

## Sommaire :

Sommaire : .....	1
Introduction.....	1
Serveur physique : .....	2
La virtualisation : .....	3
Cours de Monsieur Lebatteur : .....	5
Pour commencer sur ProxMox v4 : .....	12
Créer une machine virtuelle sur un ProxMox v4 : .....	13
Procédure pour une machine virtuelle sur ProxMox v4 : .....	13
Sauvegarde, migration et réplication de machines virtuelles avec Proxmox VE : .....	20

## Introduction

Objectif : L'objectif de ce TP est de prendre en main l'outil ProxMox afin de virtualiser des machines, de les cloner, les sauvegarder et les migrer au sein d'un cluster.

Pré-requis : Il faut des connaissances en virtualisation.

Norme : Toutes les commandes issues d'une machine avec un système d'exploitation Debian ou Windows sont écrites ***en gras et en italique***.

## Serveur physique :

### **1. Caractéristiques :**

Un serveur est une sorte de super ordinateur qui est chargé de prodiguer des services à des clients, tel qu'un service de messagerie, de partage d'imprimantes, de stockage de données, etc... Il gère l'accès aux ressources et aux périphériques et les connexions des différents utilisateurs.

Physiquement, il en existe plusieurs types, avec des caractéristiques différentes, mais généralement, il a les mêmes composants de bases qu'un ordinateur classique, mais avec de plus grandes capacités, et souvent multipliés en nombres (redondances de disques durs, de cartes mémoires, de cartes réseaux, d'alimentations, etc...).

Un serveur est allumé 24h/24, 7j/7, et doit être accessible en permanence, d'où la redondance des composants. Dans les grandes entreprises, les serveurs sont empilés dans des baies, et sont regroupés en grappe, pour permettre la répartition de la charge.

On administre un serveur via un système d'exploitation spécifique, comme les Windows Server par exemple.

### **2. Les types de serveurs :**



Serveur tour : Pour les petites entreprises, un serveur peut se trouver sous la forme d'une tour, comme un ordinateur classique.

Serveur rack : Pour les entreprises plus grandes, lorsque de baies sont utilisés, on utilise des serveurs racks, qui sont aplatit et peuvent s'insérer dans ces baies.



Serveur en lames : Les serveurs lames ne peuvent pas fonctionner seuls. Ils ont besoin d'un châssis qui contient

toute l'alimentation électrique et les connexions réseaux. Ils coûtent moins cher que les serveurs racks et peuvent être changé à chaud.

## La virtualisation :

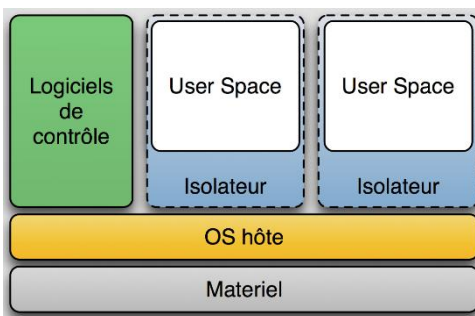
### 1. A quoi ça sert ?

La virtualisation sert à virtualiser un ou plusieurs systèmes d'exploitations, simultanément ou non, comme s'il s'agissait d'un simple logiciel, alors qu'un autre OS est installé sur votre ordinateur physique.

### 2. Comment ça fonctionne ?

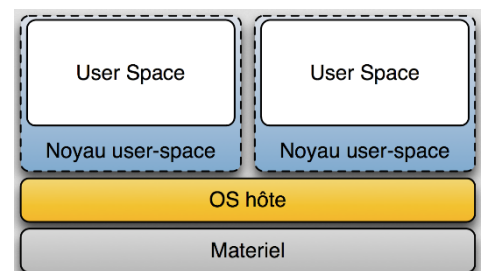
Il existe plusieurs types de virtualisation, mais globalement il s'agit d'un logiciel qui va récupérer les ressources matérielles non utilisées pour virtualiser un système d'exploitation.

