la partie administration de la partie commune.

Un switch = Une partie

Dans chaque partie, il y aura des sous-parties que l'on appellera ici zones.

Une partie = Plusieurs zones

De même, dans chaque zone, il y aura un type de services.

Une zone = Un type de service

Un switch contiendra donc plusieurs zones qui contiendront elles-mêmes un type de service.

4.4.1 Schéma de l'infrastructure

Ce schéma décrit l'infrastructure réseau, chaque partie est détaillée sommairement ci-dessous.

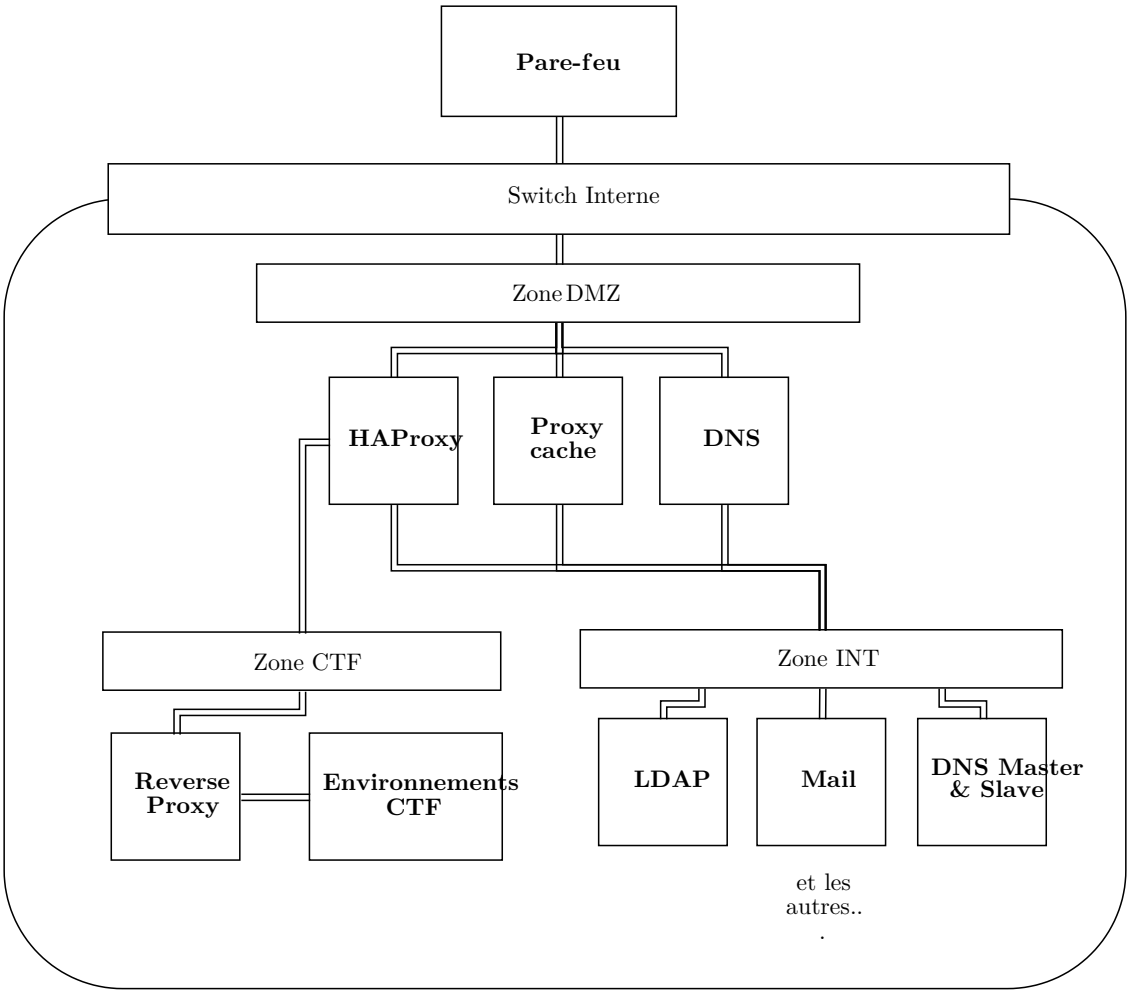

Figure 1: Topologie réseau du switch interne

5 Choix technique

Voici la liste non exhaustive des choix technique majeur que nous avons fait.

