



Université Mohammed Premier

Ecole Supérieure de Technologie

-Oujda-



PROXMOX

المدرسة العليا للتكنولوجيا
École Supérieure de Technologie

Département Génie informatique

Filière : Administrateur de Système et Réseaux

Project de Fin D'études :

La Haute Disponibilité en Technologie de Serveur

PROXMOX

Réalisé par :

TOR Mohcine

RAHAL Yousra

Encadré par :

Mr. GRARI Monir

Année Universitaire :

2019/2020

PROXMOX

Table des matières

| | |
|--|----|
| REMERCIEMENT | 3 |
| INTRODUCTION GENERALE..... | 4 |
| Partie 1:Etude Théorique..... | 5 |
| Chapitre 1 : Contexte et Objectifs..... | 6 |
| 1.1 Introduction :..... | 6 |
| 1.2 Positionnement et problématique : | 6 |
| 1.3 La solution propose : | 6 |
| Chapitre 2 : Virtualisation en général | 7 |
| 2.1 Introduction :..... | 7 |
| 2.2 Histoire de la Virtualisation..... | 8 |
| 2.3 Virtualisation : | 9 |
| Chapitre 3 : Virtualisation avec Proxmox et la Haute Disponibilité : | 15 |
| 3.1 La virtualisation à travers la solution libre Proxmox VE :..... | 15 |
| 3.2 Principe fonctionnement de Proxmox et la Haute Disponibilité :..... | 16 |
| Partie 2: Etude Pratique..... | 28 |
| Chapitre 1 : Métrise de proxmox : | 29 |
| 1.1 Installation de Proxmox | 29 |
| 1.2 Installation d'une machine Virtuelle :..... | 36 |
| 1.3 Clonage de la machine virtuelle :..... | 42 |
| 1.4 Sauvegarde et restauration de la machine virtuelle : | 43 |
| 1.5 Création d'un conteneur :..... | 46 |
| 1.6 Migration d'un conteneur : | 50 |
| Chapitre 2 : la haute disponibilité : | 51 |
| 2.1 Mise en place d'un cluster entre 3 Serveur :..... | 51 |
| 2.2 Mise en place d'un cluster HA :..... | 53 |
| CONCLUSION GENERALE | 61 |
| Bibliographie | 62 |

PROXMOX

REMERCIEMENT

C'est une tâche très agréable, mais bien délicate, de présenter nos remerciements à tous ceux qui nous ont aidés dans la réalisation de ce travail.

Nous exprimons nos hautes considérations et nos vifs remerciements à tous nos enseignants du Département de génie Informatique pour les enseignements qui nous prodigués tout au long de notre années de formation dans la filière Administration système et Réseau.

Tout particulièrement, nous remercions notre encadrant Professionnelle **M.GRARI Monir**, qui a dirigé notre travail tout au long de période de PFE avec ses précieux conseils, ses recommandations et son riche savoir-faire, aussi il a partagé avec nous sa bonne humeur, ses connaissances, et sa disponibilité durant l'encadrement.

Ce Project a été réalisé au sein du L'Ecole supérieur de technologie Oujda, pendant une durée de deux mois. Nous tenons donc à témoigner toute nos reconnaissances à tous le personnel administratif et technique de cet établissement pour l'assistance et l'aide qui nous prodiguer pour réaliser ce Project. De nombreuses personnes dont les noms ne sont pas cités nous aidé, encouragé, conseillé. Nous espérons pouvoir un jour leur témoigner nos reconnaissances.

Nous tenons à exprimer nos affectueuses reconnaissances à nos parents particulièrement nos mère pour son aide, son soutien et ses encouragements, nos famille à qui on a beaucoup d'estime et de reconnaissance. Enfin nos chers Professeurs pour avoir accepté de faire partie du Jury. Nous exprimons nos très hautes considérations, nos profonds respects et nos vifs remerciements.

PROXMOX

INTRODUCTION GENERALE

Aujourd'hui un constat s'impose dans l'informatique : tout doit marcher tout le temps. C'est d'autant plus vrai et critique en entreprise que l'activité de production (dans les services mais aussi dans l'industrie) dépend de plus en plus du système d'information de la société (logiciel logistique, site Web...) et de sa connexion au monde extérieur.

Le présent travail s'inscrit dans le cadre du Project de fin d'études en vue de l'obtention du diplôme de Techniciens spécialise en Administration système et réseaux (ASR) pour l'année 2019/2020.

Ce Project nous a permis de passer de la théorie à la pratique, et nous a permis aussi de mettre en application nos connaissances acquises pendant cette formation et de maîtriser les mécanismes susceptibles de nous faciliter l'intégration dans le monde du travail pour le contact avec la vie professionnelle.

Dans le premier chapitre, nous présentons notre Project en étudiant son cadre général et la problématique avant de critiquer les solutions existantes pour dégager leurs insuffisances et proposer les orientations de notre solution future.

La deuxième partie exposera une étude théorique concernant la virtualisation en générale et la virtualisation avec solution Proxmox.

La troisième partie est consacrée à une étude profonde de virtualisation Proxmox afin d'en dégager une solution pour permettre la haute disponibilité des services.

Partie 1:Etude Théorique

Chapitre 1 : Contexte et Objectifs.

1.1 Introduction :

La **haute disponibilité** (ou High Availability ou HA) permet d'assurer et de garantir le bon fonctionnement des services ou applications proposées et ce 7j/7 et 24h/24.

Cela consiste donc à mettre en place toutes les actions et dispositions techniques pour qu'une infrastructure informatique soit toujours disponible en appliquant certains principes tels que la réPLICATION DES DONNÉES, la sauvegarde, la répartition de la charge, la redondance, etc. pour limiter l'indisponibilité d'un SI.

Assurer la haute disponibilité d'un service et des données, signifie notamment être capable d'assurer la continuité du service malgré une panne du serveur sur lequel il est situé.

1.2 Positionnement et problématique :

Comment assurer la haute disponibilité de notre infrastructure informatique ?

1.3 La solution proposée :

Proxmox Virtual Environment (PVE) est une plateforme de virtualisation qui tourne sur système Linux et qui permet de créer des machines virtuelles de type **OpenVZ** et **KVM**.

Grâce à une interface graphique de type Web, Proxmox VE facilite la création et l'administration des machines virtuelles (VM) et fournit une vue synoptique de l'ensemble des VM installées sur plusieurs hôtes physiques. Proxmox VE permet de contrôler plusieurs machines virtuelles réparties sur plusieurs hôtes physiques, en ce sens on peut dire qu'il crée une grappe ("cluster") de machines virtuelles.

Chapitre 2 : Virtualisation en général

2.1 Introduction :

L'objectif principal de la virtualisation est de permettre à des utilisateurs d'exécuter un ou plusieurs systèmes d'exploitation au sein d'environnements isolés mais la virtualisation possède de nombreux autres avantages ! On parle d'évolutivité, de réductions de coûts, d'optimisation de ressources etc.

Ne vous êtes-vous jamais demandés quelles pouvaient être les répercussions d'une machine qui tombe en panne ? Ou encore les conséquences d'une mise à jour qui aurait échoué ! Dans le deuxième cas, vous me direz qu'il y avait moyen d'effectuer des tests sur une autre machine avant tout migration !

Vous savez quoi ? Vous avez raison ! Mais **le risque zéro** n'existe pas, il suffit d'une petite exception et hop la migration réussie sur l'autre ordinateur échoue sur votre machine de production. Si jamais une machine venait à s'éteindre brusquement en cas de coupure d'électricité ou pour tout autre raison, le risque qu'elle ne redémarre plus est toujours présent, et si jamais vous voulez revenir en arrière il se pourrait que vous soyez victime d'un très long périple pour y arriver !

Et puis même si jamais vous voulez essayer la haute disponibilité, la virtualisation s'avère toujours plus utile, en effet elle vous permettra d'économiser sur vos dépenses car la facture pourrait monter très vite. Tout d'abord parce que le temps accordé à la maintenance de vos différentes machines a un coût et l'acquisition de matériel nécessaire possède également un coût. Avec la virtualisation le temps consacré à la maintenance sera relativement moins élevé et permettra également aux administrateurs de se consacrer à des tâches beaucoup plus importantes.

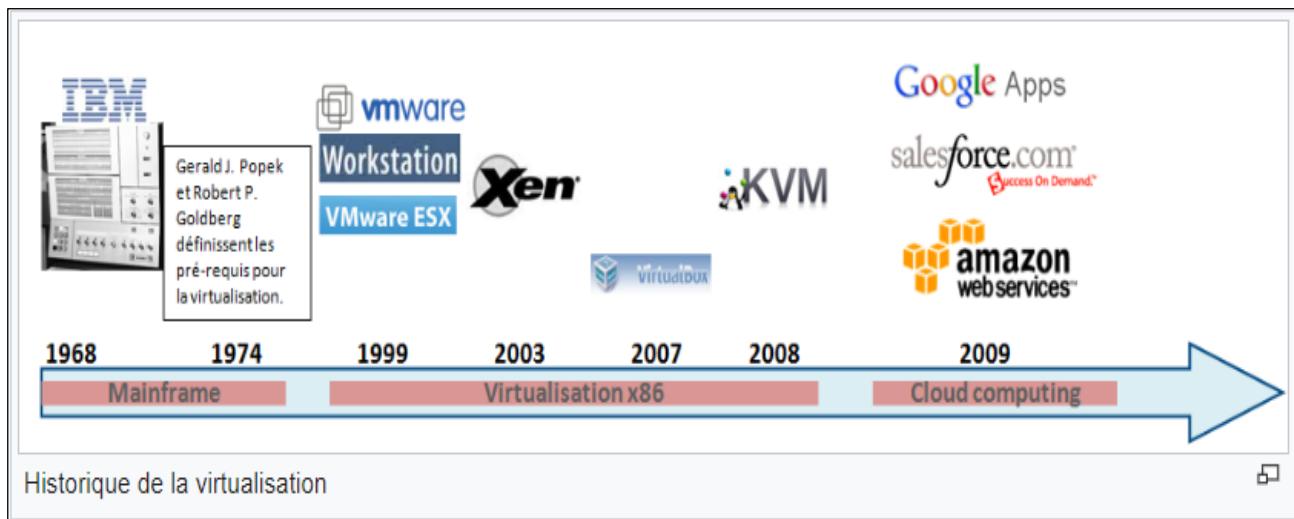
PROXMOX

Quelques arguments en faveur de cette technologie :

- ❖ Coût : Tout ajout de machine nécessite de prévoir des coûts de fonctionnement et de maintenance. La virtualisation concentre plusieurs serveurs en un seul.
- ❖ Rentabilité : Les serveurs sont sous-utilisés. D'après une étude de l'International Data Corporation, seuls 15% de la capacité totale est utilisée sur un serveur. La virtualisation permet d'utiliser les machines au maximum de leurs capacités.
- ❖ Encombrement : Il n'y a pas forcément toujours la place pour rajouter un serveur dans le local informatique, l'économie des serveurs liée à la fusion par la virtualisation résout cette problématique.
- ❖ Ecologique : Une salle serveur nécessite de l'électricité pour les serveurs mais aussi pour la climatisation. Il faudra également recycler les serveurs lors de leur fin de vie. Encore une fois, la virtualisation réduit les coûts (et éventuellement allège les consciences).
- ❖ Migration : Grâce à la migration à chaud des machines, la continuité de service est garantie (AMD migrant - via VMware Esx -, par exemple, permet une migration sans interruption de service).

On voit que la virtualisation est indispensable pour réduire efficacement les coûts, garantir une stabilité de service et une véritable continuité associée à des performances élevées.

2.2 Histoire de la Virtualisation



PROXMOX

La virtualisation a été réellement abandonnée au cours des années 1980 et 1990, lorsque les applications client-serveur, ainsi que les postes de travail et les serveurs x86 bon Marché, se sont imposés comme le modèle de l'informatique distribuée. Au lieu de partager des ressources centralisées dans le modèle de mainframe, les entreprises ont utilisé Des systèmes distribués moins coûteux pour créer des îlots de capacité de calcul. L'adoption généralisée de Windows et l'émergence de Linux comme systèmes d'exploitation serveurs dans les années 1990 ont fait des serveurs x86 la norme de l'industrie. La croissance du déploiement des postes de travail et des serveurs x86 a introduit de nouveaux défis en matière d'exploitation et d'infrastructure informatique.

2.3 Virtualisation :

1. Concepts de base de la virtualisation :

Cette partie traite des concepts de base de la virtualisation. Comme vous pouvez l'avoir deviné, il existe plusieurs techniques de virtualisation. Il sera tout d'abord question des méthodes/modes de virtualisation desquels s'en dégageront des concepts généraux.

A. La virtualisation par isolation :

L'isolation est une technique permettant d'emprisonner l'exécution des applications dans des contextes. Cette solution est très performante (le surcoût d'une application isolée/virtualisée est minime par rapport au temps d'exécution de la même application installée sur un système d'exploitation). La performance est donc au rendez-vous, cependant on ne peut pas parler de virtualisation de systèmes d'exploitation car l'isolation ne consiste à virtualiser que des applications. On pourrait par contre avoir plusieurs instances de Tomcat qui écoutent sur le même port, plusieurs Apaches sur le port 80 etc.

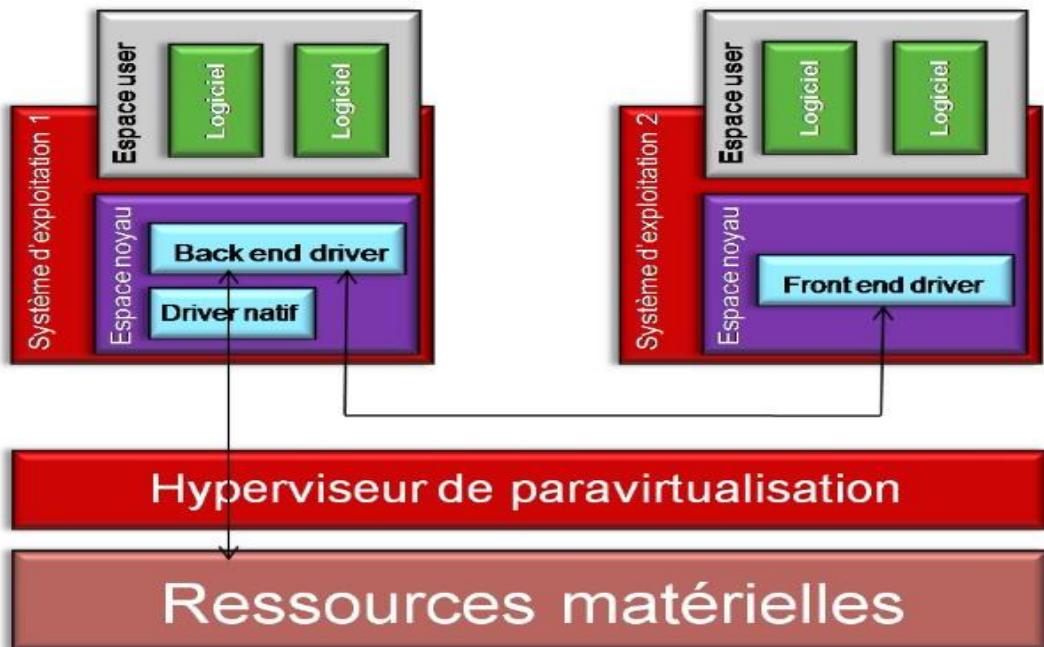
Elle permet de séparer un système en plusieurs contextes ou environnements. Chacun d'entre eux est régi par l'OS hôte, mais les programmes de chaque contexte ne sont capables de communiquer qu'avec les processus et les ressources associées à leur propre contexte.

Il est ainsi possible de partitionner un serveur en plusieurs dizaines de contextes, presque sans ralentissement.

L'isolation est utilisée sous Unix pour protéger les systèmes. Via des mécanismes comme chroot ou jail il est possible d'exécuter des applications dans un environnement qui n'est pas celui du système hôte, mais un mini -

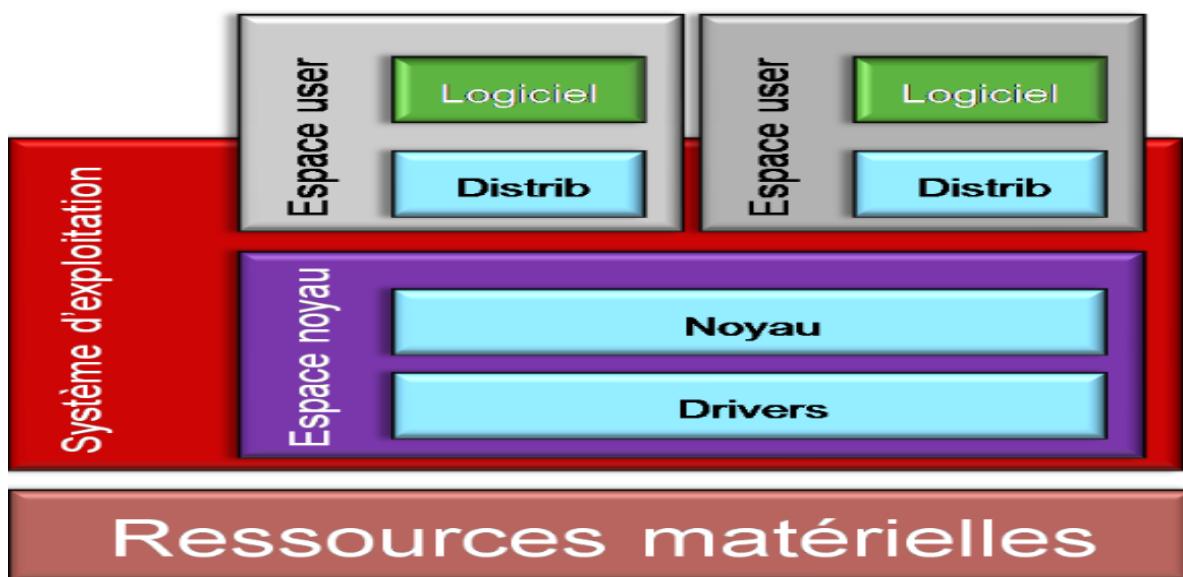
PROXMOX

Système ne contenant que ce dont l'application a besoin, et n'ayant que des accès limités aux ressources.



B. La paravirtualisation :

Par opposition à la virtualisation, on parle de Paravirtualisation lorsque les systèmes d'exploitation doivent être modifiés pour fonctionner sur un hyperviseur de Paravirtualisation. Les modifications sont en fait des insertions de drivers permettant de rediriger les appels système au lieu de les traduire



PROXMOX

La Paravirtualisation présente au système d'exploitation une machine générique spéciale, qui requiert donc des interfaces spéciales intégrées aux systèmes invités, sous la forme de drivers.

La Paravirtualisation est une technique de virtualisation de plus bas niveau que l'isolation. Elle partage avec cette dernière la nécessité d'utiliser un OS modifié. Plus précisément, en Paravirtualisation ce n'est plus seulement l'OS hôte qui doit être modifié mais également les OS appelés à s'exécuter sur les environnements virtuels.

Chaque système virtuel doit être modifié de façon à utiliser cette interface pour accéder au matériel. Contrairement à l'isolation, plusieurs OS de familles différentes peuvent fonctionner sur un même serveur physique. Il est ainsi possible de faire fonctionner Linux, NetWare, Solaris (et d'autres) simultanément sur une même machine.

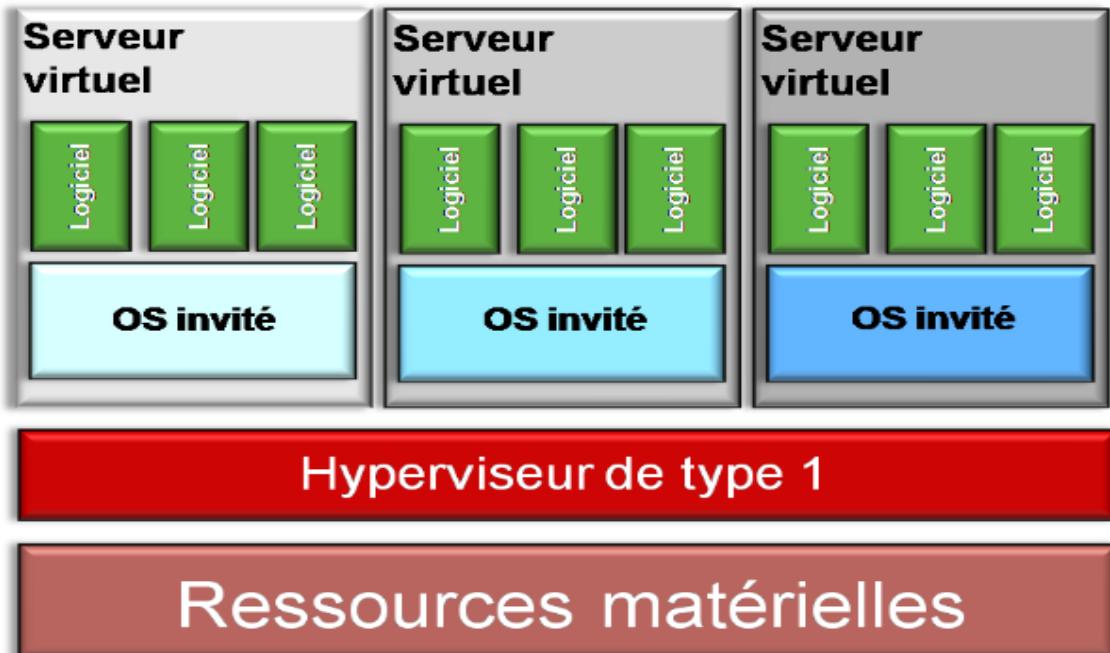
Chaque OS aura alors accès à ses propres périphériques de stockage, sa propre mémoire, sa ou ses propres interfaces réseau, son ou ses propres processeurs, chaque ressource matérielle virtualisée étant partagée avec les autres environnements.

C. La virtualisation complète :

La virtualisation est dite complète lorsque le système d'exploitation invité n'a pas conscience d'être virtualisé. Ceci est possible grâce à une virtualisation complète du matériel, réalisée par un logiciel nommé hyperviseur5. Ce dernier s'exécute à l'intérieur d'un autre système d'exploitation. Ainsi, les systèmes invités s'exécuteront en troisième niveau au-dessus du matériel.