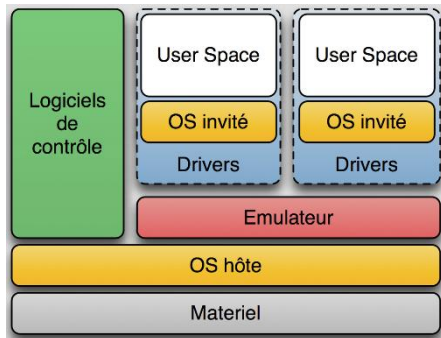
Il existe 4 types de virtualisation différentes :



Isolateur : Un isolateur permet d'isoler une zone d'exécution et de permettre à un système d'exploitation d'être exécuté en dehors des tâches de l'OS hôte.

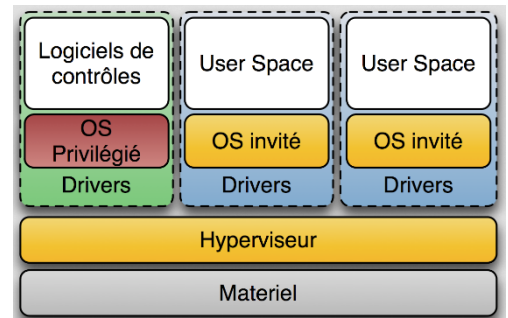
Noyau en espace utilisateur : Surtout utilisé pour le développement de noyaux. Cette méthode est peu performante pour une utilisation classique.





Hyperviseur de type 2 : Un logiciel est installé directement sur notre système d'exploitation. Ce dernier va émuler des systèmes d'exploitation, et donc émuler également le matériel nécessaire pour faire fonctionner ces OS. Cela le rend beaucoup plus dur à utiliser.

Hyperviseur de type 1 : C'est l'hyperviseur utilisé par exemple par Virtual Box. Un hyperviseur fait fonctionner notre système d'exploitation, et utilise les ressources non utilisées pour pouvoir en faire fonctionner d'autres virtuellement.



### 3. Les intérêts :

- On peut ainsi économiser l'achat de matériel physique, et utiliser à 100% les ressources de notre matériel actuel.
- On peut profiter de toutes les fonctionnalités des différents OS.

### 4. Les leaders du marché :

Les sociétés VMware et Microsoft semblent dominer le marché, bien que la plupart des logiciels de virtualisations soient libres et gratuits.

( Cf. <https://fr.wikipedia.org/wiki/Virtualisation> )

## Cours de Monsieur Lebatteur :

### **La virtualisation**

#### A) Caractéristiques d'un serveur :

##### 1. Introduction :

Un réseau met en lien différentes choses. En informatique, c'est une utilisation de plusieurs ordinateurs centraux, qui peuvent fournir des services.

Les objectifs d'un serveur sont :

- il ne doit pas s'arrêter.
- il doit être fiable en fonctionnement, le plus proche possible de 100%.
- il doit pouvoir tenir en charge et répondre à toutes les sollicitations des clients. Cela tient de la configuration matérielle du serveur.

Un serveur réseau est un ordinateur spécifique partageant ses ressources avec d'autres ordinateurs appelés clients.

##### 2. Caractéristiques :

Un serveur possède de la mémoire ECC (Error checking and correcting). Tout ce qu'on exécute est monté en mémoire vive pour y avoir accès. La mémoire vive se vide normalement à l'extinction du PC. Sur un serveur, la mémoire est capable de vérifier le code binaire et de le reconstruire. Un bloc mémoire, une fois utilisée, est libéré.

Un serveur possède des disques durs rapides. Toujours en disques mécaniques, mais à 15 000 ou 20 000 tours/minutes. Ils peuvent être pluggés à chaud.

Un serveur possède des cartes contrôleurs RAID, souvent intégrées à la carte mère.

On doit sauvegarder les données d'un serveur. La sauvegarde des données applicatives est primordiale.

L'alimentation doit être redondante, pour éviter une panne d'électricité.

Sur un bon serveur, tous les composants doivent être doublés : deux cartes réseaux (pour prévenir une panne, et pour répartir les charges), plusieurs contrôleurs de disques.

Pour l'alimentation, un onduleur (batterie) ou un groupe électrogène (moteur diesel) doivent pouvoir alimenter le serveur en cas de panne électrique.

TCO (Total Cost of Ownership)

##### 3. Sécurisation physique du serveur :

Un serveur doit se trouver dans une salle climatisée (20°C) et filtrée (pour avoir le moins de poussières possibles).

Un système anti-incendie.