5.1 Création et configuration initiale

La création des CT/VM se fera via des rôles Ansible tout comme la configuration SSH et le déploiement des clés SSH.

5.2 Déploiement

Le déploiement de l'intégralité de l'infrastructure se ferra via des rôles Ansible afin de permettre de tout remettre en place rapidement.

5.3 Cloisonnement réseau

Les conteneurs/VMs seront séparés en plusieurs zones, chaque zone sera dans une VLAN séparée. La gestion des VLANs se ferra via OpenvSwitch.

5.4 Firewall

Un firewall principal, de type OPNsense, sur chaque node avec une IP virtuelle sur la WAN,

Il s'occupera :

- Faire passer l'IP principale entre les deux nodes en fonction de la disponibilité
- D'autoriser que ce qui est nécessaire sur une zone
- Des communications entre zones qui sont nécessaire pour les proxy et l'administration
- De fournir un VPN pour l'administration du serveur

Chaque conteneur/VM aura un firewall plus léger du style UFW qui autorisera uniquement les ports nécessaire

5.5 Reverse proxy

Les reverse proxy sont séparés entre la partie CTF et la partie service,

- Un HAProxy en DMZ qui renvoie la partie CTF vers un reverse proxy spécifique pour la zone CTF et qui renvoie les requêtes sur la partie services sur le/les bons conteneurs
- Le reverse nginx pour la partie CTF est un choix, même si cela rajoute une seconde couche de proxy avant d'arriver au service cela permet de pouvoir modifier la configuration des environnements CTF sans toucher à HAProxy

En résumé : Zone CTF, deux couches de reverse proxy. Zone Interne pour les services, une seule couche de reverse proxy

Pour l'accès aux services il y aura obligatoirement un serveur internet nginx devant le service.

5.6 DNS

Gestion par vues avec une vue pour les résolutions depuis l'extérieur et une zone pour les résolutions interne.

- Un hidden master faisant autorité
- La vue externe sera accessible via un master/slave en frontend
- La vue interne sera accessible via un slave en backend

TSIG pour les transferts de zone et DNSSEC

5.7 Gestion des accès

5.7.1 Partie Admin

Sans VPN il sera possible d'accéder au bastion et donc aux conteneurs du serveur via SSH (limite à définir). Le VPN sera nécessaire pour accéder aux différents panels d'administration (PVE, PMG, OPNsense...)

- Pour ce qui est des accès sans VPN on utilisera un bastion avec des comptes nominatifs et des logs, l'authentification se fera par clé SSH. La remonté des données se ferra probablement via l'annuaire LDAP.
- Pour la partie avec VPN ce sera des comptes nominatif soit locaux soit via l'annuaire LDAP en fonction des possibilités technique.

5.7.2 Partie User

Pour la connexion aux services tout passera par un annuaire LDAP manageable via une interface web.

5.8 Sauvegarde

Deux options :

- Backup externalisé sur un tiers avec Borg Backup
- Backup entre nodes avec Proxmox Backup Server

Seul Borg Backup à été testé, il faut voir les performances avec Proxmox Backup Server

5.9 Accès à l'extérieur, proxy cache et apt proxy

Les requêtes HTTP.S et APT passerons par

- Un proxy Squid pour HTTP.S
- Un proxy APT (apt-cacher-ng) pour apt

De cette manière les conteneurs n'auront pas directement accès à internet

5.10 Certificats SSL/TLS

Pour Let's Encrypt on utilisera la validation par DNS ce qui permet de ne pas avoir de serveur à modifier pour l'obtention du certificats et de /pouvoir obtenir des certificats wildcard.

Pour ce qui des communications en "interne" (ex. entre un serveur et un proxy) on utilisera une autorité de certification locale.

6 Pour aller plus loin...

Une documentation plus technique sur la façon de le mettre en place est disponible à cette adresse :

`https://github.com/coimbrap/serveur_proxmox_dev_krkn`

Une grande partie des rôles Ansible en rapport avec les choix technique sont disponible ici :

`https://github.com/coimbrap`