Nous pouvons citer par exemple les logiciels Oracle VM VirtualBox ou VMware Workstation. Les performances de la virtualisation complète peuvent cependant être améliorées grâce à l'assistance par le matériel. Il simplifie la virtualisation logicielle et réduit la dégradation des performances en intégrant des instructions spécifiques à la virtualisation. Cela permet de réduire la charge de travail de l'hyperviseur et permet de consommer moins de ressources. On peut citer l'assistance à la virtualisation d'Intel : Intel VT (Virtualization Technology).

PROXMOX

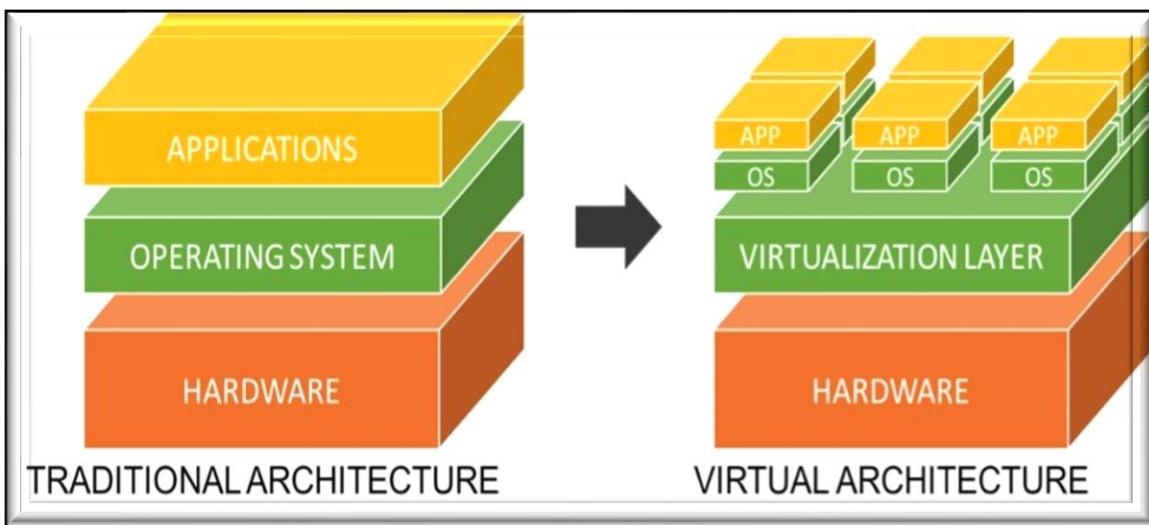


D. La virtualisation des systèmes

La virtualisation des systèmes permet à plusieurs instances d'un système d'exploitation (OS) de s'exécuter simultanément sur un seul hôte physique. Chaque système d'exploitation s'exécute dans une machine virtuelle (VM) et fonctionne comme s'il était dédié à un ordinateur physique. Il s'agit d'un logiciel intermédiaire interfaçant les machines virtuelles avec le matériel. Cette couche supplémentaire se présente sous différentes formes selon les techniques de virtualisation, qui divergent en termes de performances.

On distingue tout d'abord la virtualisation par isolation, qui n'est pas une technique de virtualisation. Le système d'exploitation hôte crée des environnements cloisonnés au niveau applicatif, les processus peuvent donc s'exécuter indépendamment les uns des autres. La virtualisation complète s'effectue en émulant entièrement le matériel. Ainsi les systèmes d'exploitation invités ont pour impression d'être les seuls à s'exécuter sur la machine physique. Enfin, la Paravirtualisation est une technique de virtualisation complète optimisée, qui présente une interface logicielle légère diminuant les coûts d'accès au matériel.

PROXMOX



E. L'hyperviseur

En informatique, un hyperviseur est une plate-forme de virtualisation qui permet à plusieurs systèmes d'exploitation de travailler sur une même machine physique en même temps.

L'hyperviseur est la couche logicielle qui s'insère entre le matériel et les différents systèmes d'exploitation. C'est un composant clé, que l'on retrouve dans la plupart des technologies de virtualisation de bas niveau.

Ainsi, par rapport au schéma de base d'un serveur distinguant le matériel, le système d'exploitation, et ses applications. L'hyperviseur peut soit gérer lui-même toutes les ressources matérielles du serveur, soit s'appuyer pour cela sur un système d'exploitation existant.

- ◆ Les hyperviseurs sont classés actuellement en deux catégories :

Type 1 : l'hyperviseur natif :

La principale caractéristique d'un hyperviseur de type 1 c'est qu'il s'installe directement sur la couche matérielle du serveur.

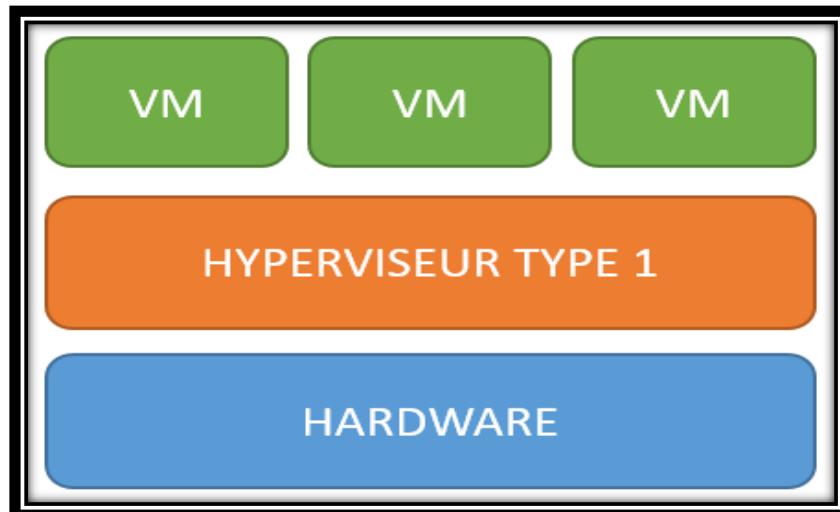
Au démarrage d'une machine physique, ces hyperviseurs prennent directement le contrôle du matériel.

Avantages : Un maximum de ressources peut être alloué aux machines virtuelles car ce type d'hyperviseur est directement lié à la couche matérielle.

Inconvénients : Il n'est possible d'exécuter qu'un seul hyperviseur à la fois. Cette problématique n'est toutefois pas vraiment impactant puisque dans la grande

PROXMOX

Majorité des cas, un seul et même hyperviseur est capable de gérer tous les applicatifs d'une entreprise.

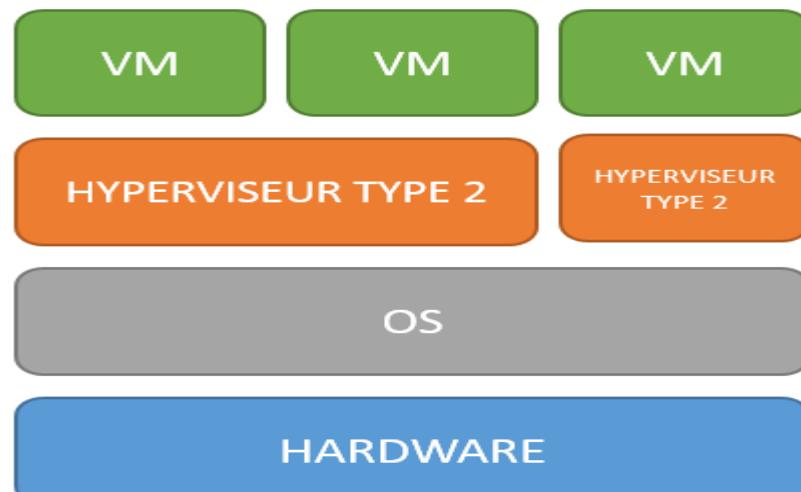


Type 2 : l'hyperviseur hosted :

Un hyperviseur de type 2 est assez comparable à un émulateur car celui-ci s'installe sur un système d'exploitation déjà en place. Il se démarre la plupart du temps comme une simple application.

Avantages : Il est possible d'exécuter plusieurs hyperviseurs en même temps car ceux-ci ne s'installent pas directement sur la couche matérielle.

Inconvénients : Ce type d'hyperviseur ne peut pas fournir autant de ressources matérielles que les hyperviseurs de type 1 puisqu'ils sont installés sur un système d'exploitation, lui-même consommateur de ressources.



Chapitre 3 : Virtualisation avec Proxmox et la Haute Disponibilité :

3.1 La virtualisation à travers la solution libre Proxmox VE :

1. Introduction :

Proxmox Virtual Environment est une solution de virtualisation open-source, basée sur une distribution GNU/Linux de type Debian. Cet environnement permet de déployer et de gérer des machines virtuelles ainsi que des conteneurs à travers une interface web et en ligne de commande.

L'intérêt de cette solution est son installation simple sur plusieurs nœuds, permettant la mise en place d'un cluster⁸. Cette fonction offre des possibilités intéressantes telles que la migration de machines virtuelles d'un serveur physique à un autre.

On dit que la migration s'effectue à froid lorsque la machine virtuelle doit être mise hors tension puis redémarrée dans un autre nœud du cluster.

Cette stratégie est coûteuse en temps de redémarrage et ne nous permet pas d'assurer une haute disponibilité du service.

L'autre stratégie consiste à migrer à chaud la machine virtuelle, en transférant la machine virtuelle d'un hôte à un autre sans être obligé de l'arrêter.

Afin de permettre un transfert de données réduisant au maximum le temps de migration, il est nécessaire d'avoir un stockage partagé entre les nœuds du cluster. L'intégration de plusieurs outils dans l'environnement Proxmox, notamment le système de stockage partagé CEPH que nous utiliserons, nous a dirigé vers ce choix.

La large communauté autour de cette solution incluant de nombreux retours d'expérience et des références institutionnelles ont confirmé notre choix.

PROXMOX

2. Présentation de Proxmox VE :

Proxmox est un environnement open source (licence aGPL) avec service de support payant s'appuyant sur l'hyperviseur Linux KVM et sur LXC.

Proxmox propose ainsi un système de gestion centralisée de machines virtuelles et de conteneurs. Cet environnement est conceptuellement équivalent à celui fourni par VMWare (Vsphere Web Client) ou Hyper-V.

Proxmox VE est un hyperviseur de Type 1, « bare metal » qui s'exécute directement sur une plateforme matérielle : cette plateforme est considérée comme un outil de contrôle de système d'exploitation. Un OS9 secondaire peut, de ce fait, être exécuté au-dessus du matériel.

L'hyperviseur de type 1 est un noyau hôte allégé et optimisé pour gérer les accès des noyaux d'OS invités à l'architecture matérielle sous-jacent.

3.2 Principe fonctionnement de Proxmox et la Haute Disponibilité :

Au moment de la publication de ce tutoriel, la version 6.0 de Proxmox ainsi que Debian 10 Buster viennent d'être publiés.

La version 6.0 apporte les changements suivants par rapport à la version 5.3 :

| Proxmox 5.3 | Proxmox 6.0 |
|-----------------------------|-----------------------------|
| | |
| Basé sur Debian stretch 9.6 | Basé sur Debian Buster 10.0 |
| Noyau 4.15.18 | Noyau 5 |
| Qemu 2.12.1 | Qemu 4.0.0 |
| LXC 3.0.2 | LXC 3.1 |
| Ceph Luminous 12.2.8 | Ceph Nautilus 14.2.x |

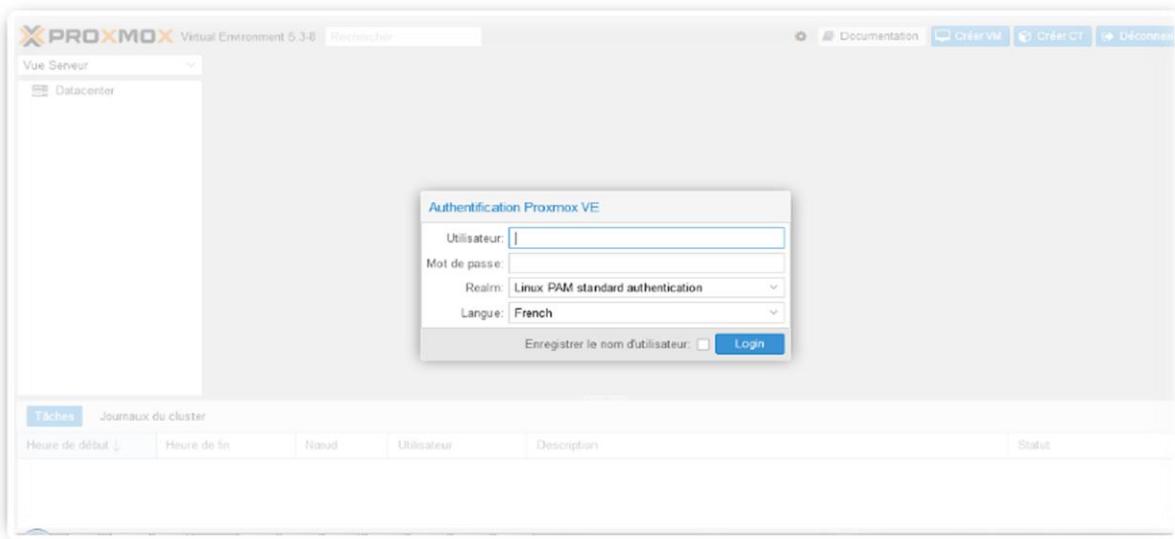
Pas de changement apparent au niveau de l'interface graphique. Les apports des nouvelles versions du noyau, de Qemu et de LXC sont intéressants, mais Debian Buster et Proxmox 6 venant de sortir.

Le changement de version majeure de Ceph peut générer des difficultés de par sa complexité en cas de mise à jour.

PROXMOX

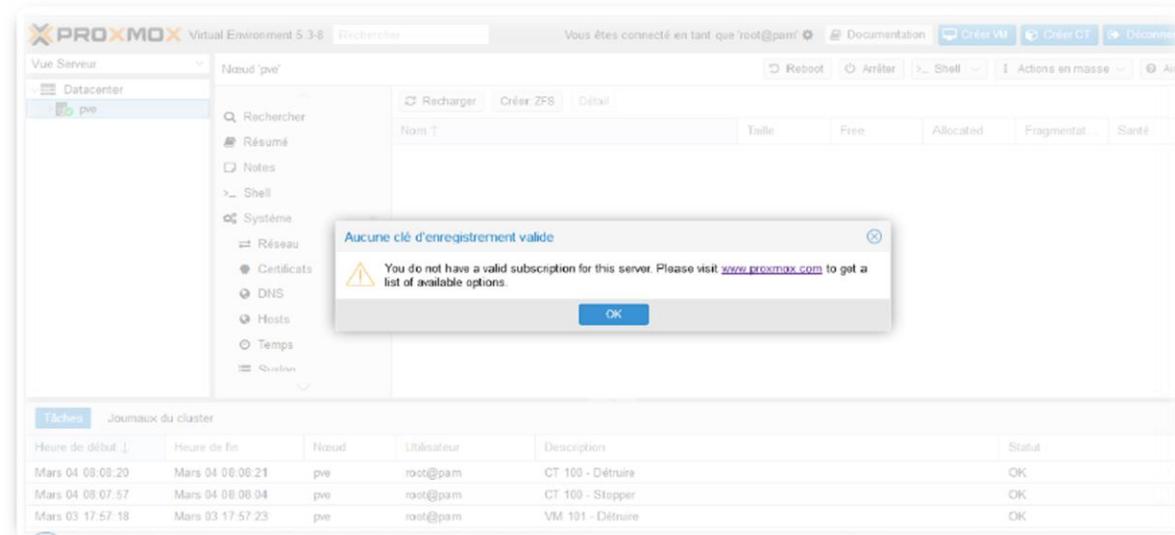
1. Découverte de l'interface d'administration :

Nous nous connectons sur l'interface web d'administration via le port 8006, comme indiqué dans le message de bienvenue :



Il faudra nous authentifier avec le mot de passe root en ayant sélectionné « Linux PAM standard authentication ».

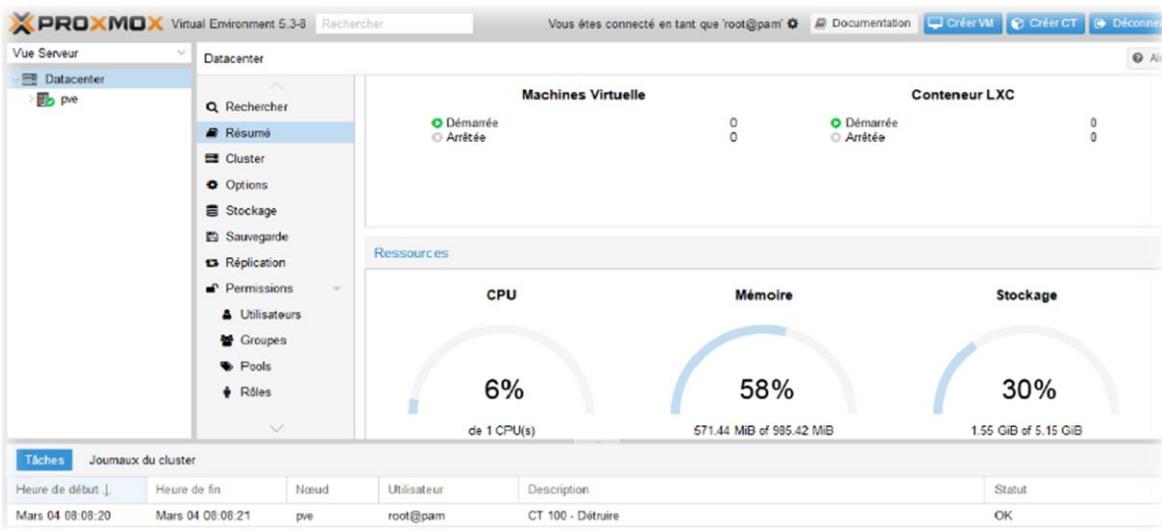
Nous aurons ensuite un message « Aucune clé d'enregistrement valide », spécifique à la version gratuite de Proxmox :



PROXMOX

Il suffira de cliquer OK.

Une fois connecté, Nous aurons l'écran principal de gestion de votre Datacenter :



Nous voyons la liste des serveurs du Datacenter (dans ce cas uniquement srv1).

Nous verrons ensuite les espaces disques des serveurs pour srv1 par défaut en cliquant sur srv1 (ou en allant sur stockage) :

- local.
- local-lvm (présent uniquement depuis l'installation avec le live-cd ou si nous avons installé un LVM (voir partie ajout LVM)).

La vue par défaut présente l'arborescence du Datacenter, l'onglet Datacenter affichant :

- **Résumé** : va afficher le nombre de noeuds, de VM, de conteneurs, va donner également un état de la charge du matériel.
- **Cluster** : permet de créer/rejoindre un cluster.
- **Options** : réglages d'options telles que la disposition du clavier, l'adresse mail de l'administrateur.
- **Stockage** : permet d'ajouter, supprimer, éditer les stockages disponibles (ce point sera étudié au **chapitre de pratique**).
- **Sauvegarde** : sera vu en détail au **chapitre de pratique**.
- **RéPLICATION** : sera vu au **chapitre de pratique**.
- **Permissions** :

PROXMOX

- Utilisateurs : pour ajouter des utilisateurs (stockés dans le fichier `/etc/pve/user.cfg` pour les utilisateurs pve locaux).
 - Groupes : création de groupes pour les utilisateurs.
 - Authentification : gère les méthodes d'authentification utilisées par Proxmox :
 - Par défaut :
 - pam : Authentification PAM standard de Linux.
 - pve : Authentification interne à Proxmox.
 - Méthodes supplémentaires :
 - Active Directory.
 - Serveur LDAP.
- **HA** : pour le contrôle de la haute disponibilité (fonction avancée et nécessitant plusieurs serveurs).
- **Pare-feu** : permet l'ajout de règles IPTables (un champ commentaire est également présent).
- **Support** : accessible uniquement dans la version payante.

Un clic sur srv1 permettra d'obtenir les options suivantes :

- **Résumé** : détail sur le serveur proprement dit, plus précis que le résumé général du datastore et filtrant au niveau serveur dans le cas où il y en a plusieurs.
- **notes** : un espace où vous pouvez inscrire ce que vous souhaitez.
- **Shell** : ouvre une console sur le serveur via VNC.
- **Système** :
- réseau : permet la gestion des cartes réseau en interface graphique.
 - certificats : le chargement de vos propres certificats via l'icône « upload » vous permettra de ne plus avoir l'avertissement « connexion non sécurisée » dans votre navigateur.
 - DNS : les serveurs DNS pour l'accès externe.
 - Hosts : permet la modification du fichier `/etc/hosts`.
 - temps : permet le réglage de la timezone.
 - syslog : accès au contenu de syslog.
- **Mise à jour** : permet la mise à jour des paquets de l'OS.
- **Disques** : affiche les différents disques physiques du serveur (accès à l'état S, M, A, R, T,) :

PROXMOX

- LVM : accès à la gestion des Volume Group LVM (si LVM présent).
- LVM-Thin : Accès aux Thinpools.
- Répertoire.
- ZFS.

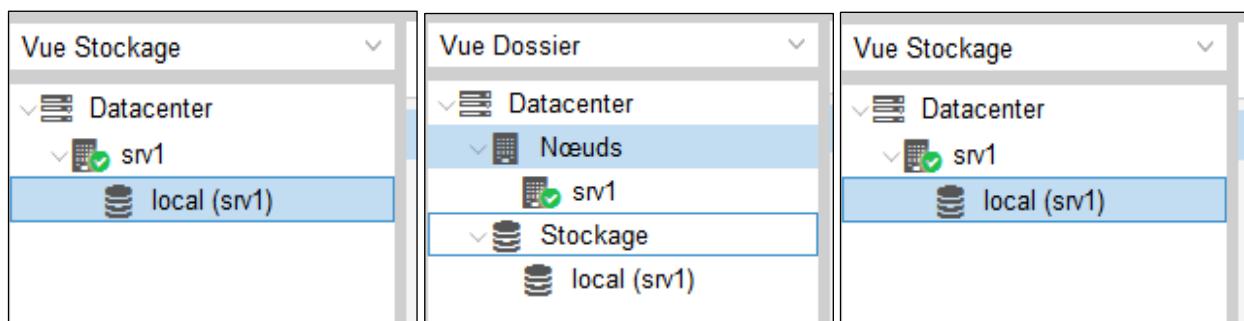
➤ **Résumé.**

➤ **Contenu.**

➤ **Permissions.**

Dans la partie gauche, vous pourrez avoir différentes arborescences de vues pour ces affichages :

- vue serveur (par défaut).
- vue dossier.
- vue stockage.
- vue pool.