Les accès à la salle serveur doivent être protégée : par un digicode, une badgeuse, une authentification biométrique, une clé, une caméra avec un logiciel de détection.

Les armoires qui contiennent les serveurs doivent être fermées à clé.

Les boutons Power et Reset doivent être bloqués, pour empêcher les erreurs de manipulations.

Les câbles doivent être attachés, que ce soit les câbles réseaux ou d'alimentations.

Une détection d'ouverture de capot peut être mise en place.

#### 4. Les types de serveur :

Les capacités ne changent pas forcément, mais juste le format. Il existe donc des serveurs tour, serveurs rack (plusieurs formats 1U, 2U...), serveurs lame (qui permettent de créer des clusters) et les armoires rack réunissent des serveurs rack et un écran pour les administrer.

#### 5. Les garanties :

Les extensions de garanties sont plus chères et obligatoires, pour une durée de 3 ans. La garantie de temps de rétablissement (GTR) est le temps nécessaire pour le prestataire de réparer le problème, elle peut être en J+1 ou H+4 par exemple.

## **B) La virtualisation :**

### 1. Introduction :

Qu'est-ce que la virtualisation ?

En moyenne, une machine est utilisée entre 12% et 20% de sa capacité totale.

La virtualisation est l'ensemble des techniques matérielles et/ou logicielles qui permettent de faire fonctionner sur une seule machine physique plusieurs systèmes d'exploitation et/ou plusieurs applications, séparément les uns des autres, comme s'ils fonctionnaient sur des machines physiques distinctes.

La virtualisation permet de réduire les coûts, et donne une image de green-IT (informatique écologique).

- Années 60 : travaux de centre de recherche IBM de Grenoble.
- Années 90 : émulation sur x86 des premiers ordinateurs personnels tels qu'Atari, Amiga, Amstrad...
- Fin des années 90 : introduction de VMware par la société du même nom, virtualisation logicielle des architectures des x86 pour machines x86.
- Suivront (entre autres) dans le monde x86 : QEMU, Bochs, Xen, Linux-VServeur (libres), Virtual PC (MS), VirtualBox, ProxMox VE...

Un émulateur virtualise tout le logiciel et tout le matériel sur nos machines actuelles, et est en principe très lent. Alors que la virtualisation n'émule pas le matériel, mais seulement la partie logicielle.

## 2. Les différents types de virtualisation :

### **Machine virtuelle (ou superviseur) :**

C'est un logiciel qui s'exécute sous le contrôle d'un système hôte (host). Il permet de lancer un ou plusieurs OS invités (guest).

Le rôle du système hôte est de procurer l'accès générique aux ressources physiques (disques, périphériques, connexions).

L'exécution des instructions est native (à quelques exceptions, voir ci-après), les OS hôtes et invités doivent donc être de même architecture matérielle (processeur en particulier), sauf... s'il s'agit d'un « émulateur », l'architecture de l'OS invités ne dépend que de la nature du processeur émulé.

Le superviseur, via l'hôte, doit être capable d'utiliser le matériel physique et ses pilotes pour ses invités.

Les machines virtuelles (OS invités) peuvent être de types différents.

Les performances sont bonnes à excellentes dans le cas d'une machine virtuelle en mode natif.

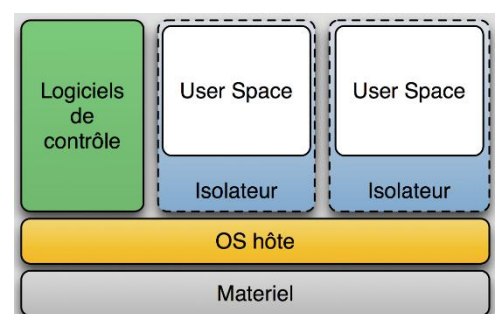
Elles sont mauvaises à déplorables dans le cas d'un émulateur.

Le premier cas est très adapté à des solutions en production, alors que le second est peu utilisable dans ces conditions.

**Attention :** souvent ces deux techniques (exécution native et émulation) sont confondues, pour des performances en totale opposition.