Nous allons aborder le stockage et la gestion du réseau dans Proxmox.

2. le stockage dans Proxmox :

Pour le stockage, Proxmox gère des magasins de données. Les différents magasins de données disponibles dans Proxmox sont :

- les dossiers standards.
- les points de montage SMB, NFS.
- les volumes LVM pour les disques virtuels des VM (par défaut si installé depuis l'ISO).
- les périphériques SAN via le protocole iSCSI.
- les volumes **ZFS**.
- les espaces de stockage distribués Ceph.

PROXMOX

- les espaces de stockage distribués et/ou répliqués GlusterFS.

Les magasins de données vont contenir :

- des VM/conteneurs.
- des ISO ou templates de conteneurs.

Si LVM est présent, les VM seront automatiquement stockées dans des volumes LVM (un volume LVM par disque virtuel) dans l'espace de stockage local-lvm, les ISO ou templates seront eux stockés dans l'espace de stockage local.

Un magasin de données **Ceph** permettra le stockage de tout élément.

Le type de stockage détermine ce qui peut être mis dedans. Ce que peut contenir un magasin de données peut être modifié dans le menu Datacenter → stockage → éditer. Il est possible d'ajouter/retirer certains éléments d'un type de stockage.

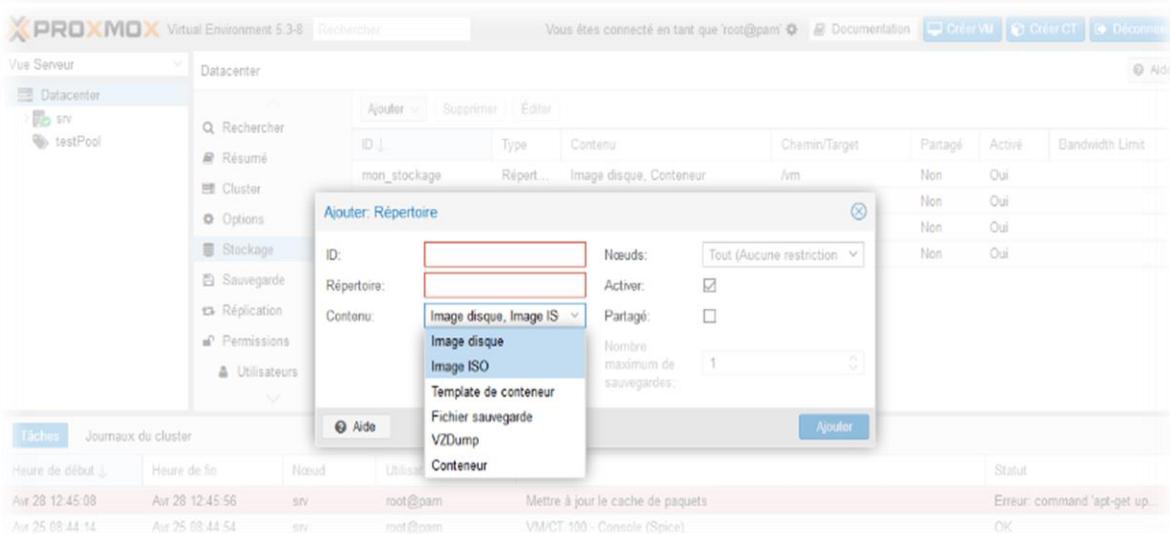
Exemple d'ajout d'un dossier/point de montage :

| Répertoire | Type | Contenu | Chemin/Target | Partagé | Activé | Bandwidth Limit |
|----------------|-----------|-----------------------------------|---------------|---------|--------|-----------------|
| LVM | Répert... | Image disque, Conteneur | /vm | Non | Oui | |
| LVM-Thin | LVM-Thin | Image disque, Conteneur | | Non | Oui | |
| NFS | Répert... | Fichier sauvegarde VZDump, Ima... | /var/lib/vz | Non | Oui | |
| CIFS | Répert... | Fichier sauvegarde VZDump, Ima... | /svgs | Non | Oui | |
| GlusterFS | | | | | | |
| iSCSI | | | | | | |
| CephFS | | | | | | |
| RBD | | | | | | |
| ZFS over iSCSI | | | | | | |
| ZFS | | | | | | |

Il faudra alors renseigner :

- l'**id** : il s'agit du nom de l'espace de stockage.
- le **répertoire** : le point de montage.
- le **contenu** : le type d'élément contenu par l'espace de stockage (images, sauvegardes, templates, etc.).
- le **nœud** : uniquement nécessaire en cas de présence de plusieurs hyperviseurs.

PROXMOX



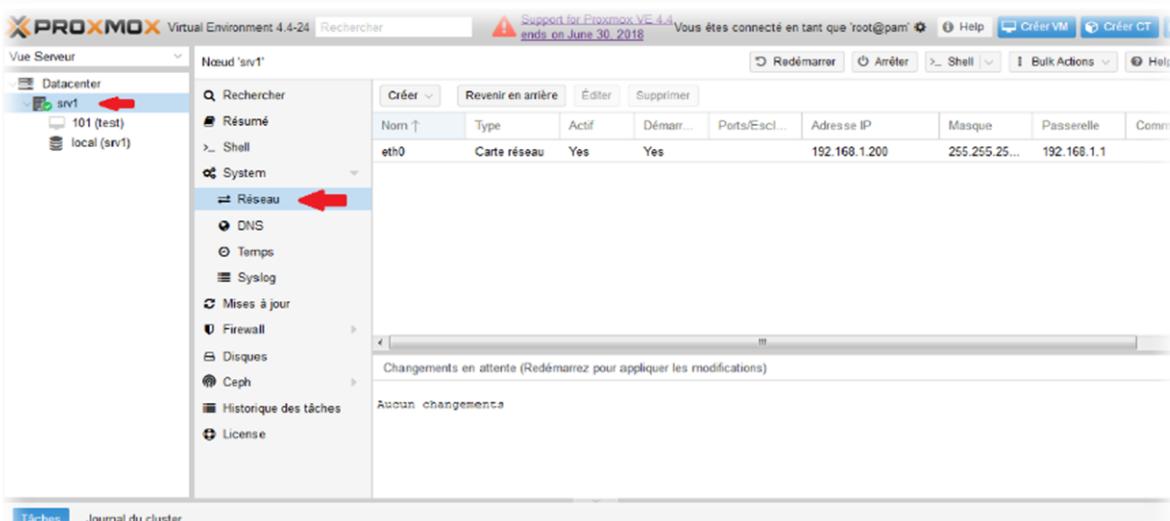
The screenshot shows the Proxmox VE 5.3-8 interface. On the left, there's a sidebar with 'Vue Serveur' and 'Datacenter' sections. In the main area, there's a table for storage pools and a modal window for adding a new storage pool ('Ajouter Répertoire'). The modal has fields for 'ID', 'Répertoire', 'Content' (set to 'Image disque, Image IS'), 'Activer' (checkbox checked), 'Partagé' (checkbox unchecked), and 'Nombre maximum de sauvegardes' (set to 1). At the bottom right of the modal is a blue 'Ajouter' button.

Selon le type d'espace de stockage, les champs présents seront adaptés à la situation. Exemples :

- export pour montage **NFS**.
- point de partage pour montage **CIFS**.
- choix du Volume Group pour **LVM**.
- Volume Name pour **GlusterFS**.

3. Le réseau dans Proxmox :

La gestion des cartes réseau se gère depuis système → réseau :



The screenshot shows the Proxmox VE 4.4-24 interface. The left sidebar shows a tree structure with 'sr1' selected. Under 'sr1', '101 (test)' and 'local (sr1)' are listed. The 'Réseau' tab is highlighted with a red arrow. The main panel shows a table of network interfaces:

| Nom | Type | Actif | Démarr... | Ports/Esci... | Adresse IP | Masque | Passerelle | Comm |
|------|--------------|-------|-----------|---------------|---------------|---------------|-------------|------|
| eth0 | Carte réseau | Yes | Yes | | 192.168.1.200 | 255.255.25... | 192.168.1.1 | |

Below the table, a message says 'Changements en attente (Redémarrez pour appliquer les modifications)'. At the bottom, it says 'Aucun changement'.

PROXMOX

Pour pouvoir utiliser le réseau avec une VM, il nous faudra créer un Bridge ou utiliser un Bond (que nous verrons un peu plus bas). Le Bridge est une carte.

virtuelle permettant la création d'un pont entre une carte virtuelle, utilisée par une ou plusieurs VM et une carte réelle. En cas de présence de plusieurs cartes réseau physiques (classique sur un serveur), certaines VM pourront utiliser une carte précise pendant que d'autres VM pourront utiliser une autre carte.

Si nous avons installé Proxmox depuis le CD, l'installation aura automatiquement créé un bridge. Si vous n'avez pas installé Proxmox depuis le CD, il nous faudra créer un Linux Bridge (ou un bond). Nous pouvons affecter une IP ou non au bridge.

Création du Linux Bridge

Pour créer le bridge, il faudra cliquer sur « créer » et sélectionner « Linux Bridge » dans le menu serveur → système → réseau.

Créer: Linux Bridge

| | | | |
|----------------------|---------------|------------------------|-------------------------------------|
| Nom: | vmbr0 | Démarrage automatique: | <input checked="" type="checkbox"/> |
| Adresse IP: | 192.168.1.201 | VLAN aware: | <input type="checkbox"/> |
| Masque: | 255.255.255.0 | Ports membres: | |
| Passerelle: | | Commentaire: | |
| Adresse IPv6: | | | |
| Longueur du préfixe: | | | |
| Passerelle: | | | |

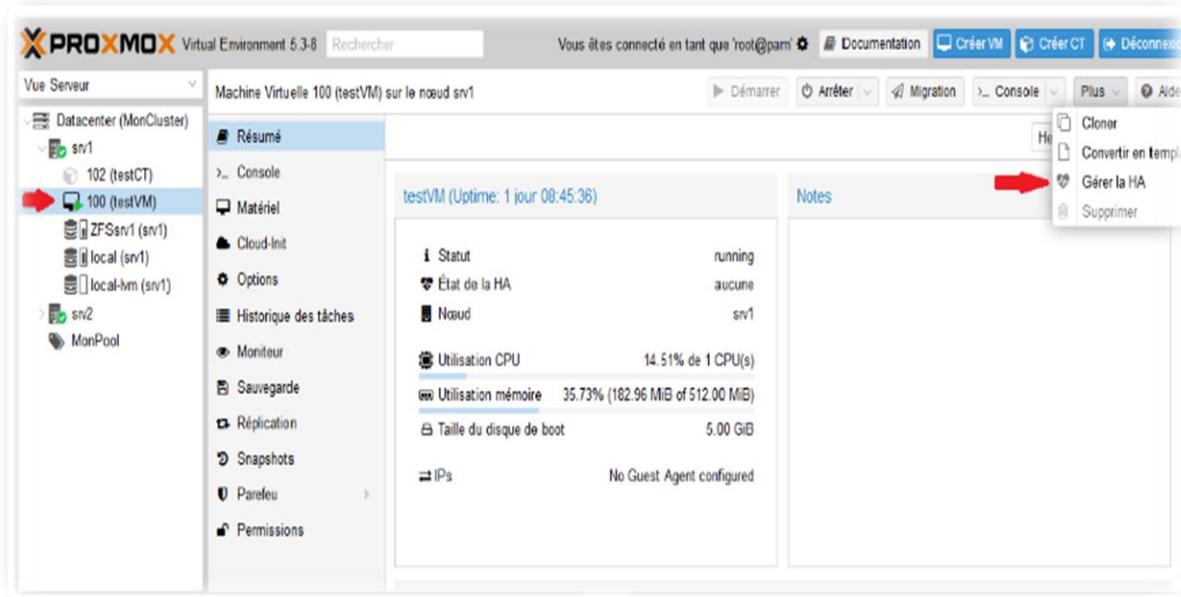
Créer

Le bridge sera connecté sur la première interface réseau. Nous pourrons manuellement affiner les réglages dans le fichier [*/etc/network/interfaces*](#).

4. Activation de la haute disponibilité sur une VM :

Pour activer la haute disponibilité pour une VM, il vous faudra depuis celle-ci cliquer sur le menu « Plus », puis gérer la HA (pour high availability) :

PROXMOX



Nous aurons l'écran suivant :

Ajouter: Ressource: Conteneur/Machine Virtuelle

| | | | |
|--------------------------------|---|-------------------------|----------------------|
| VM: | 100 | Groupe: | <input type="text"/> |
| Nombre maximum de redémarrage: | 1 | Request State: | started |
| Max. Relocate: | 1 | | |
| Commentaire: | <p>At least three quorum votes are recommended for reliable HA.</p> | | |
| Aide | | Ajouter | |

Éléments à renseigner :

- nombre maximum de redémarrages : nombre maximum d'essais de redémarrage après échec.
- Max relocate : nombre maximum d'essais de déplacement après échec de démarrage.
- groupe : sera vue dans la section suivante.

PROXMOX

- Request state : état attendu, déclenchement de traitement si l'état n'est pas adéquat.
- Started : le système va tenter de garder la VM à l'état démarré.
- Stopped : le système va tenter de garder la VM à l'état stoppé.
- Disabled : le système va tenter de garder la VM à l'état stoppé sans tenter de déplacement.
- Ignored : le système va ignorer la VM pour le traitement de la haute disponibilité.

5. Gestion de la haute disponibilité :

La gestion globale de la haute disponibilité s'effectuera dans Datacenter → HA :

The screenshot shows the Proxmox VE 5.3-8 interface. On the left, there's a sidebar titled "Vue Serveur" with a tree view of the cluster. The "Datacenter" node is expanded, showing "srv1" and "srv2". Under "srv1", there are entries for "101 (testVM2)", "local (snr2)", and "local-lvm (snr2)". There's also a "MonPool" entry. The main panel has two sections: "Statut" and "Ressources". The "Statut" section shows a table with columns "Type" and "Statut". It includes rows for "quorum" (OK), "master" (srv1 active, Tue Jul 2 07:42:23 2019), and two "lrm" entries (both active, Tue Jul 2 07:42:25 2019 and Tue Jul 2 07:42:26 2019). The "Ressources" section has tabs for "Ajouter", "Éditer", and "Supprimer". A table below shows resources with columns ID, Etat, Nœud, Nombre m..., Max. Reloc..., Groupe, and Description. It lists "vm:100" (started, srv1) and "vm:101" (starting, srv2).

Dans l'écran ci-dessus, nous voyons qu'il y a un **quorum** (Quorum OK), que le serveur maître (master) est srv1, que le LRM (Local Ressource Manager) est actif sur srv1 et idle (en attente) sur srv2.

La partie Ressource présente les VM intégrées à la gestion de la haute disponibilité.

6. Quorum :

Le Quorum est généralement un système de vote qui permet de déterminer quel nœud peut avoir accès aux ressources. Il est utilisé pour éviter le phénomène de « split-brain » qui apparaît lorsque plusieurs nœuds du cluster essayent de prendre le contrôle de celui-ci. Ceci peut entraîner de la duplication de services, voire de la corruption de données. Sous Proxmox, ce service est effectué par le programme

PROXMOX

corosync. Il attribue un Seul vote par nœud. Le quorum correspond au nombre minimum de votes nécessaire pour permettre aux nœuds le contrôle du cluster. Ainsi, il est nécessaire de déployer un cluster contenant au minimum 3 nœuds pour obtenir un quorum pertinent dans le cas de la Haute Disponibilité.

7. Group de la haute disponibilité :

La notion de groupe va faciliter l'administration. La gestion de groupe se fera en allant dans le menu sous HA :

The screenshot shows the Proxmox Web Interface. On the left, there's a sidebar with 'Vue Serveur' and 'Datacenter' sections. Under 'Datacenter', there are links for Sauvegarde, Réplication, Permissions, Utilisateurs, Groupes (which has a red arrow pointing to it), Pools, Rôles, Authentification, HA, and Fencing. In the center, a modal window titled 'Créer: Groupe HA' is open. It has fields for 'ID' (set to 'test'), 'restricted' (unchecked), and 'nofailback' (unchecked). Below these are 'Commentaire:' and a table of host utilization:

| Nœud ↑ | Utilisation mémoire % | Utilisation CPU | Priority |
|--------|-----------------------|-----------------|----------|
| srv1 | 67.2 % | 20.7% of 1CPU | ▼ |
| srv2 | 34.9 % | 12.9% of 1CPU | ▼ |

At the bottom right of the modal is a 'Créer' button. The main interface below the modal shows 'Tâches' and 'Journaux du cluster' sections.

Ces réglages vous permettront de restreindre par exemple des VM à pouvoir par exemple se trouver sur deux serveurs parmi trois, de gérer prioritairement sur quel serveur elles doivent se trouver.

Restricted : cette option va justement servir à restreindre la présence de VM dans les nœuds sélectionnés. L'option devient une contrainte.

Nofailback : va empêcher la migration automatique vers le nœud à plus haute priorité.

Ces fonctions permettent d'affiner les réglages et d'ajouter la répartition de charge à la disponibilité garantie.

8. Fencing :

Le Fencing est le mécanisme permettant de s'assurer qu'une VM ne puisse pas être démarrée sur deux nœuds.

PROXMOX

Ceci peut se présenter sous forme d'un dispositif matériel nommé watchdog. Si vous n'en avez pas, Proxmox utilisera par défaut le linux kernel softdog, un watchdog logiciel disponible en userland.

En général, un watchdog fonctionne de la façon suivante : si un compteur n'est pas mis à jour, le watchdog va déclencher un reset de la machine, ceci afin de détecter un éventuel blocage.

9. Stockage partagé

Dans l'environnement Proxmox, chaque nœud du cluster dispose d'un stockage local et d'un stockage partagé. Les images des machines virtuelles KVM ou les conteneurs LXC sont stockés sur le stockage partagé.

L'intérêt de ce type de stockage est la possibilité de migrer à chaud les machines virtuelles/conteneurs sans temps d'arrêt, étant donné que tous les nœuds du cluster ont un accès direct aux images des VMs¹⁴.

Il est alors nécessaire d'utiliser deux réseaux indépendants pour supporter la montée en charge : un réseau pour le trafic des VMs et un second consacré au trafic du stockage. Le service pve-ha-manager gère l'ensemble des services du cluster pour assurer la haute disponibilité. Les machines virtuelles doivent être gérées grâce à ce service pour éviter un temps d'arrêt durant leur migration.

10. Haute disponibilité :

Comme vu dans le précédent chapitre, la haute disponibilité permet de pallier automatiquement la perte d'un nœud, le système allant automatiquement basculer les VM d'un nœud à l'autre en cas de perte de leur nœud d'origine.

Une fois le problème résolu, il vous restera à éventuellement rebasculer les VM sur leur nœud initial, d'éventuellement refaire le réglage au cas où le nœud rétabli devient une machine de secours (pour éviter de redéplacer les VM).

11. Les outils en ligne de commande :

Voici un résumé des outils disponibles en ligne de commande.

Les principales commandes spécifiques Proxmox :

- **pvecm** : PVE Cluster Manager.
- **pveam** : gestion des templates.
- **pveceph** : gestion ceph.

PROXMOX

- **pvesr** : gestion de réplication.
- **vzdump** : sauvegarde de VM et conteneurs (vzdump help).
- **qmrestore** : restauration de VM.
- commandes **Proxmox** spécifiques à LXC : **pct** (pct help).

12. Conclusion :

Certaines fonctionnalités sont absentes (mais leur fréquence d'utilisation ne nécessite peut-être pas l'accès depuis l'interface graphique). Nous constaté des problèmes de stabilité, peut-être dus à nos environnement de test, comme l'upload d'ISO, ou l'échec apparent de certaines fonctionnalités (faux positif lié à un problème de rafraîchissement).

Il ne faut pas hésiter à utiliser la ligne de commande en complément.

Proxmox offre toutes les fonctionnalités que l'on peut attendre d'un gestionnaire de machines virtuelles (sauvegarde, réplication, haute disponibilité). Sans aucun doute, **Proxmox** est donc un concurrent **libre** sérieux pour **VMWare** et **Hyper-V**. L'ouverture du code et le prix sont inévitablement un gros avantage pour **Proxmox**.

PROXMOX

Chapitre 1 : Métrise de proxmox :

1.1 Installation de Proxmox

Proxmox peut être installé soit à partir d'un fichier ISO, soit à travers le gestionnaire de paquets de votre système déjà en place. Proxmox n'est disponible que pour Debian.

Sur chacun des serveurs, ne prend que quelques minutes. Lors de cette dernière, très peu de paramètres sont à renseigner : l'adresse IP, le nom du serveur, la langue et le mot de passe de l'administrateur : « root » pour linux. Proxmox se charge d'installer les paquets nécessaires avec les dépendances. C'est-à-dire les paquets correspondant à la virtualisation avec KVM et les éléments nécessaires à la mise en place d'un serveur WEB afin de pouvoir accéder à l'interface de gestion des machines virtuelles.

Voici l'écran de démarrage :

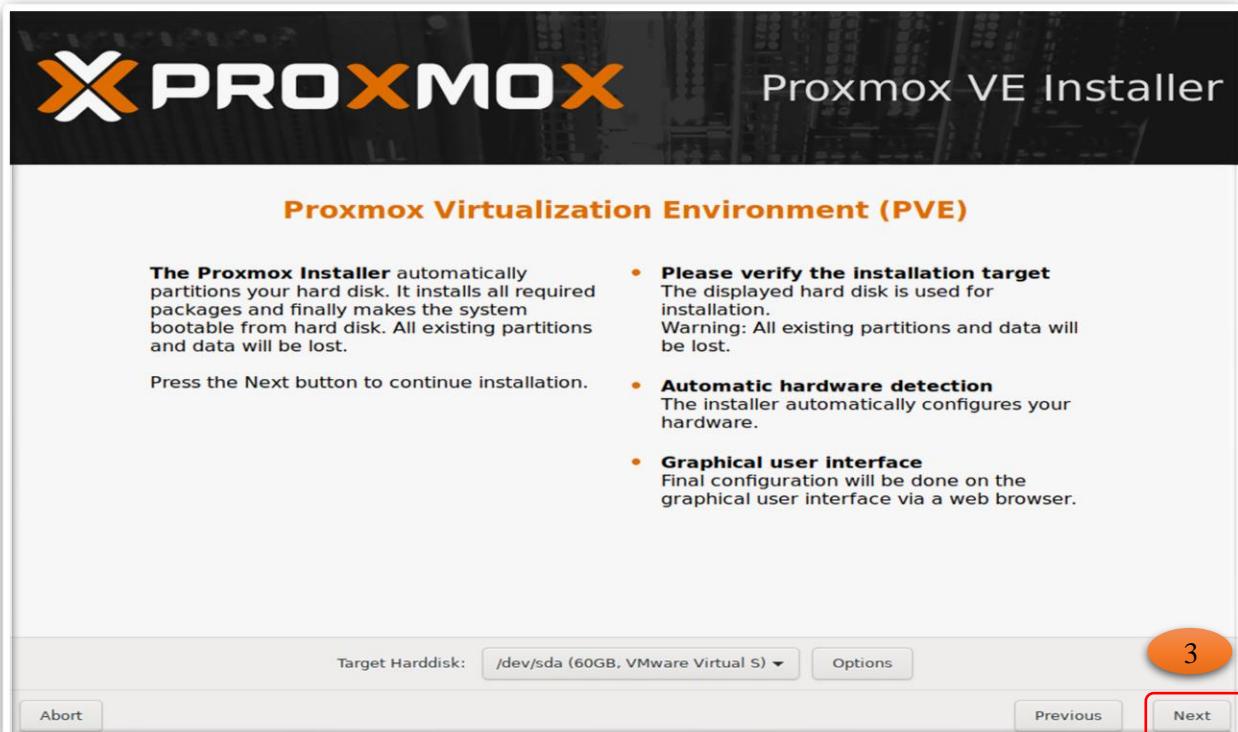


PROXMOX

Après avoir démarré, nous aurons le premier écran d'installation suivant :



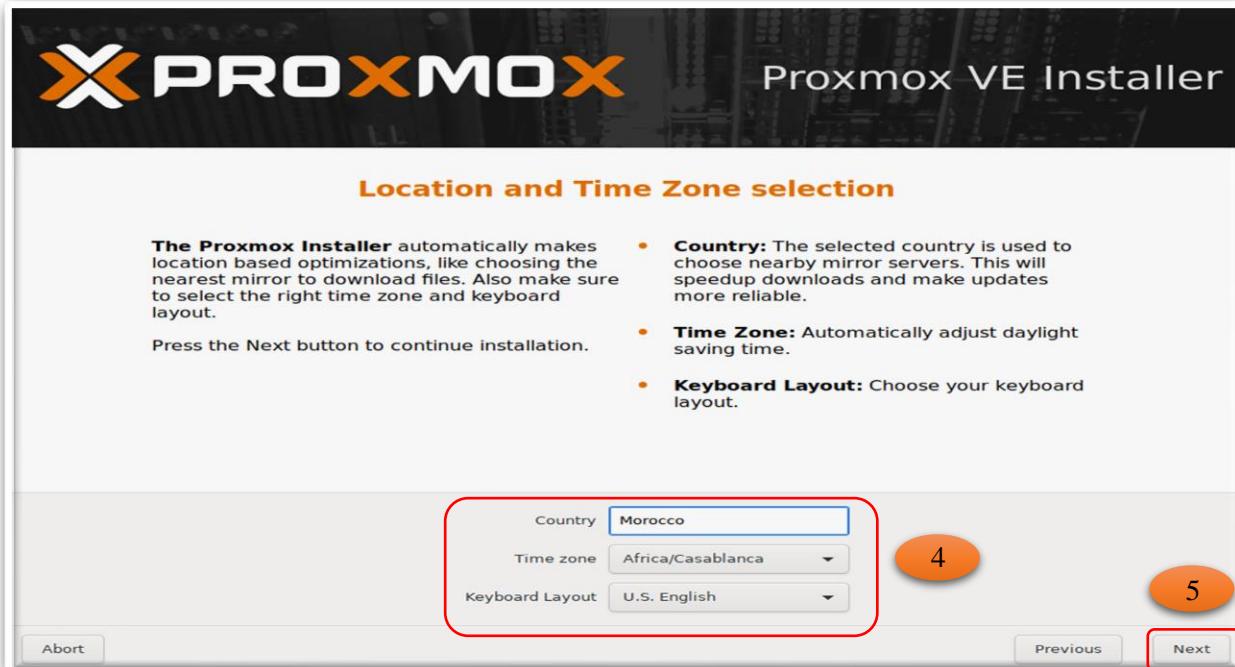
Puis :



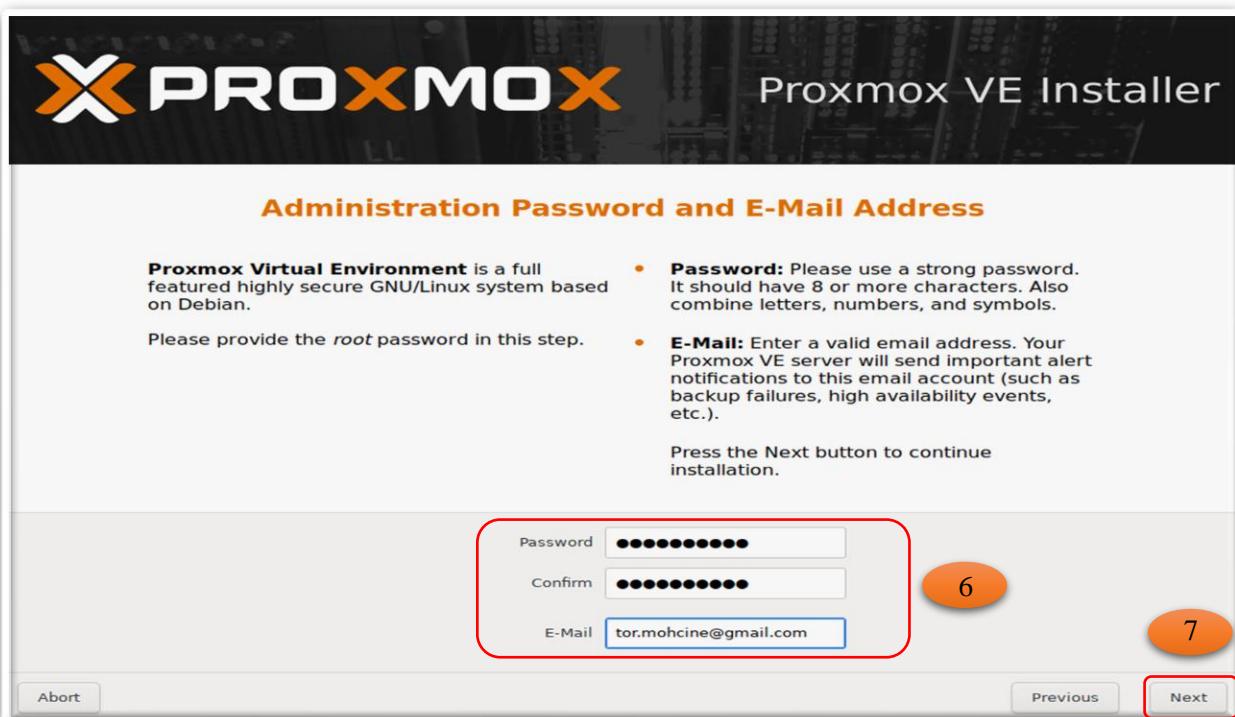
PROXMOX

Le bouton options permettra de personnaliser le partitionnement.

L'écran suivant permettra de sélectionner notre pays, le choix du pays remplit automatiquement les champs « timezone » et la disposition des touches du clavier (On peut personnaliser notre guise).

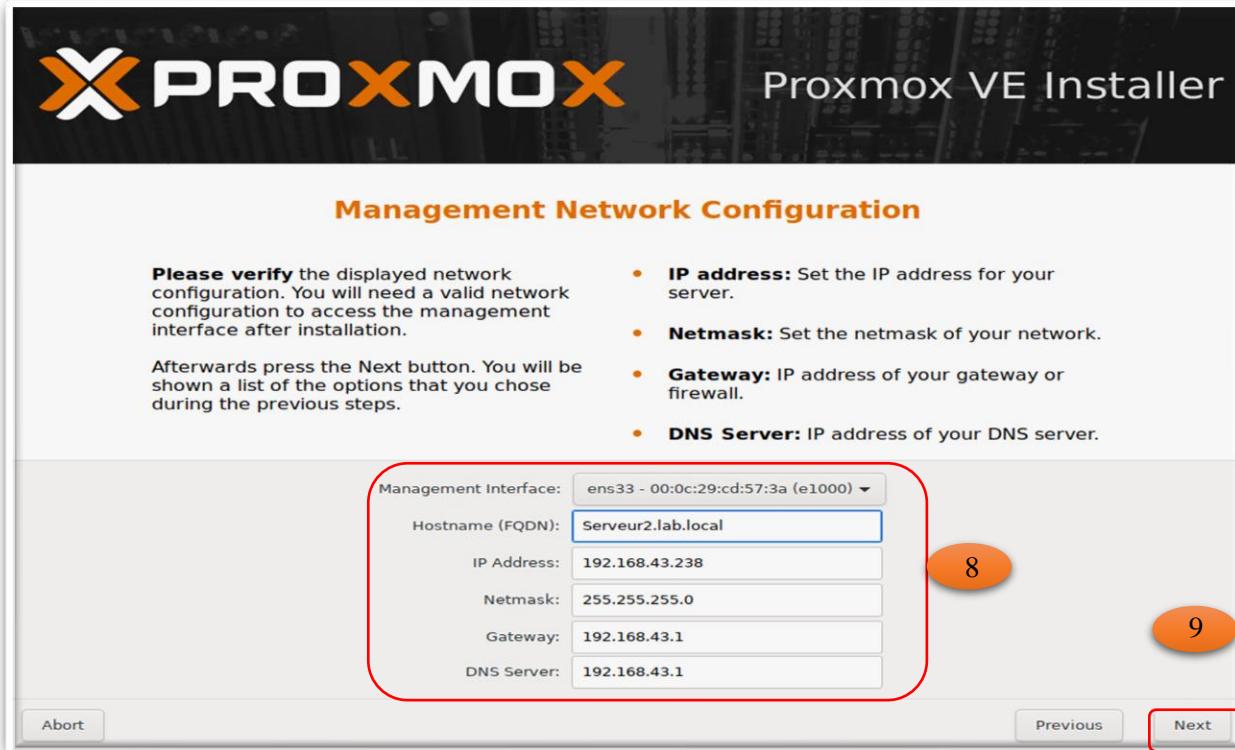


Il faudra ensuite renseigner le mot de passe et l'adresse mail de l'administrateur :

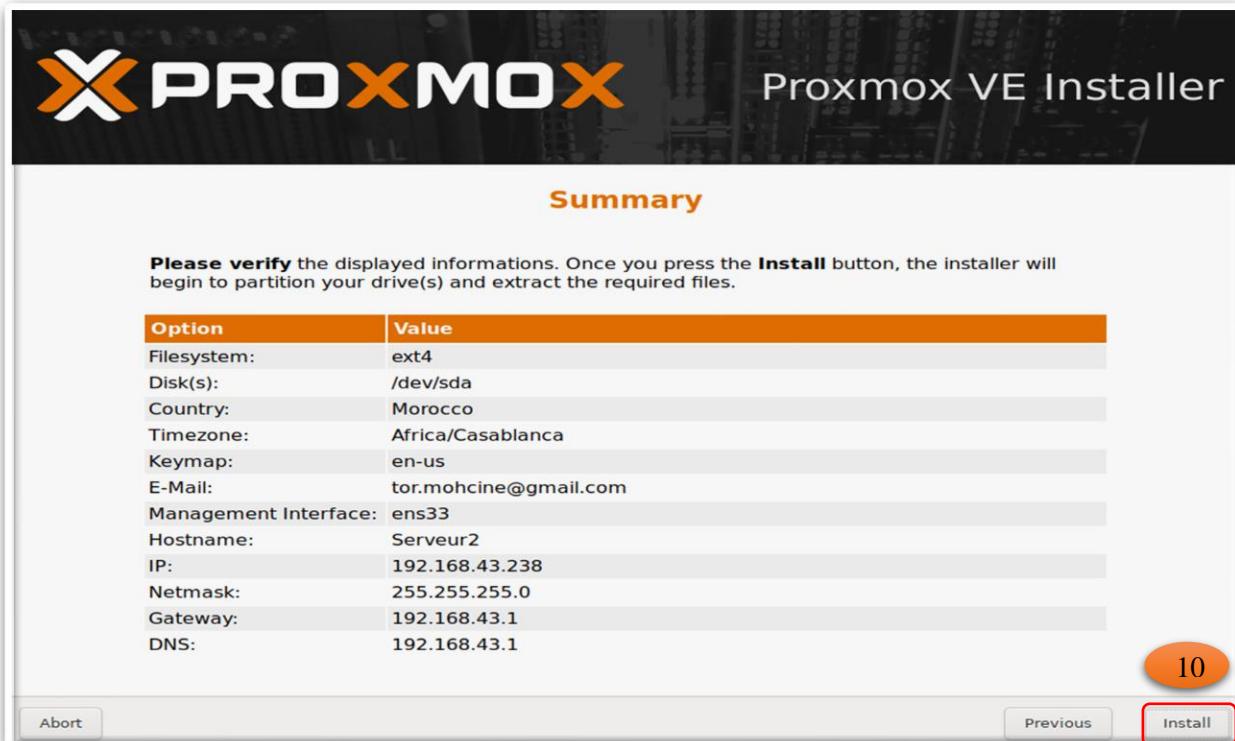


PROXMOX

Ensuite personnaliser le nom de machine (FQDN) ainsi que les réglages réseau :



Voila :

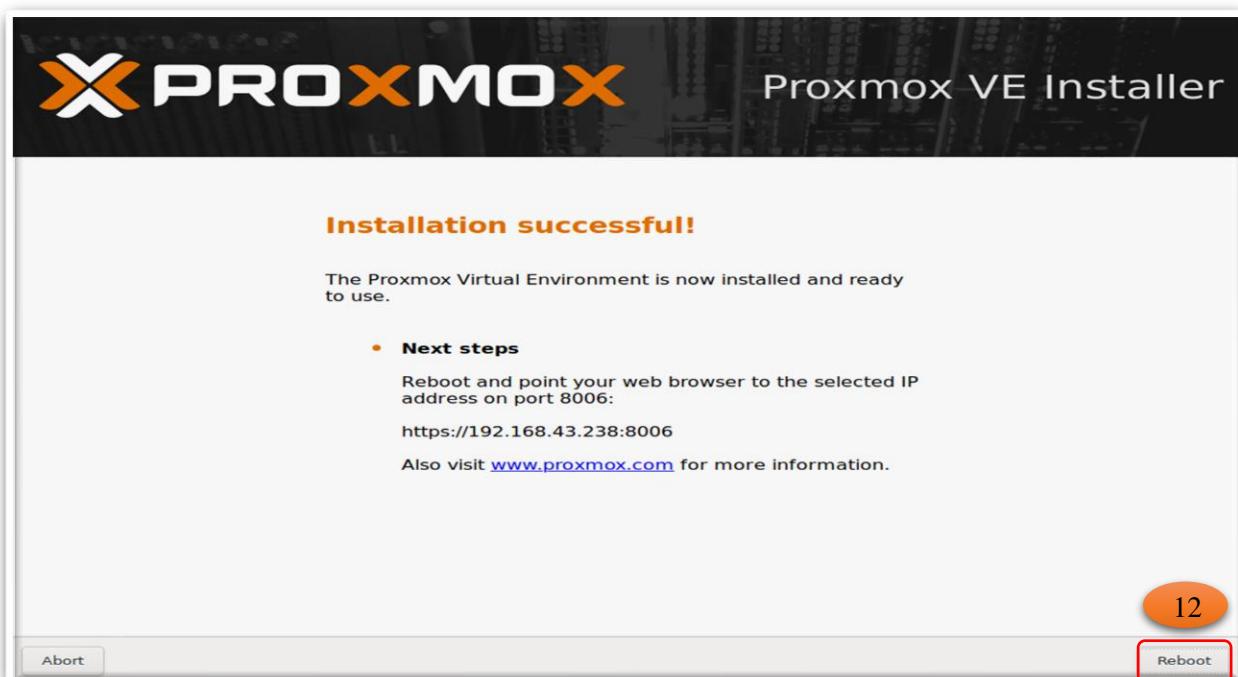


PROXMOX

L'installation s'effectue ensuite automatiquement :



L'écran en fin d'installation nous aurons l'écran suivant nous rappelant l'URL d'accès à l'interface d'administration :



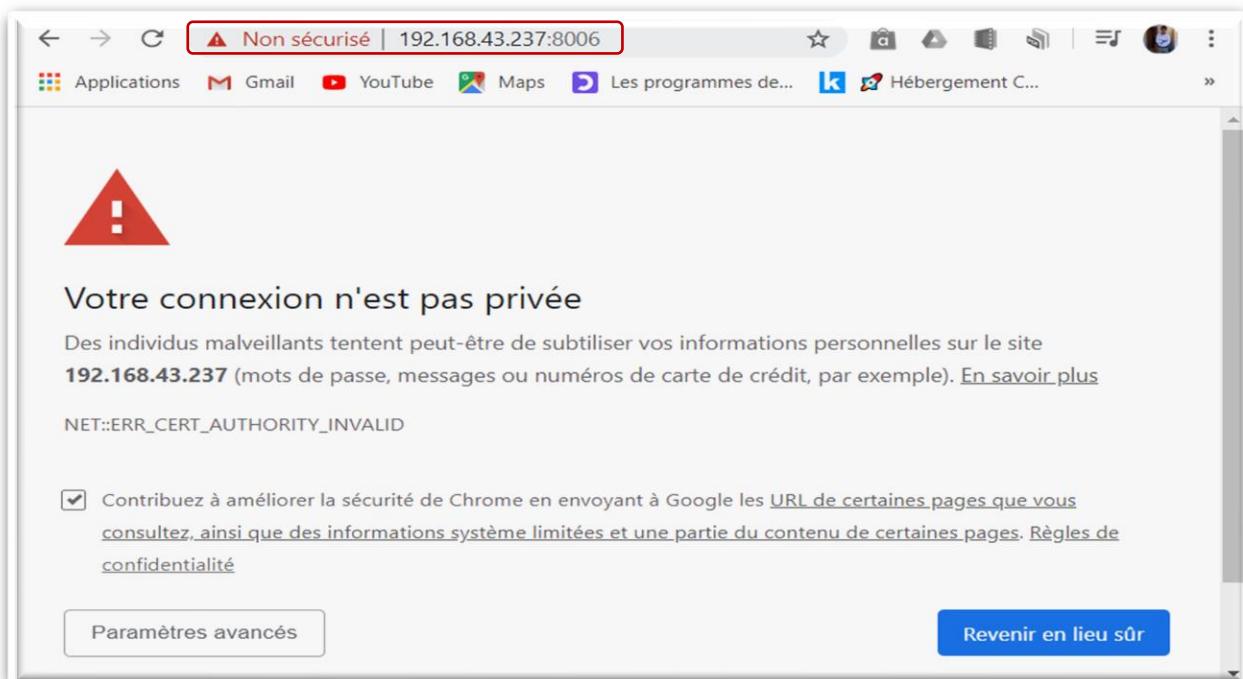
PROXMOX

Quand l'installation est terminée on va cliquez sur Reboot pour le serveur il va démarrer automatiquement ensuite. J'ai tapé mon nom d'utilisateur et mon mot de passe.

```
Welcome to the Proxmox Virtual Environment. Please use your web browser to
configure this server - connect to:
https://192.168.43.237:8006/
-----
[redacted]
Serveur1 login: root
Password:
Last login: Sat Mar 28 19:07:49 +01 2020 from 192.168.43.232 on pts/1
Linux Serveur1 5.3.10-1-pve #1 SMP PVE 5.3.10-1 (Thu, 14 Nov 2019 10:43:13 +0100) x86_64
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.

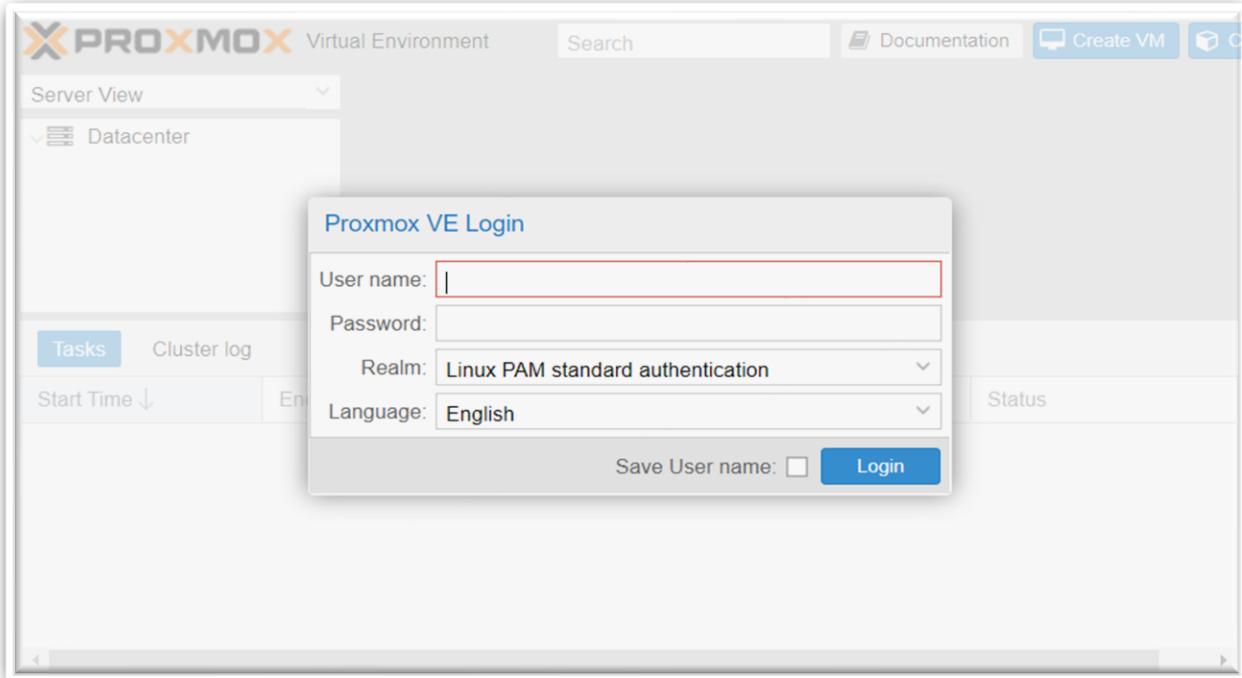
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@Serveur1:~#
```

Finalement j'ai ouvert mon serveur via un navigateur web avec son adresse IP.

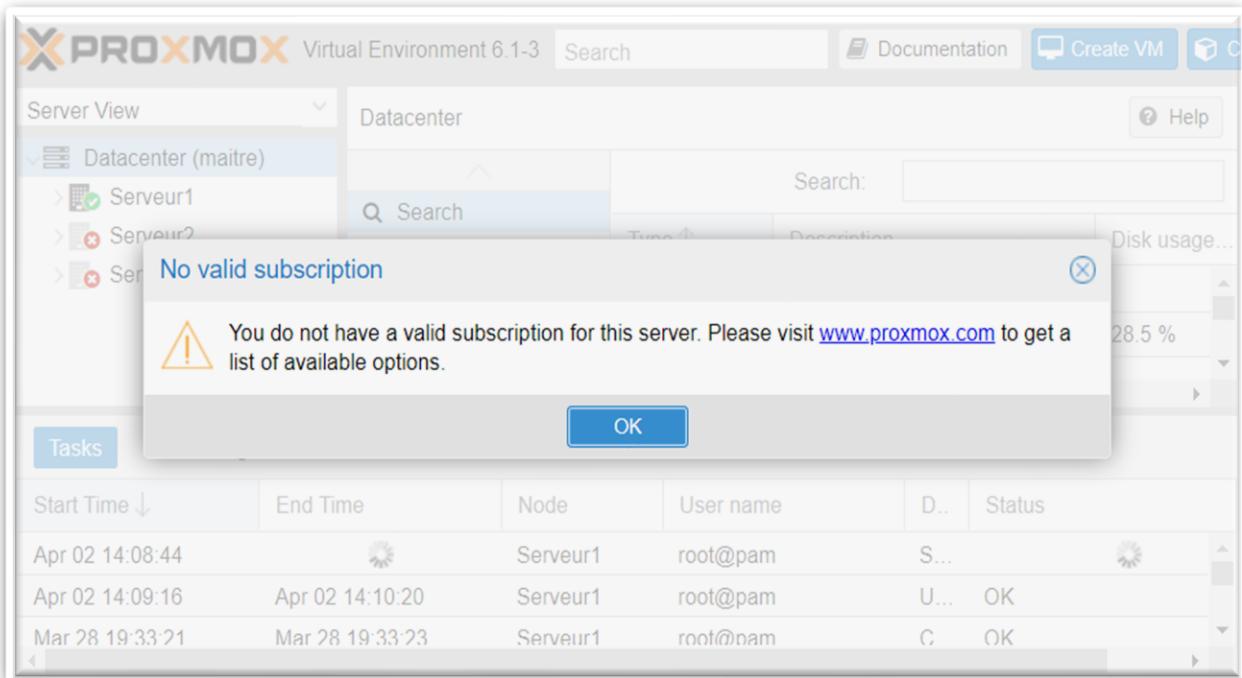


PROXMOX

L'interface Web de Proxmox écoute par défaut en https sur le port 8006. Il n'y a pas de réglage pour changer ce port. Nous pourrons contourner ce comportement depuis un reverse proxy. Puis nous nous connectons sur l'interface web d'administration via le port 8006, comme indiqué dans le message de bienvenue :



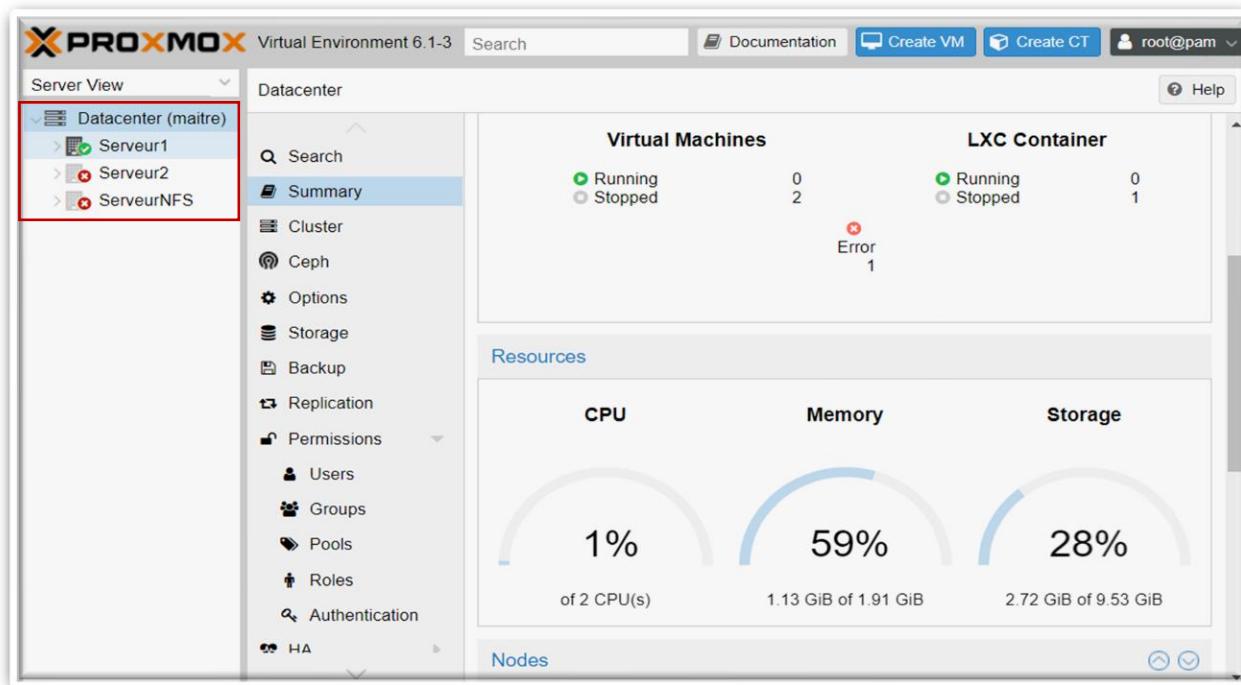
Il faudra nous authentifier avec le mot de passe root en ayant sélectionné « Linux PAM standard authentication ». Nous aurons ensuite un message « Aucune clé d'enregistrement valide », spécifique à la version gratuite de Proxmox :



PROXMOX

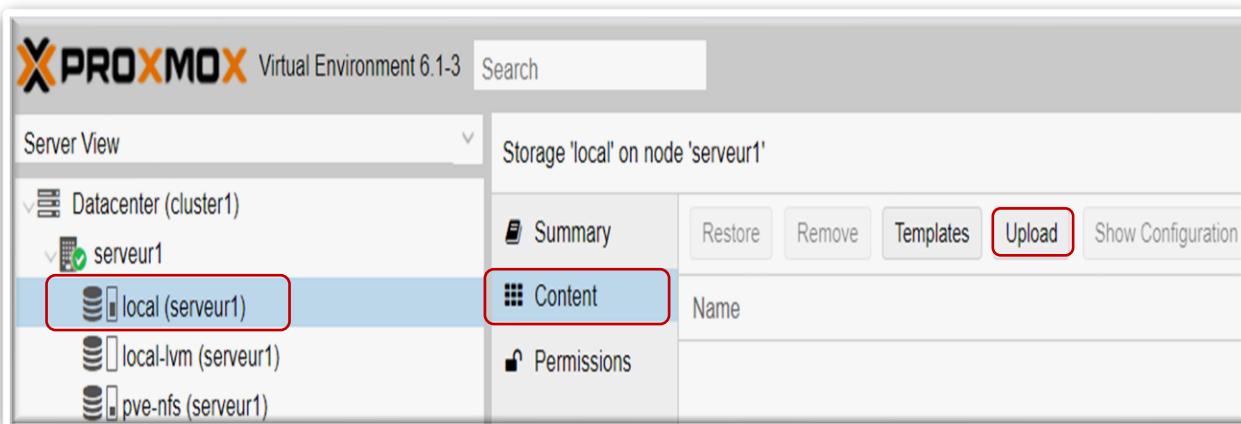
Il suffira de cliquer OK.

L'installation terminée, nous pouvons accéder à l'interface via l'adresse IP du serveur physique. Juste après l'authentification nous arrivons sur la vue qui affiche les informations principales de notre infrastructure, les serveurs physiques et virtuels, les espaces de stockage, les sauvegardes, les restaurations. On y retrouve également les parties : configuration du serveur, authentification, permissions, Le cadre du bas affiche les actions, les résultats des opérations faites sur les serveurs.



1.2 Installation d'une machine Virtuelle :

On va cliquer sur la partition du locale (serveur1) puis content en suite uploadé pour ajouter une image iso et créer un machine virtuelle.



PROXMOX

Une boîte de dialogue vous demandera de sélectionner le fichier (soit fichier ISO, soit fichier de conteneur).



Si Nous allons dans le stockage local-lvm, les boutons **upload** et **templates** seront grisés, car nous ne pouvons pas stocker d'ISO ou templates de conteneurs dans un volume LVM.

Les données sont stockées dans **/var/lib/vz/template/iso** pour les fichiers ISO, et dans **/var/lib/vz** pour les templates de conteneurs ou dans les dossiers équivalents du magasin de données. L'ISO sera alors disponible et visible.

The screenshot shows the 'Storage' interface for 'local' storage on node 'Serveur1'. The 'Content' tab is selected. At the top, there are buttons for 'Restore', 'Remove', 'Templates', 'Upload', and 'Show Configuration'. Below is a list of items under 'Name': 'iso (1 item)' followed by 'ubuntu-12.04.5-desktop-i386.iso', 'vma.lzo (1 item)' followed by 'vzdump-qemu-101-2020_03_19-22_12_42.vma.lzo'. Both file names are highlighted with a red border.

Nous pouvons passer ensuite à la création de notre VM proprement dite en cliquant sur « Créer VM » dans la barre en haut ou via le bouton droit sur le nom de votre serveur à gauche.

PROXMOX

The screenshot shows the PROXMOX Web Interface. On the left, the 'Server View' sidebar lists several nodes: 'Datacenter (cluster)', 'serveur1' (with VMs 103, 104, 105), 'serveur2', and 'serveurNFS'. The 'Virtual Machine 104 (ServeurWeb) on node 'serveur1'' details page is open. The 'Edit' button is highlighted with a red box. The 'Options' tab is selected. The configuration table includes:

| Name | ServeurWeb |
|-----------------------------|---|
| Start at boot | No |
| Start/Shutdown order | order=any |
| OS Type | Linux 5.x - 2.6 Kernel |
| Boot Order | Disk 'scsi0', CD-ROM, Network |
| Use tablet for pointer | Yes |
| Hotplug | Disk, Network, USB |
| ACPI support | Yes |
| KVM hardware virtualization | No |
| Freeze CPU at startup | No |
| Use local time for RTC | No |
| RTC start date | now |
| SMBIOS settings (type1) | uuid=074c7ff2-29cd-4f2b-a8e1-5d3a77958991 |
| QEMU Guest Agent | Default (Disabled) |
| Protection | No |
| Spice Enhancements | none |

Depuis la fenêtre de création de VM, sur le premier écran, Nous pourrons choisir le serveur s'il y en a plusieurs, dans notre cas serveur1. L'id VM par défaut conviendra, libre à nous de le modifier tant qu'il reste unique. Il restera à renseigner le nom de la VM :

The screenshot shows the 'Create: Virtual Machine' dialog box. The 'General' tab is active. The configuration fields are:

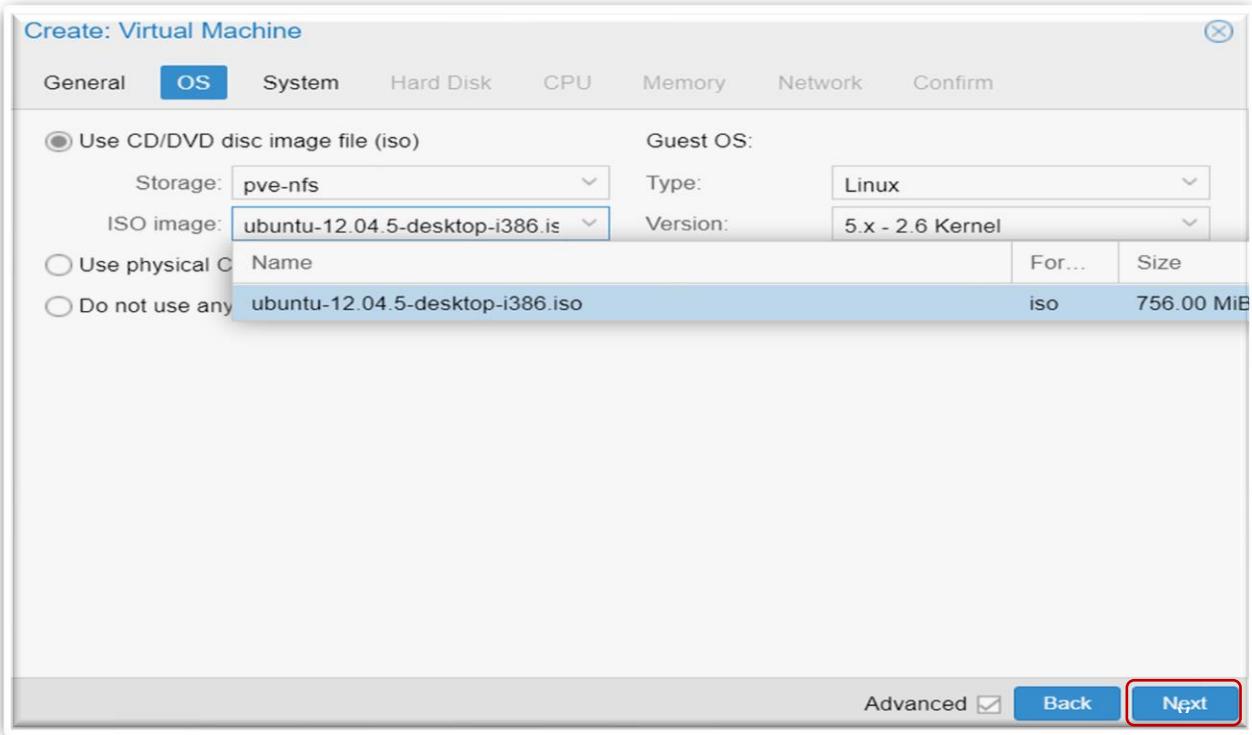
| Field | Value |
|-----------------------|--------------------------|
| Node: | serveur1 |
| VM ID: | 105 |
| Name: | ServeurWeb |
| Start at boot: | <input type="checkbox"/> |
| Start/Shutdown order: | any |
| Startup delay: | default |
| Shutdown timeout: | default |

At the bottom right, the 'Next' button is highlighted with a red box.

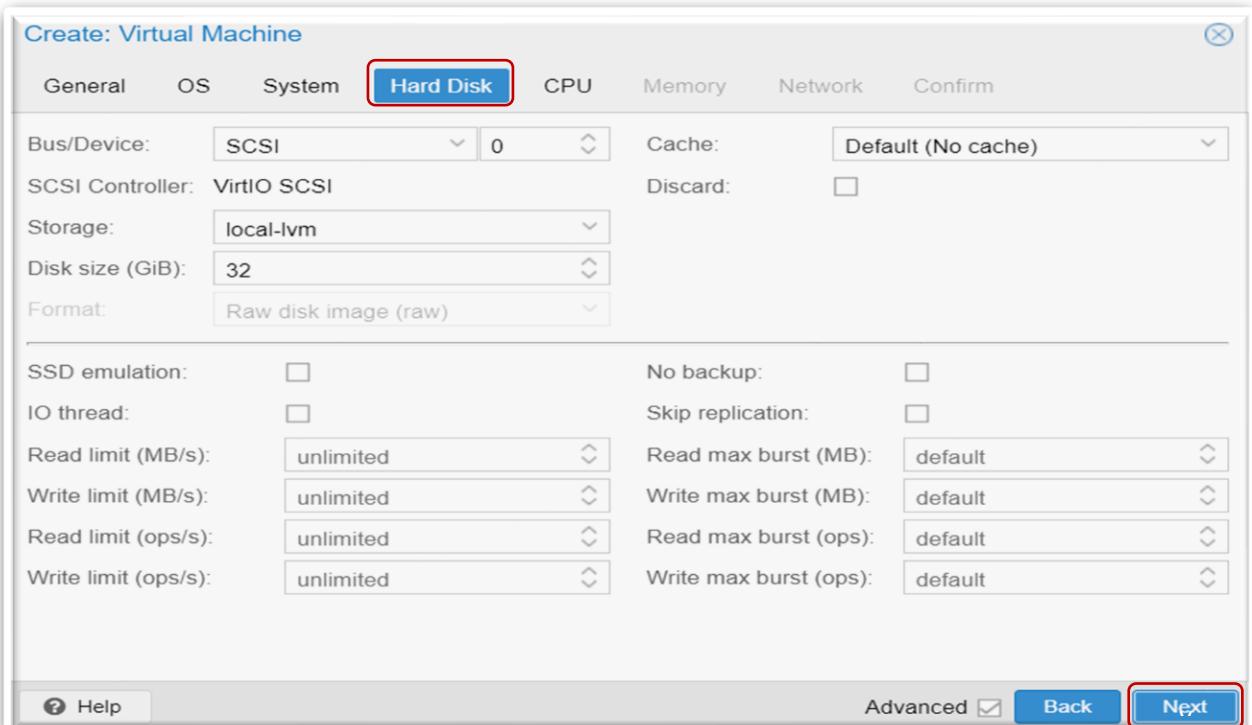
Le second écran va nous demander :

PROXMOX

Une image disque : il faudra sélectionner l'ISO et avant l'espace de stockage (local ou autre).

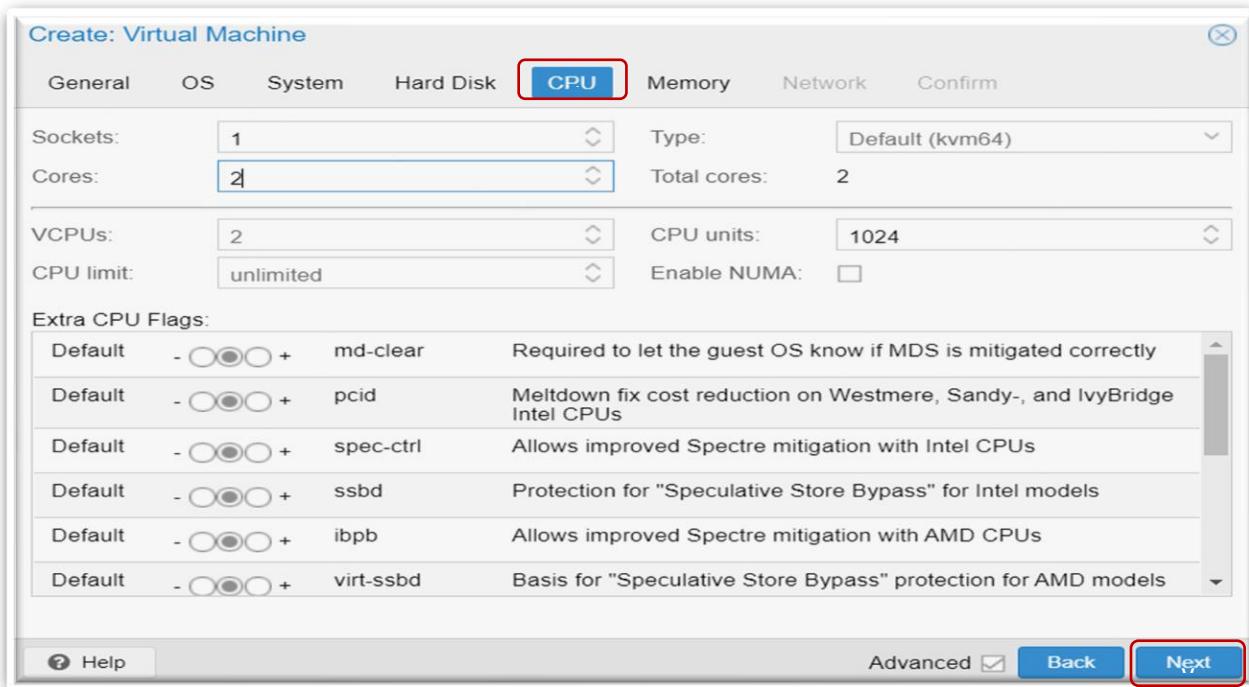


Sur l'écran suivant, nous allons créer un disque virtuel. Par défaut, celui-ci sera en SCSI (nous pourrons sélectionner le numéro de périphérique sur le bus). Nous pourrons sinon utiliser un disque SATA, IDE, ou VirtIO :

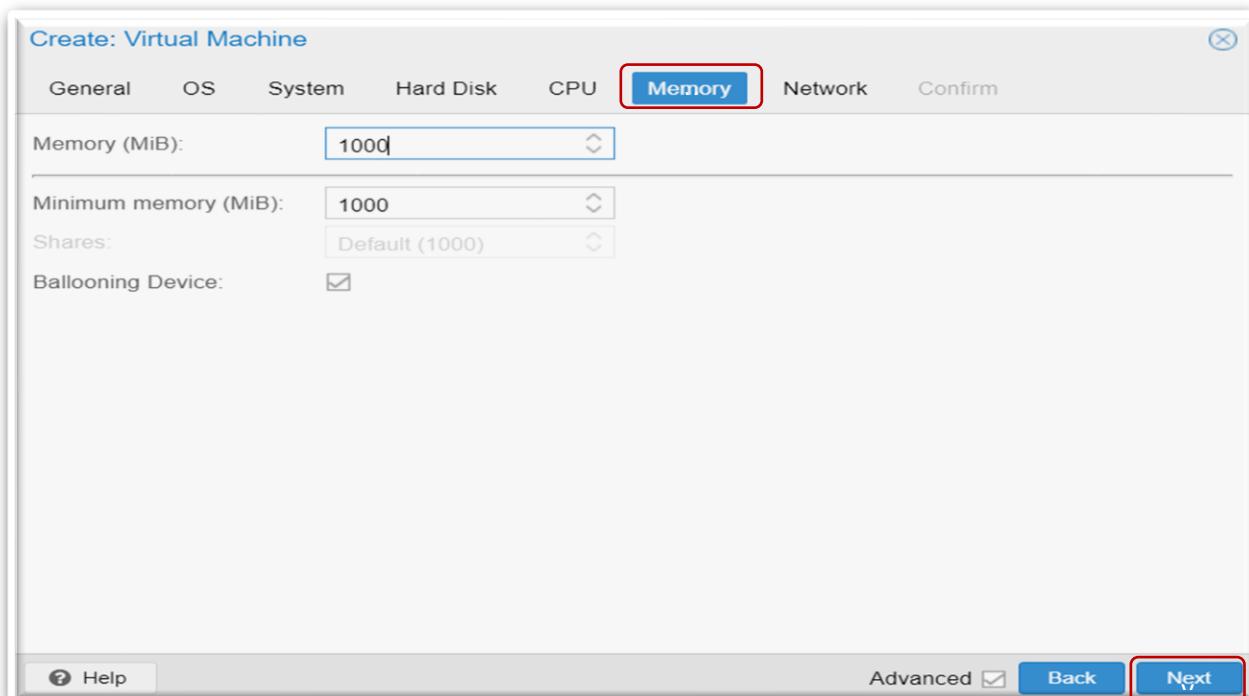


PROXMOX

Nous pourrons ensuite effectuer les réglages CPU (nombre de cœurs, socket, son type).



L'écran suivant permettra de choisir la taille mémoire :



L'écran suivant concerne les réglages réseau :

PROXMOX

Create: Virtual Machine

General OS System Hard Disk CPU Memory Network Confirm

No network device

Bridge: Model:

VLAN Tag: MAC address:

Firewall:

Disconnect: Rate limit (MB/s):

Multiqueue:

Advanced Back

Nous aurons ensuite l'écran récapitulatif de la VM. Nous pourrons alors modifier ceux-ci en retournant sur les différents écrans en cliquant sur les onglets (ou les modifier après création de la VM).

Create: Virtual Machine

General OS System Hard Disk CPU Memory Network Confirm

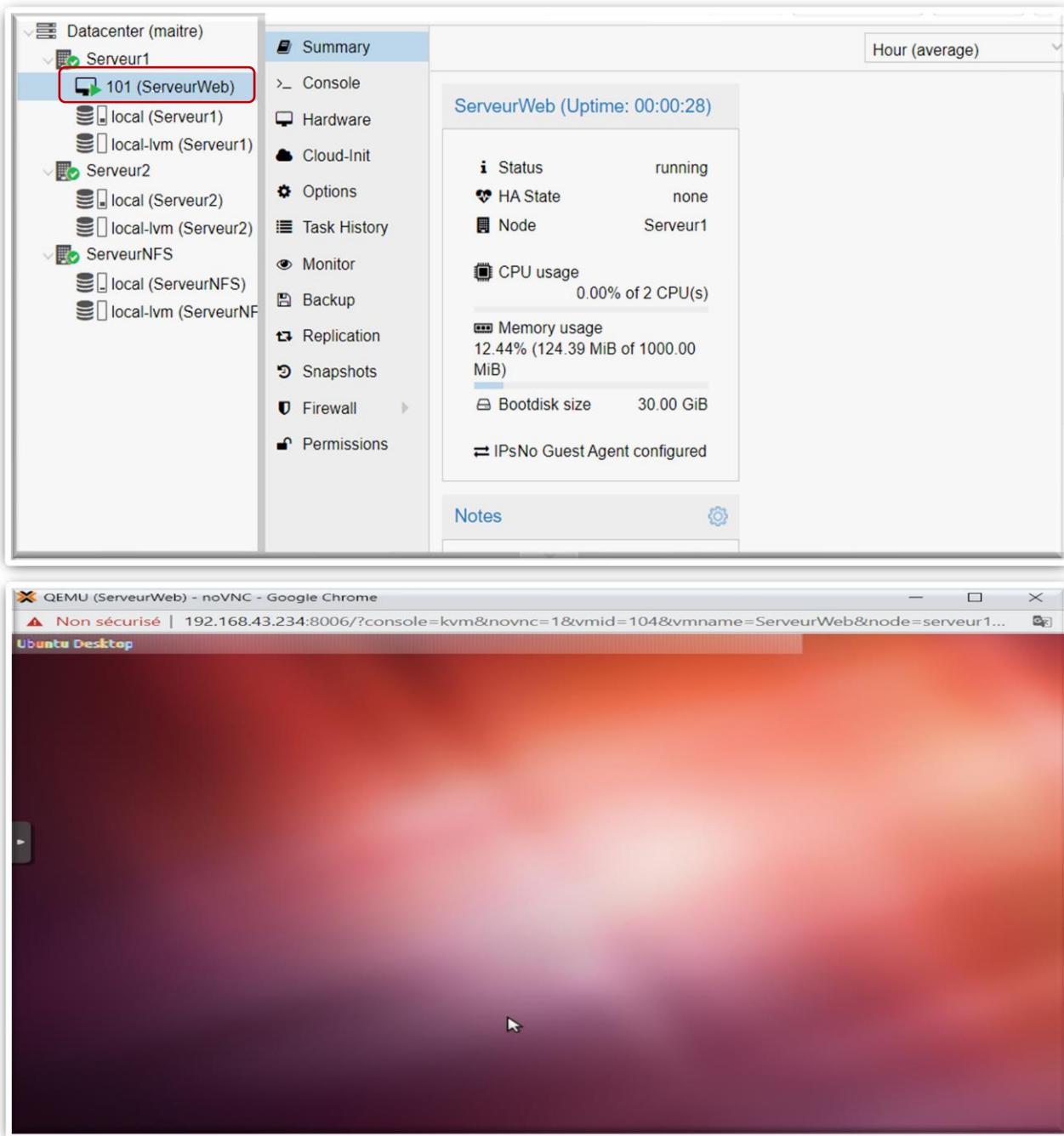
| Key ↑ | Value |
|----------|---|
| cores | 2 |
| ide2 | pve-nfs:iso/ubuntu-12.04.5-desktop-i386.iso,media=cdrom |
| memory | 1000 |
| name | ServeurWeb |
| net0 | virtio,bridge=vmbr0,firewall=1 |
| nodename | serveur1 |
| numa | 0 |
| ostype | I386 |
| scsi0 | local-lvm:32 |
| scsihw | virtio-scsi-pci |
| sockets | 1 |
| vmid | 105 |

Start after created

Advanced Back

Une fois la VM créée, elle apparaîtra sur la partie gauche, et Nous verrons son tableau de bord dans la partie droite en la sélectionnant :

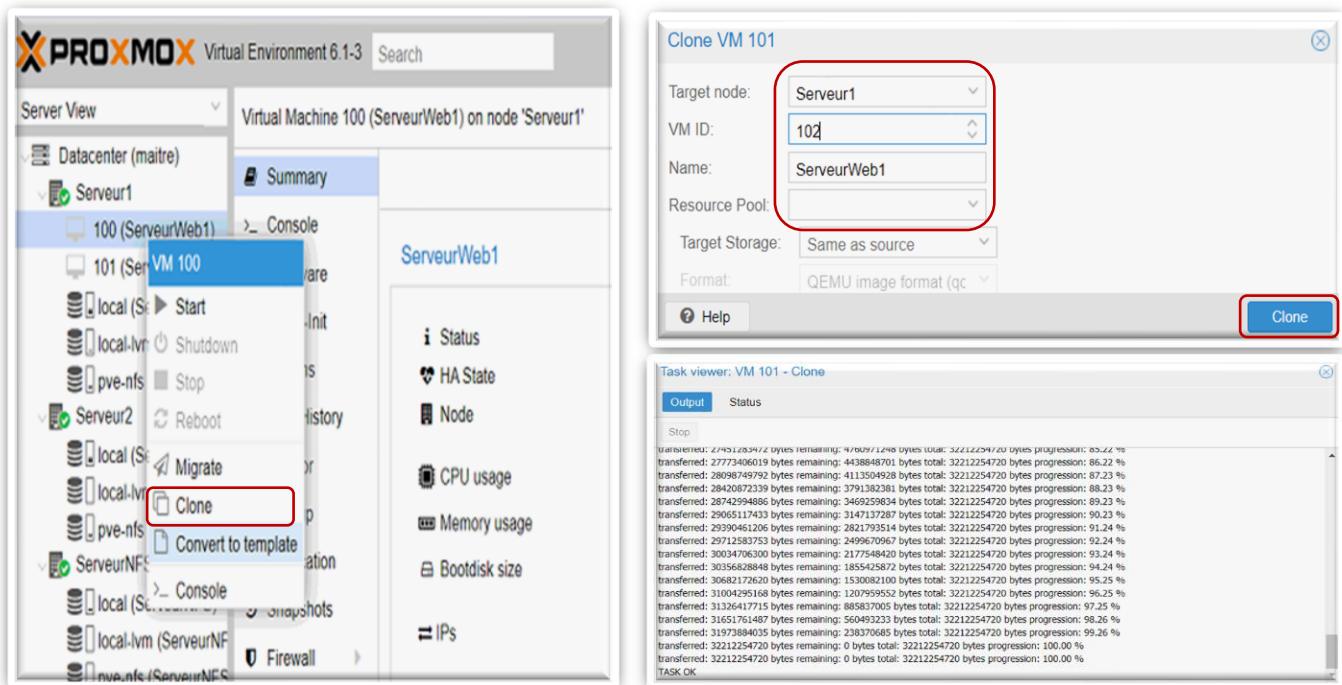
PROXMOX



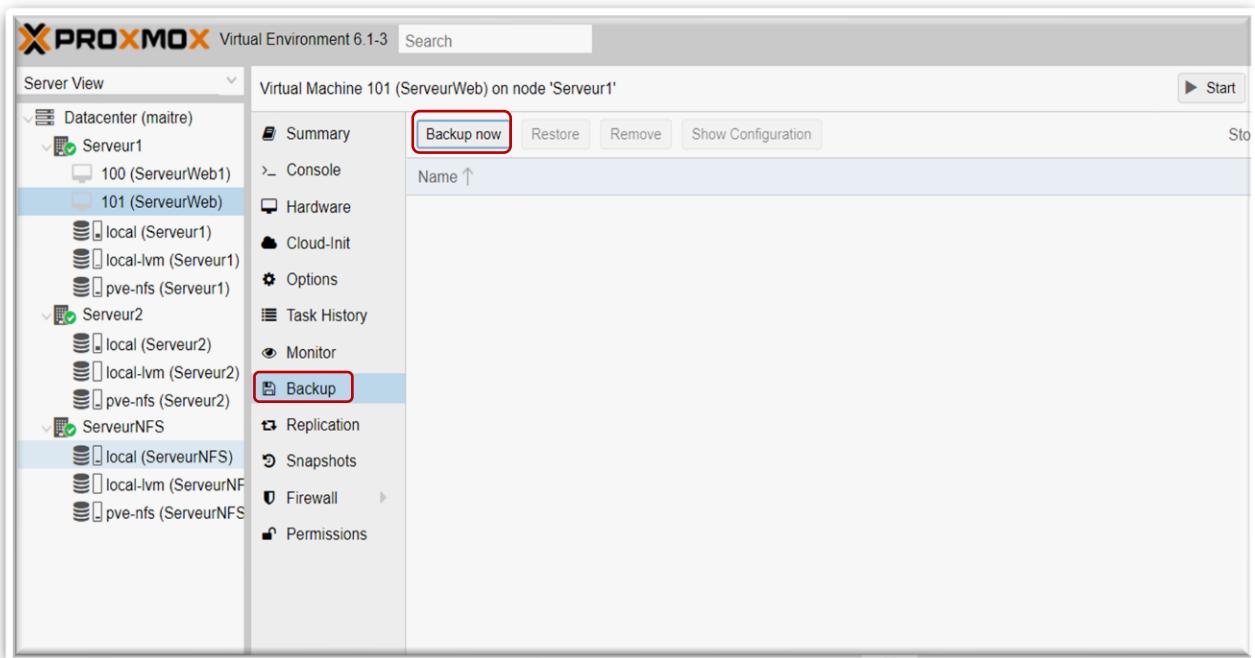
1.3 Clonage de la machine virtuelle :

Cloner une VM permet d'en faire une copie. Cette copie sera alors autonome par rapport à sa source. Celle-ci aura sa propre vie et les mêmes réglages que la VM source, du moins tant que vous ne les changez pas. Pour cloner une VM, il faudra cliquer sur le bouton droit sur le nom de la VM, il nous donne après une petite fenêtre pour entrer un nouveau ID et le nom de la machine :

PROXMOX



1.4 Sauvegarde et restauration de la machine virtuelle :



Nous créerons alors un job de sauvegarde, ou nous sélectionnerons, les nœuds à sauvegarder (en cas de présence de plusieurs serveurs), par défaut tous, l'espace de stockage où mettre les sauvegardes

PROXMOX

Backup VM 101

Storage: local

Mode: Snapshot

Compression: LZO (fast)

Send email to: tor.mohcine@gmail.com

Backup

INFO: status: 79% (25760563200/32212254720), sparse 79% (25760563200), duration 57, read/write 483/0 MB/s
INFO: status: 84% (27118010368/32212254720), sparse 84% (27118010368), duration 60, read/write 452/0 MB/s
INFO: status: 88% (28522840064/32212254720), sparse 88% (28522840064), duration 63, read/write 468/0 MB/s
INFO: status: 93% (30033051648/32212254720), sparse 93% (30033051648), duration 66, read/write 503/0 MB/s
INFO: status: 97% (31478579200/32212254720), sparse 97% (31478579200), duration 69, read/write 481/0 MB/s
INFO: status: 100% (32212254720/32212254720), sparse 100% (32212254720), duration 72, read/write 244/0 MB/s
INFO: transferred 32212 MB in 72 seconds (447 MB/s)
INFO: stopping kvm after backup task
INFO: archive file size: 2MB
INFO: Finished Backup of VM 101 (00:01:18)
INFO: Backup finished at 2020-03-19 22:14:00
INFO: Backup job finished successfully
TASK OK

Virtual Machine 101 (ServeurWeb) on node 'Serveur1'

Name ↑

vzdump-qemu-101-2020_03_19-22_12_42.vma.lzo

Storage: local Format: vma.lzo Size: 2.02 MiB

On doit sauvegarder en premier les machines virtuelles pour qu'on trouve la sauvegarde dans le dossier **/var/lib/vz/dump**.

```
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@Serveur1:~# cd /var/lib/vz/
root@Serveur1:/var/lib/vz# cd dump/
root@Serveur1:/var/lib/vz/dump# ls
vzdump-qemu-101-2020_03_19-22_12_42.log  vzdump-qemu-101-2020_03_19-22_12_42.vma.lzo
root@Serveur1:/var/lib/vz/dump#
```

A l'aide de la commande **scp** On a copié le dossier dump du premier Proxmox qui contient les sauvegardes des machines virtuelles dans le dossier dump du 2eme Proxmox.

```
root@Serveur1:/var/lib/vz/dump# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source          destination
ACCEPT    tcp  --  anywhere        anywhere          state NEW tcp spt:ssh
ACCEPT    tcp  --  anywhere        anywhere          tcp spt:ssh
ACCEPT    tcp  --  anywhere        anywhere          tcp spt:ssh
ACCEPT    tcp  --  anywhere        anywhere          tcp dpt:ssh
DROP      tcp  --  anywhere        anywhere          tcp dpt:ssh
DROP      tcp  --  anywhere        anywhere          tcp spt:ssh

Chain FORWARD (policy ACCEPT)
target     prot opt source          destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source          destination
root@Serveur1:/var/lib/vz/dump# scp -rp /var/lib/vz/dump root@192.168.43.238:/var/lib/vz/dump
vzdump-qemu-101-2020_03_19-22_12_42.vma.lzo                                100% 2073KB   22.0MB/s  00:00
vzdump-qemu-101-2020_03_19-22_12_42.log                                     100% 3867     1.0MB/s  00:00
root@Serveur1:/var/lib/vz/dump# mkdir /var/lib/p1
root@Serveur1:/var/lib/vz/dump# ls
```

PROXMOX

Puis On a restauré comme exemple les machines 100 et 101 avec la commande **qmrestore** « le nom du fichier de sauvegarde » « ID de la machine virtuelle restauré ». Voilà le fichier il est bien copiez dans 2eme Proxmox dans **/dump**.

```
root@Serveur2:/etc/pve/nodes/Serveur2# cd ..
root@Serveur2:/etc/pve/nodes# cd ..
root@Serveur2:/etc/pve# cd ..
root@Serveur2:/etc# cd ..
root@Serveur2:/# ls /var/lib/vz/dump/
dump
root@Serveur2:/# ls /var/lib/
```

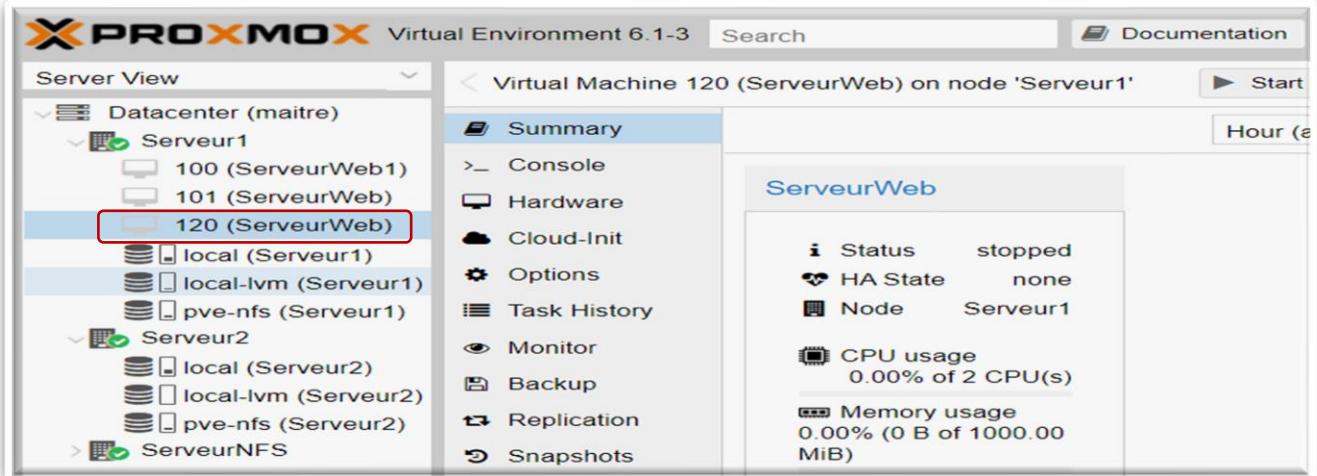


```
progress 99% (read 31890145280 bytes, duration 3 sec)
progress 100% (read 32212254720 bytes, duration 3 sec)
total bytes read 32212254720, sparse bytes 32212254720 (100%)
rescan volumes...
root@Serveur1:/var/lib/vz/dump/dump# qmrestore /var/lib/vz/dump/dump/dump/vzdump-qemu-101-2020_03_19-22_12_42.vma.lzo 120
```

```
progress 67% (read 21582249984 bytes, duration 3 sec)
progress 68% (read 21904359424 bytes, duration 3 sec)
progress 69% (read 22226468864 bytes, duration 3 sec)
progress 70% (read 22548578304 bytes, duration 3 sec)
progress 71% (read 22870753280 bytes, duration 3 sec)
progress 72% (read 23192862720 bytes, duration 3 sec)
progress 73% (read 23514972160 bytes, duration 3 sec)
progress 74% (read 23837081600 bytes, duration 3 sec)
progress 75% (read 24159191040 bytes, duration 3 sec)
progress 76% (read 24481366016 bytes, duration 3 sec)
progress 77% (read 24803475456 bytes, duration 3 sec)
progress 78% (read 25125584896 bytes, duration 3 sec)
progress 79% (read 25447694336 bytes, duration 3 sec)
progress 80% (read 25769803776 bytes, duration 3 sec)
progress 81% (read 26091978752 bytes, duration 3 sec)
progress 82% (read 26414088192 bytes, duration 3 sec)
progress 83% (read 26736197632 bytes, duration 3 sec)
progress 84% (read 27058307072 bytes, duration 3 sec)
progress 85% (read 27380416512 bytes, duration 3 sec)
progress 86% (read 27702591488 bytes, duration 3 sec)
progress 87% (read 28024700928 bytes, duration 3 sec)
progress 88% (read 28346810368 bytes, duration 3 sec)
progress 89% (read 28668919808 bytes, duration 3 sec)
progress 90% (read 28991029248 bytes, duration 3 sec)
progress 91% (read 29313204224 bytes, duration 3 sec)
progress 92% (read 29635313664 bytes, duration 3 sec)
progress 93% (read 29957423104 bytes, duration 3 sec)
progress 94% (read 30279532544 bytes, duration 3 sec)
progress 95% (read 30601641984 bytes, duration 3 sec)
progress 96% (read 30923816960 bytes, duration 3 sec)
progress 97% (read 31245926400 bytes, duration 3 sec)
progress 98% (read 31568035840 bytes, duration 3 sec)
progress 99% (read 31890145280 bytes, duration 3 sec)
progress 100% (read 32212254720 bytes, duration 3 sec)
total bytes read 32212254720, sparse bytes 32212254720 (100%)
rescan volumes...
root@Serveur1:/var/lib/vz/dump/dump/dump#
```

PROXMOX

Voilà la machine il est bien restaurez.



1.5 Crédation d'un conteneur :

Avant de pouvoir créer un conteneur, il nous faut un Template qui servira de modèle. Nous pourrons en récupérer en cliquant « Template » en allant dans le serveur (serveur1 dans notre cas) → pve-nfs → contenu → Template :

| Type ↑ | Package | Version | Description |
|----------------------------|----------------------------------|-----------|--|
| lxc | proxmox-mailgateway-6.0-standard | 6.0-2 | Proxmox Mailgateway 6.0 |
| Section: system (18 Items) | | | |
| lxc | alpine-3.10-default | 20190626 | LXC default image for alpine 3.10 (20190626) |
| lxc | alpine-3.9-default | 20190224 | LXC default image for alpine 3.9 (20190224) |
| lxc | archlinux-base | 201909... | ArchLinux base image. |
| lxc | centos-6-default | 20191016 | LXC default image for centos 6 (20191016) |
| lxc | centos-7-default | 20190926 | LXC default image for centos 7 (20190926) |
| lxc | centos-8-default | 20191016 | LXC default image for centos 8 (20191016) |

Une fois le template uploadé, Nous pourrons le voir et éventuellement le supprimer depuis l'espace de stockage pve-nfs → contenu, dans la section « Template de conteneur » :

| Name |
|---|
| txz (1 Item) |
| centos-7-default_20190926_amd64.tar.xz |
| vma.lzo (1 Item) |
| vzdump-qemu-100-2020_03_24-20_24_14.vma.lzo |

PROXMOX

La création du conteneur se fera par le clic en haut à droite à côté de créer VM. Dans le premier écran, il nous faudra sélectionner : le nœud de destination en cas de plusieurs serveurs et le nom du conteneur et son mot de passe et éventuellement une clé SSH.

The screenshot shows the 'Create: LXC Container' dialog box. The 'General' tab is selected. The 'Node' dropdown is set to 'Serveur2'. The 'CT ID' dropdown is set to '102'. The 'Hostname' input field contains 'CentOS-HA'. A checked checkbox is next to 'Unprivileged container'. On the right, there are fields for 'Resource Pool' (empty), 'Password' (redacted), 'Confirm password' (redacted), and 'SSH public key' (empty). A 'Load SSH Key File' button is below these. At the bottom, there are 'Help', 'Advanced', 'Back', and a redboxed 'Next' button.

Sur le second écran, Nous sélectionnerons le stockage de destination (si nous avons plusieurs stockages), le modèle de conteneur :

The screenshot shows the 'Create: LXC Container' dialog box. The 'Template' tab is selected. The 'Storage' dropdown is set to 'pve-nfs'. The 'Template' dropdown is set to 'centos-7-default_20190926_amc'. Below these, a table lists the selected template: Name 'centos-7-default_20190926_amd64.tar.xz', Format 'txz', and Size '65.47 MiB'.

Nous pourrons ensuite sélectionner la taille du disque :

The screenshot shows the 'Create: LXC Container' dialog box. The 'Root Disk' tab is selected. The 'Storage' dropdown is set to 'pve-nfs'. The 'Disk size (GiB)' dropdown is set to '4'.

PROXMOX

Nous sélectionnerons enfin le nombre de cœurs et la mémoire affectée au conteneur :

Create: LXC Container

General Template Root Disk **CPU** Memory Network DNS Confirm

Cores: 2

L'écran suivant permettra de fixer la mémoire dédiée au conteneur ainsi que son swap :

Create: LXC Container

General Template Root Disk CPU **Memory** Network DNS Confirm

Memory (MiB): 1012

Swap (MiB): 1012

L'écran suivant concerne le réseau :

Create: LXC Container

General Template Root Disk CPU Memory **Network** DNS Confirm

Name: eth0

MAC address: auto

Bridge: vmbr0

VLAN Tag: no VLAN

Rate limit (MB/s): unlimited

Firewall:

IPv4: Static DHCP

IPv4/CIDR:

Gateway (IPv4):

IPv6: Static DHCP SLAAC

IPv6/CIDR:

Gateway (IPv6):

Help Advanced Back **Next**

Et enfin les DNS :

Create: LXC Container

General Template Root Disk CPU Memory Network **DNS** Confirm

DNS domain: use host settings

DNS servers: use host settings

PROXMOX

Nous aurons ensuite l'écran de résumé, avec une case permettant le démarrage du conteneur juste après sa création :

Create: LXC Container

| General | Template | Root Disk | CPU | Memory | Network | DNS | Confirm |
|--------------|---|-----------|-----|--------|---------|-----|---------|
| Key ↑ | Value | | | | | | |
| cores | 2 | | | | | | |
| hostname | CentOS-HA | | | | | | |
| memory | 1012 | | | | | | |
| net0 | bridge=vmbr0,name=eth0,ip=dhcp,firewall=1 | | | | | | |
| nodename | Serveur2 | | | | | | |
| ostemplate | pve-nfs:vztmpl/centos-7-default_20190926_amd64.tar.xz | | | | | | |
| pool | | | | | | | |
| roots | pve-nfs:4 | | | | | | |
| swap | 1012 | | | | | | |
| unprivileged | 1 | | | | | | |
| vmid | 102 | | | | | | |

Start after created

Advanced Back Finish

Le conteneur en création :

```
extracting archive '/mnt/pve/pve-nfs/template/cache/centos-7-default_20190926_amd64.tar.xz'  
Total bytes read: 422809600 (404MiB, 12MiB/s)  
Detected container architecture: amd64  
Creating SSH host key 'ssh_host_ecdsa_key' - this may take some time ...  
done: SHA256:WXlqNSE2K5vj8wuQJIDZupNB7aK08Bc8Nze5p0woaA8 root@CentOS-HA  
Creating SSH host key 'ssh_host_ed25519_key' - this may take some time ...  
done: SHA256:OLisa9c617LBRDPToYf4eYrYpR29WoLDOVOOrX6dm/1U root@CentOS-HA  
Creating SSH host key 'ssh_host_rsa_key' - this may take some time ...  
done: SHA256:E2bxQiEMpNLK1fcRB8pIEW7+Yxto6FuMHy5oSuV0gv0 root@CentOS-HA  
Creating SSH host key 'ssh_host_dsa_key' - this may take some time ...  
done: SHA256:hakoHDuzSd0cMwytJTCVhdH+xxxkTLEs0qtkZ5b+nHM root@CentOS-HA  
TASK OK
```

X Serveur1 - Proxmox Console - Google Chrome

⚠ Non sécurisé | 192.168.43.237:8006/?console=lxc&xtermjs=1&vmid=102&vmname=CentOS-HA

```
Centos Linux 7 (Core)
Kernel 5.3.10-1-pve on an x86_64

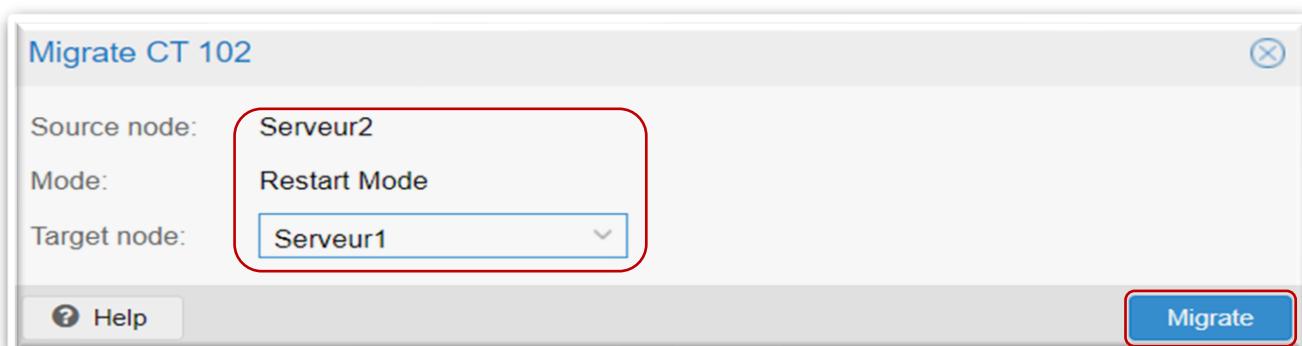
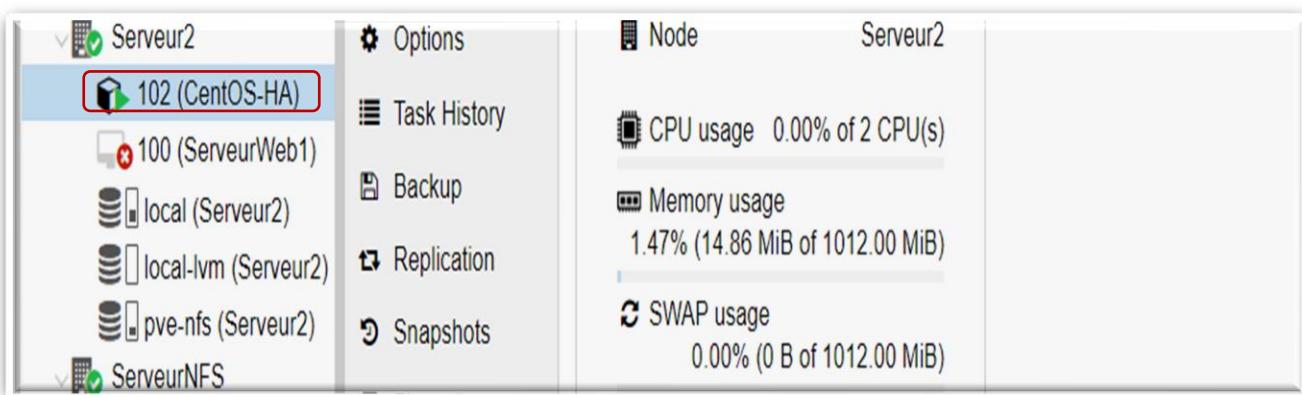
CentOS-HA login: █
```

PROXMOX

1.6 Migration d'un conteneur :

Migrer une Conteneur consistera à cloner celle-ci vers un autre nœud (un autre hyperviseur du Datacenter) en activant celui-ci et supprimant la Conteneur du nœud source. Ceci peut se faire à chaud c'est-à-dire VM en activité. Il faudra cliquer sur le bouton droit sur le nom de la Conteneur, il nous donne après une petite fenêtre :

Voilà avant la migration de conteneur il est démarré dans le serveur 2 :



Donc voilà le conteneur il est bien Migrate a 1^{er} Serveur 1 :

PROXMOX

The screenshot shows the Proxmox Web Interface. On the left, the 'Server View' sidebar lists 'Datacenter (maître)' and 'Serveur1'. Under 'Serveur1', several virtual machines are listed: '102 (CentOS-HA)' (highlighted with a red box), '101 (ServeurWeb)', '120 (ServeurWeb)', 'local (Serveur1)', 'local-lvm (Serveur1)', and 'pve-nfs (Serveur1)'. The main content area is titled 'Node 'Serveur2''. It includes a search bar and a table with columns: Type ↑, Description, Disk usage..., and Memory u. The table contains the following data:

| Type ↑ | Description | Disk usage... | Memory u |
|---------|----------------------|---------------|----------|
| qemu | 100 (ServeurWeb1) | | |
| storage | local (Serveur2) | 38.4 % | |
| storage | local-lvm (Serveur2) | 0.0 % | |
| storage | pve-nfs (Serveur2) | 28.2 % | |

Chapitre 2 : la haute disponibilité :

2.1 Mise en place d'un cluster entre 3 Serveur :

Le “cluster” Proxmox permet notamment de faire de la migration de container d’une machine à une autre “à chaud”, de gérer l’ensemble de ces containers à partir de l’un ou l’autre des hyperviseurs, Pour la Création du cluster sur le premier serveur il faut choisir une des nœuds pour être le master.

```
root@Serveur1:~# pvecm create maître
Corosync Cluster Engine Authentication key generator.
Gathering 2048 bits for key from /dev/urandom.
Writing corosync key to /etc/corosync/authkey.
Writing corosync config to /etc/pve/corosync.conf
Restart corosync and cluster filesystem
root@Serveur1:~# _
```

On peut consulter son bon fonctionnement avec la commande [pvecm status](#), qui devrait nous renvoyer les informations suivantes.

PROXMOX

```
root@Serveur1:~# pvecluster status
Cluster information
-----
Name:          maitre
Config Version: 1
Transport:     knet
Secure auth:   on

Quorum information
-----
Date:           Mon Mar 16 18:23:32 2020
Quorum provider: corosync_votequorum
Nodes:          1
Node ID:        0x0000000001
Ring ID:        1.4
Quorate:       Yes

Votequorum information
-----
Expected votes: 1
Highest expected: 1
Total votes:    1
Quorum:         1
Flags:          Quorate

Membership information
-----
  Nodeid      Votes Name
0x0000000001      1 192.168.43.237 (local)
root@Serveur1:~#
```

Avec la commande **pvecluster add @ip** de serveur maître on a ajouté les nœuds de serveur secondaire et serveur NFS sur le serveur maître.

```
root@Serveur2:~# pvecluster add 192.168.43.237
Please enter superuser (root) password for '192.168.43.237': *****
Establishing API connection with host '192.168.43.237'
The authenticity of host '192.168.43.237' can't be established.
X509 SHA256 key fingerprint is D7:DF:88:E3:4E:5D:15:51:6F:BE:B8:45:65:D8:A0:65:86:1E:DD:71:EB:2A:9D:
E8:1C:C1:CC:9B:9C:0A:59:A0.
Are you sure you want to continue connecting (yes/no)? yes
Login succeeded.
Request addition of this node
Join request OK, finishing setup locally
stopping pve-cluster service
backup old database to '/var/lib/pve-cluster/backup/config-1584379729.sql.gz'
waiting for quorum...OK
(re)generate node files
generate new node certificate
merge authorized SSH keys and known hosts
generated new node certificate, restart pveproxy and pvedaemon services
successfully added node 'Serveur2' to cluster.
root@Serveur2:~#
```

PROXMOX

Et voilà le résultat on peut accéder au 2 serveur à l'aide d'une seule interface.

The screenshot shows the Proxmox Virtual Environment 6.1-3 interface. On the left, the 'Server View' sidebar is open, showing 'Datacenter' selected. Under 'Datacenter', there are two entries: 'Serveur1' and 'Serveur2'. Each host entry has two sub-items: 'local (Serveur1)' and 'local-lvm (Serveur1)' for Serveur1, and 'local (Serveur2)' and 'local-lvm (Serveur2)' for Serveur2. On the right, the main panel is titled 'Datacenter' and contains a 'Cluster Information' section. It shows 'Cluster Name: maitre', 'config Version: 3', and 'Number of Nodes: 2'. Below this is a 'Cluster Nodes' section with two entries: 'Node 1: Serveur1' and 'Node 2: Serveur2'. There are also 'Search', 'Summary', 'Cluster', 'Ceph', and 'Options' buttons in the sidebar.

2.2 Mise en place d'un cluster HA :

La mise en place d'un cluster haute disponibilité d'hyperviseurs Proxmox avec un serveur NFS. Contrairement à la version précédente de proxmox, il n'est plus possible de créer un cluster HA sur 2 noeuds depuis la version 4.x de Proxmox donc on ajouté le 3 éme noeud (ServeurNFS) avec la même commande.

Pour le stockage partagé, Nous avons choisi la technologie NFS pour sa facilité de mis en œuvre et sa maintenance il faut en premier installer le paquet de NFS Sur le 3 eme noeud ServeurNFS on a rendu un serveur de partage de fichier.

The screenshot shows the Proxmox Virtual Environment 6.1-3 interface. On the left, the 'Server View' sidebar is open, showing 'Datacenter' selected. Under 'Datacenter', there are three entries: 'Serveur1', 'Serveur2', and 'ServeurNFS'. Each host entry has two sub-items: 'local (Serveur1)', 'local-lvm (Serveur1)', 'local (Serveur2)', 'local-lvm (Serveur2)', 'local (ServeurNFS)', and 'local-lvm (ServeurNFS)'. A red oval highlights the 'Datacenter' section of the sidebar where all three hosts are listed. On the right, the main panel is titled 'Datacenter' and contains a 'Cluster Information' section. It shows 'Cluster Name: maitre', 'config Version: 3', and 'Number of Nodes: 3'. Below this is a 'Cluster Nodes' section with three entries: 'Node 1: Serveur1', 'Node 2: Serveur2', and 'Node 3: ServeurNFS'. There are also 'Search', 'Summary', 'Cluster', 'Ceph', 'Storage', 'Backup', 'Replication', 'Permissions', 'Users', 'Groups', 'Pools', and 'Roles' buttons in the sidebar.

PROXMOX

Puis on a créé un dossier **/pve** pour le partage de fichier et on a modifié dans le fichier de configuration **/etc/export** par la commande nano **/etc/exports**.

```
root@Serveur1:~#  
root@Serveur1:~# mkdir /var/lib/vz/pve_
```

```
root@ServeurNFS:~# nano /etc/exports _
```

```
# /etc/exports: the access control list for filesystems which may be exported  
#           to NFS clients. See exports(5).  
  
# Example for NFSv2 and NFSv3:  
# /srv/homes      hostname1(rw,sync,no_subtree_check) hostname2(ro,sync,no_subtree_check)  
  
# Example for NFSv4:  
# /srv/nfs4      gss/krb5i(rw,sync,fsid=0,crossmnt,no_subtree_check)  
# /srv/nfs4/homes  gss/krb5i(rw,sync,no_subtree_check)  
  
#/var/lib/vz/pve 192.168.43.237(rw,no_subtree_check,sync,no_root_squash)  
/var/lib/vz/pve 192.168.43.238(rw,no_subtree_check,sync,no_root_squash)  
/var/lib/vz/pve 192.168.43.239(rw,no_subtree_check,sync,no_root_squash)
```

Puis on exporter la configuration NFS avec la commande **exportfs -r** et Vérification avec la commande **exportfs -v**.

```
root@ServeurNFS:~# exportfs -r  
root@ServeurNFS:~# exportfs -v  
/var/lib/vz/pve  
    192.168.43.237(rw,wdelay,no_root_squash,no_subtree_check,sec=sys,rw,secure,no_root_s  
quash,no_all_squash)  
/var/lib/vz/pve  
    192.168.43.238(rw,wdelay,no_root_squash,no_subtree_check,sec=sys,rw,secure,no_root_s  
quash,no_all_squash)  
/var/lib/vz/pve  
    192.168.43.239(rw,wdelay,no_root_squash,no_subtree_check,sec=sys,rw,secure,no_root_s  
quash,no_all_squash)  
root@ServeurNFS:~#
```

Puis on va ajouter une partition NFS sur les pve en cluster, On doit Cliquer sur « Datacenter », puis sur « Storage » et « Add », on va sélectionner NFS.

PROXMOX

The screenshot shows the Proxmox VE 6.1-3 interface. In the left sidebar, under 'Datacenter (maître)', there are three servers: 'Serveur1', 'Serveur2', and 'ServeurNFS'. The 'Storage' tab is selected in the center panel. A red box highlights the 'Add' button and the 'NFS' option in the list. Below the list, a table shows existing storage configurations: 'VZDump backup...' (Path/Tar... /var/lib/vz, S... No, E... Yes, Bandwid...), 'Disk image, Co...' (Path/Tar... /var/lib/vz, S... No, E... Yes, Bandwid...). At the bottom, a table lists cluster log tasks.

| Start Time | End Time | Node | User name | Description | Status |
|-----------------|-----------------|------------|-----------|----------------|--------|
| Mar 16 18:41:05 | Mar 16 18:41:07 | Serveur1 | root@pam | VM 101 - Start | OK |
| Mar 16 18:40:23 | Mar 16 18:40:24 | Serveur1 | root@pam | VM 101 - Cr... | OK |
| Mar 16 18:37:53 | Mar 16 18:38:06 | Serveur1 | root@pam | Copy data | OK |
| Mar 16 18:30:31 | Mar 16 18:30:58 | ServeurNFS | root@pam | Join Cluster | OK |
| Mar 16 18:28:39 | Mar 16 18:29:04 | Serveur2 | root@pam | Join Cluster | OK |

Ensuite il faut nommer la partition (pve-nfs) puis ajouté @IP de serveur nfs puis Export (indiquer le partage NFS).

The dialog box for adding a new NFS storage pool. The 'ID' field is 'pve-nfs'. The 'Server' field is '192.168.43.239'. The 'Export' field is set to '/var/lib/vz/pve'. The 'Content' field is set to 'Disk image, ISO image,'. The 'Nodes' dropdown is set to 'All (No restrictions)'. The 'Enable' checkbox is checked. The 'Max Backups' dropdown is set to 1. At the bottom right, there are 'Advanced' and 'Add' buttons.

Alors le stockage NFS est ajouté automatiquement à tous les nœuds.

The screenshot shows the Proxmox VE 6.1-3 interface. The 'Storage' tab is selected in the center panel. The 'Datacenter' table shows three nodes: 'Serveur1', 'Serveur2', and 'ServeurNFS'. Each node has a new 'pve-nfs' entry in its storage list, highlighted with a red box. The 'pve-nfs' entry in the 'Serveur1' list has 'Content' set to 'Disk image, ISO image,' and 'Path/Tar...' set to '/mnt/pve/'. The other two entries have 'Content' set to 'VZDump backup...' and 'Path/Tar...' set to '/var/lib/vz'.

PROXMOX

Depuis la version 4.x de proxmox, la mise place d'un cluster HA a été très simplifiée et peut être entièrement configurée depuis l'interface web d'administration. Création d'un groupe comprenant les nœuds proxmox :

Il faut se rendre dans l'onglet « Datacenter » puis dans la section « HA » cliquer sur « Groups » puis cliquer sur « Create ».

The screenshot shows the Proxmox Web Interface with the title bar "PROXMOX Virtual Environment 6.1-3". The left sidebar is titled "Server View" and lists "Datacenter (maître)", "Serveur1", "Serveur2", and "ServeurNFS". The main panel is titled "Datacenter" and contains a sidebar with "Options", "Storage", "Backup", "Replication", "Permissions" (with sub-options "Users", "Groups", "Pools", "Roles", "Authentication"), and "HA" (which is expanded to show "Groups" and "Fencing"). A red box highlights the "Create" button and the "Groups" tab under the HA category.

Donner un nom au groupe et sélectionner les hyperviseurs à intégrer. Il est aussi possible de définir une priorité dans la colonne « priority »:

The screenshot shows the "Create: HA Group" dialog. It has fields for "ID" (set to "HA"), "restricted" (unchecked), and "nofailback" (unchecked). There is a "Comment:" text input field. Below is a table with columns "Node ↑", "Memory usage %", "CPU usage", and "Priority". Three nodes are listed: "Serveur1" (53.5%, 7.0% of 2CPUs, priority ▲), "Serveur2" (50.8%, 7.2% of 2CPUs, priority ▲), and "ServeurNFS" (70.2%, 7.3% of 2CPUs, priority ▲). A red box highlights the table area.

Puis voilà le groupe HA il est ajouté.

The screenshot shows the "Datacenter" section with the "Groups" table. The table has columns "Group ↑", "restricted", "nofailback", and "Nodes". One row is present: "HA" (No, No, Nodes: "Serveur1,ServeurNFS,Serveur2"). A red box highlights this row.

PROXMOX

Dans cette fenêtre, il suffit de sélectionner le groupe précédemment créé :

Add: Resource: Container/Virtual Machine

| | | | |
|----------------------|--|---------------------|---------|
| VM: | 102 | Group: | HA |
| Max. Restart: | 1 | Request State: | started |
| Max. Relocate: | 1 | | |
| Comment: | Nous ajoutons cette VM au groupe de HA | | |
| Help | | Add | |

On peut voir que la machine virtuelle est dans le menu HA.

Resources

| ID | State | Node | Name | Max. Restart | Max. Reloc... | Group |
|--------|---------|----------|-----------|--------------|---------------|-------|
| ct:102 | stopped | Serveur1 | CentOS-HA | 1 | 1 | HA |
| vm:100 | error | Serveur2 | | 1 | 1 | HA |

Pour rappel, ce sont les nœuds Serveur1 et Serveur2 qui doit gérer la disponibilité des VM et ServeurNFS uniquement l'hébergement et seulement la gestion des VM en cas d'indisponibilité des deux autres nœuds ou bien tomber en panne.

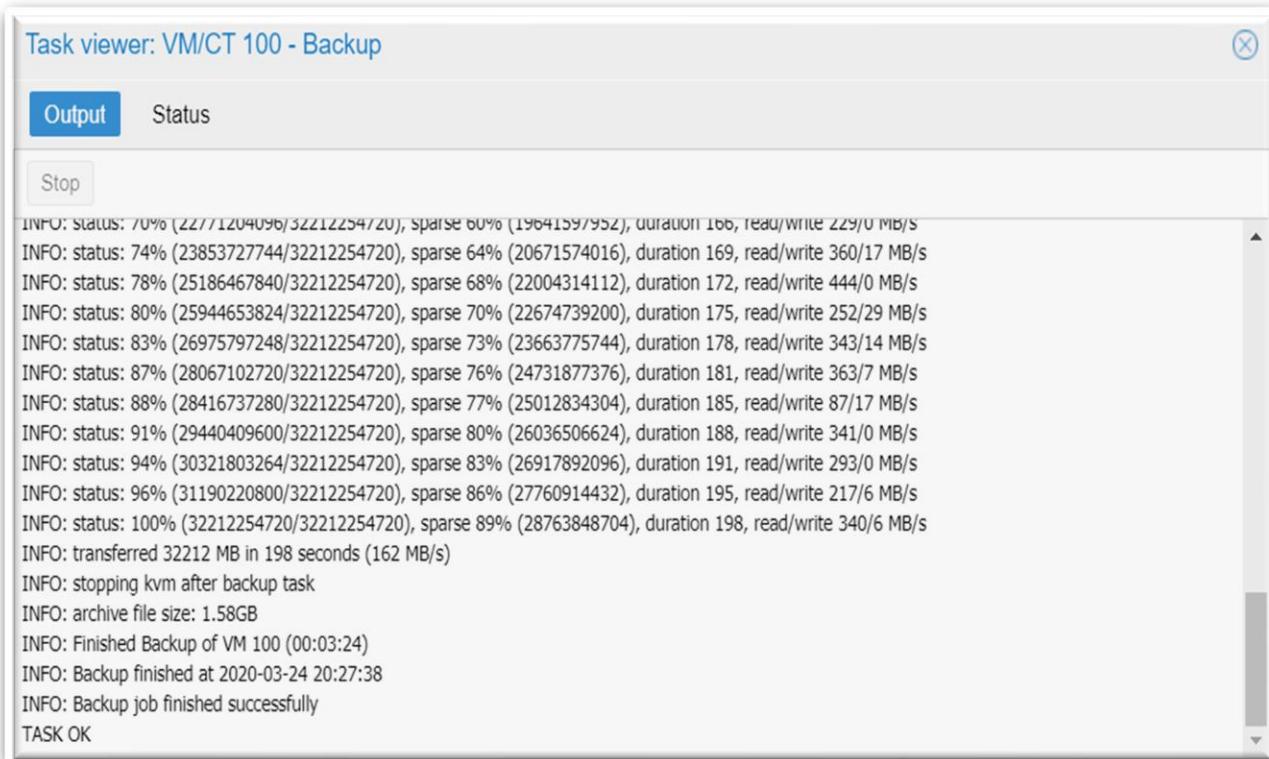
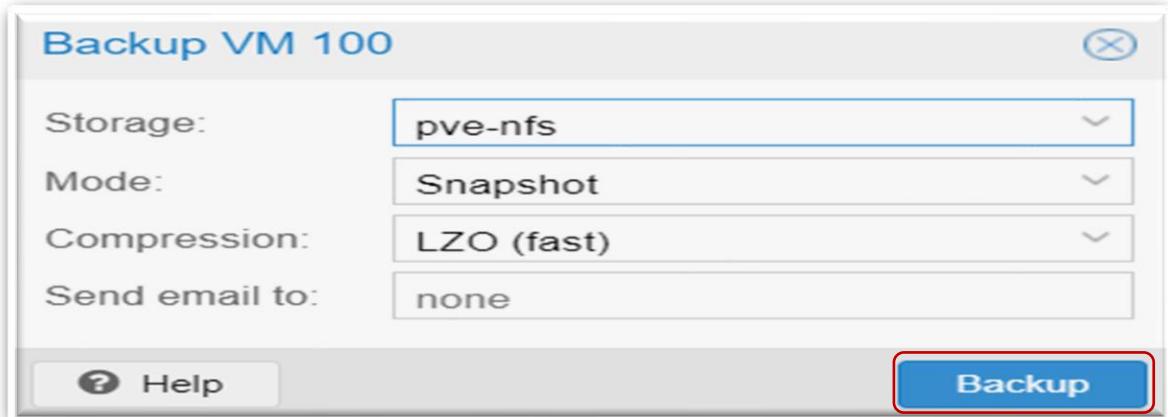
Edit: HA Group

| ID: | HA | restricted: | <input type="checkbox"/> |
|------------|----------------|---------------|--------------------------|
| | | nofailback: | <input type="checkbox"/> |
| Comment: | | | |
| Node ↑ | Memory usage % | CPU usage | Priority |
| Serveur1 | 54.5 % | 4.9% of 2CPUs | 1 |
| Serveur2 | 50.9 % | 4.5% of 2CPUs | 1 |
| ServeurNFS | 70.5 % | 4.0% of 2CPUs | 0 |

[Help](#) [OK](#) [Reset](#)

PROXMOX

Puis quand on a fini on a sauvegardé la machine virtuelle pour tester la fonctionnalité de la haute disponibilité.



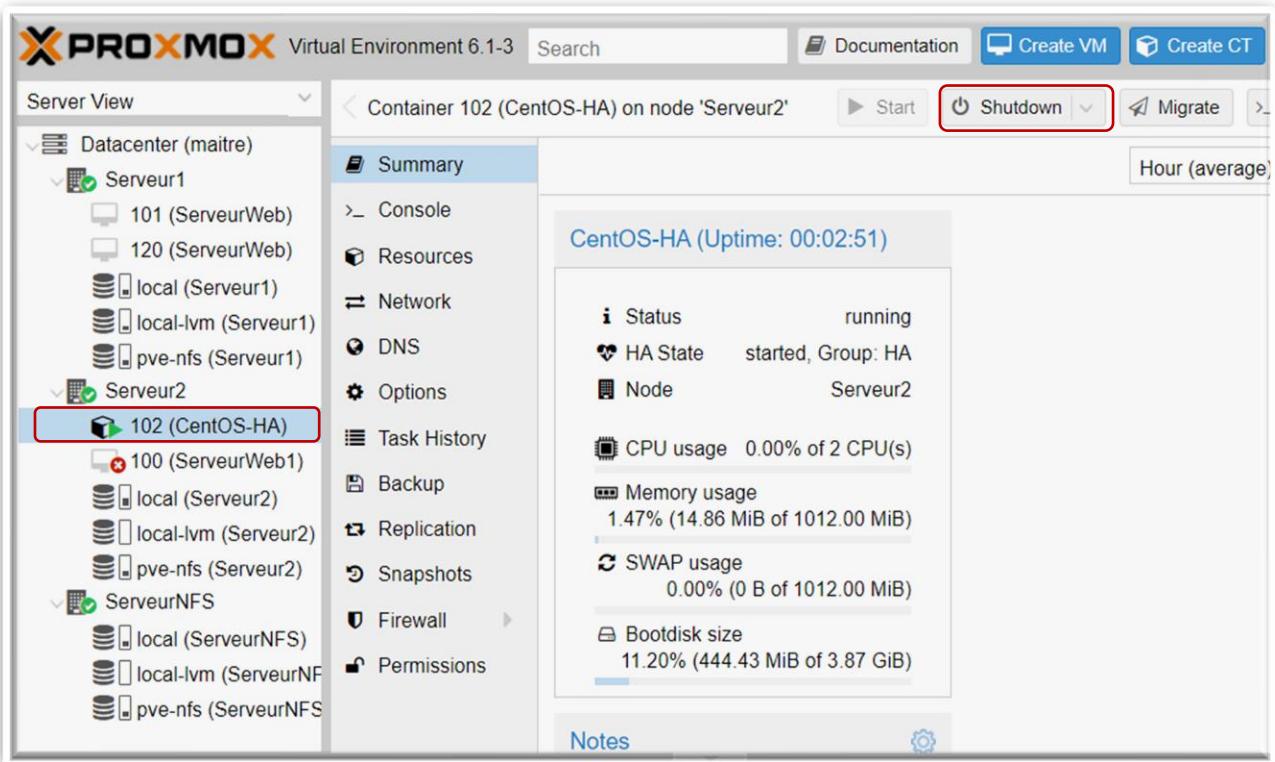
Lors de la désactivation de la carte réseau de l'hyperviseur 2 la machine S-A-D on a mettre le serveur2 en panne on a juste arrêtée le service de cluster « 102(CentOS-HA) » est migrée automatiquement sur l'autre hyperviseur.

On peut faire avec la ligne de commande ou bien graphiquement.

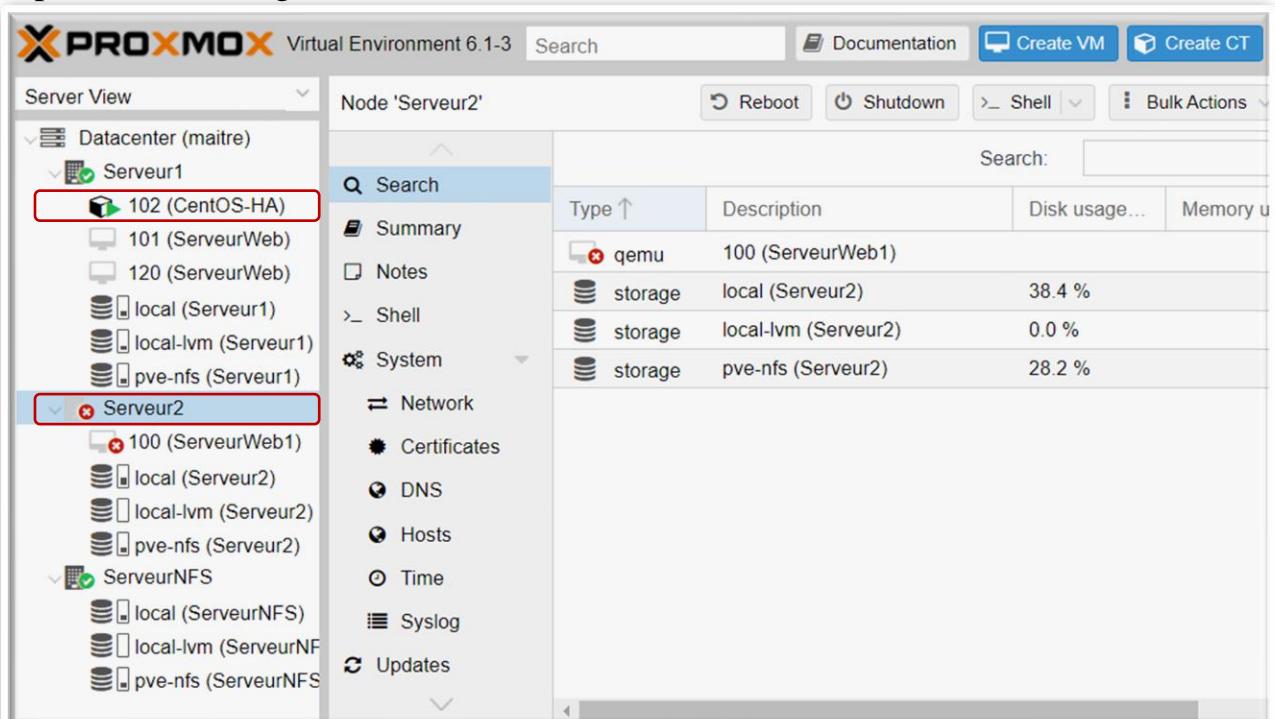
La commande **/etc/init.d/pve-cluster stop**

PROXMOX

Graphiquement voilà note machine ou bien conteneur il est dans le deuxième serveur2 on va cliquez sur Shutdown.

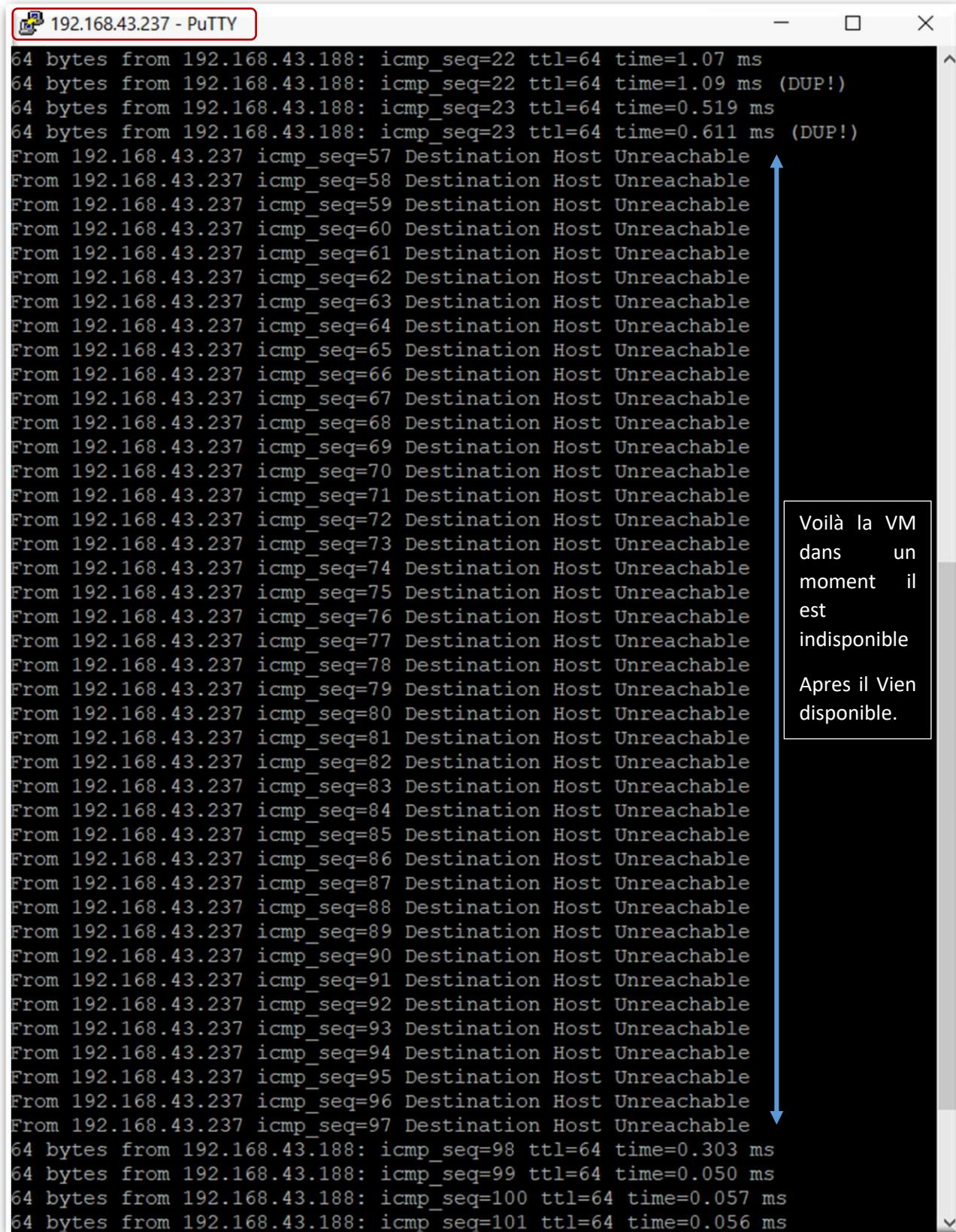


Puis voilà on a constaté que la machine ou bien conteneur fait une migration automatique sur le 1^{er} serveur1. On a lancé un Ping de serveur2 vers le conteneur pour avoir la migration.



PROXMOX

Voilà le Ping :



```
64 bytes from 192.168.43.188: icmp_seq=22 ttl=64 time=1.07 ms
64 bytes from 192.168.43.188: icmp_seq=22 ttl=64 time=1.09 ms (DUP!)
64 bytes from 192.168.43.188: icmp_seq=23 ttl=64 time=0.519 ms
64 bytes from 192.168.43.188: icmp_seq=23 ttl=64 time=0.611 ms (DUP!)
From 192.168.43.237 icmp_seq=57 Destination Host Unreachable
From 192.168.43.237 icmp_seq=58 Destination Host Unreachable
From 192.168.43.237 icmp_seq=59 Destination Host Unreachable
From 192.168.43.237 icmp_seq=60 Destination Host Unreachable
From 192.168.43.237 icmp_seq=61 Destination Host Unreachable
From 192.168.43.237 icmp_seq=62 Destination Host Unreachable
From 192.168.43.237 icmp_seq=63 Destination Host Unreachable
From 192.168.43.237 icmp_seq=64 Destination Host Unreachable
From 192.168.43.237 icmp_seq=65 Destination Host Unreachable
From 192.168.43.237 icmp_seq=66 Destination Host Unreachable
From 192.168.43.237 icmp_seq=67 Destination Host Unreachable
From 192.168.43.237 icmp_seq=68 Destination Host Unreachable
From 192.168.43.237 icmp_seq=69 Destination Host Unreachable
From 192.168.43.237 icmp_seq=70 Destination Host Unreachable
From 192.168.43.237 icmp_seq=71 Destination Host Unreachable
From 192.168.43.237 icmp_seq=72 Destination Host Unreachable
From 192.168.43.237 icmp_seq=73 Destination Host Unreachable
From 192.168.43.237 icmp_seq=74 Destination Host Unreachable
From 192.168.43.237 icmp_seq=75 Destination Host Unreachable
From 192.168.43.237 icmp_seq=76 Destination Host Unreachable
From 192.168.43.237 icmp_seq=77 Destination Host Unreachable
From 192.168.43.237 icmp_seq=78 Destination Host Unreachable
From 192.168.43.237 icmp_seq=79 Destination Host Unreachable
From 192.168.43.237 icmp_seq=80 Destination Host Unreachable
From 192.168.43.237 icmp_seq=81 Destination Host Unreachable
From 192.168.43.237 icmp_seq=82 Destination Host Unreachable
From 192.168.43.237 icmp_seq=83 Destination Host Unreachable
From 192.168.43.237 icmp_seq=84 Destination Host Unreachable
From 192.168.43.237 icmp_seq=85 Destination Host Unreachable
From 192.168.43.237 icmp_seq=86 Destination Host Unreachable
From 192.168.43.237 icmp_seq=87 Destination Host Unreachable
From 192.168.43.237 icmp_seq=88 Destination Host Unreachable
From 192.168.43.237 icmp_seq=89 Destination Host Unreachable
From 192.168.43.237 icmp_seq=90 Destination Host Unreachable
From 192.168.43.237 icmp_seq=91 Destination Host Unreachable
From 192.168.43.237 icmp_seq=92 Destination Host Unreachable
From 192.168.43.237 icmp_seq=93 Destination Host Unreachable
From 192.168.43.237 icmp_seq=94 Destination Host Unreachable
From 192.168.43.237 icmp_seq=95 Destination Host Unreachable
From 192.168.43.237 icmp_seq=96 Destination Host Unreachable
From 192.168.43.237 icmp_seq=97 Destination Host Unreachable
64 bytes from 192.168.43.188: icmp_seq=98 ttl=64 time=0.303 ms
64 bytes from 192.168.43.188: icmp_seq=99 ttl=64 time=0.050 ms
64 bytes from 192.168.43.188: icmp_seq=100 ttl=64 time=0.057 ms
64 bytes from 192.168.43.188: icmp_seq=101 ttl=64 time=0.056 ms
```

Voilà la VM
dans un
moment il
est
indisponible

Apres il Vien
disponible.

CONCLUSION GENERALE

Les travaux présentés dans ce rapport s'inscrivent dans le cadre de notre projet de fin d'étude. L'objectif de notre travail était de trouver une solution pour permettre la haute disponibilité des services. Les techniques de virtualisation répondent à ce problème, car elles permettent de migrer un service d'un ordinateur à un autre en cas de dysfonctionnement. Notre rôle a été tout d'abord de nous informer sur les caractéristiques de la virtualisation pour en déduire l'état de l'art actuel. Puis nous avons identifié les deux éléments essentiels pour la mise en place de la solution :

1. Un environnement de virtualisation pour créer et migrer les machines virtuelles.

2. Un stockage partagé pour permettre la migration des machines virtuelles sans temps d'arrêt.

En conclusion, notre projet de fin d'étude nous a permis de mettre en œuvre des compétences scolaires, professionnelles et humaines pour un sujet intéressant. On a de plus acquis de nouvelles compétences dans le domaine du système informatique.

PROXMOX

Bibliographie

<https://memo-linux.com/proxmox-4-x-mise-en-place-dun-cluster-ha/>

<https://alexbacher.fr/unixlinux/proxmox-ha/>

<https://chrtophe.developpez.com/tutoriels/proxmox/#L15-2>

<https://blog.elao.com/fr/infra/creer-un-cluster-2-nodes-proxmox/>