### **Isolation (Operating system level virtualization) :**

Il ne faut jamais qu'un guest empêche un autre guest de fonctionner en cas de problème. Il permet d'isoler l'exécution d'un programme du reste du système d'exploitation :



Le programme est enfermé dans un espace clos pour garder le système sain en cas de problème ou défaillance,

L'isolateur permet de faire tourner plusieurs fois la même application dans un mode multi-instance,

Cette solution est performante. Ce n'est pas une vraie solution de virtualisation de SE. Elle est liée au système Linux.

### **Para-virtualisation (hyperviseur – HV) :**

C'est une extension (une épuration) de l'architecture précédente (machine virtuelle).

L'hyperviseur est un noyau hôte restreint (allégé) et optimisé pour assurer l'exécution exclusive d'OS invités.

Les performances sont normalement excellentes (quasi natives), il s'agit de la technique de virtualisation logicielle la plus efficace.

Les machines virtuelles (OS invités) peuvent être de type différents (mais architecture identique).

Un environnement de virtualisation doit être d'une fiabilité parfaite : un hyperviseur ne doit jamais crasher. S'il tombe en panne, tous les invités sont perdus.

Exemple : Proxmox VE, Hyper-V avec Windows 2008 et 2012 (installation en mode core), Xen et VMware ESX.

### **Compromis flexibilité/performance :**

Le meilleur compromis est la para-virtualisation.

### **Du côté matériel :**

Le processeur prend en charge le support de la virtualisation, et permet d'éviter de faire effectuer à l'hyperviseur (ou à la machine virtuelle) des tâches « micrologicielles » qui ne peuvent être déléguées à un OS hôte dans le cadre d'un processeur non spécialisé.

Aujourd'hui, il y a principalement deux technologies : Intel VT et AMD-V.

Le premier intérêt est d'améliorer les performances, pour les rapprocher encore plus de performances natives.

Selon les logiciels de virtualisation, il y a plusieurs techniques de procuration d'un service de connexion réseau :

- Pas de réseau : le système invité n'a pas d'accès réseau.
- Réseau à travers le système hôte : le système invité possède un (ou plusieurs) interface réseau virtualisé, qui permet de communiquer uniquement avec le système hôte.



- L'accès au réseau externe (réel) peut éventuellement être procuré par le système hôte qui agit en routeur (avec ou sans NAT – translation d'adresse : notre box fait de la translation d'adresse car nous avons qu'une seule adresse IP publique, mais plusieurs IP privées – c'est-à-dire que plusieurs adresses privées sont traduites en une adresse publique – il existe le NAT statique, dynamique et NAT overload ou PAT (Port Address Translation)).
- Le réseau virtuel créé peut éventuellement permettre la communication entre plusieurs machines invitées.

Réseau externe « natif » : le système hôte n'agit que comme un pont Ethernet, le système invité « voit » directement le réseau physique externe.

### **Outils complémentaires :**

Les produits les plus aboutis contiennent également des utilitaires d'administration, très utiles en environnement de production, ainsi que des fonctionnalités/outils tels que « snapshots », export/import de machines, environnement d'exécution seul...

Selon les besoins, sur une machine multiprocesseurs, regarder si le système invité est bridé en environnement monoprocesseur, ou s'il peut lui-même s'exécuter en environnement multiprocesseurs.

VIRTIO est une abréviation pour Virtual Input-Output :

- Virtio est une interface de programmation du noyau Linux utile aux machines virtuelles.
- Canaux de communications particuliers vers le matériel d'un système hôte : accès mémoire, disque, horloge temps réel, réseau.
- Plus performant.
- Nécessite des drivers spécifiques sur la VM.

### **Format des images disques :**

raw : simple et interopérable.

cow : Copy On Write format, une image peut être partagée par plusieurs VM.

qcow, qcow2 : QEMU format, compression, chiffrement, snapshot ( <--- les plus universels).

VMDK : VMWare.

VDI : Virtual Box.

### 3. Intérêts de la virtualisation :

On ne peut pas augmenter indéfiniment le nombre de machines parce que :

Empreinte écologique planétaire :

- Problèmes de déchets électroniques.
- Problèmes d'utilisation de l'énergie : directe + refroidissement.

Et localement :

- Salles machines saturées.
- Problèmes de nuisance sonore, résistance au sol, mise en place de climatisation, puissance électrique.

Contraintes économiques :

- Coûts d'achats / recyclages.
- Coûts de fonctionnements.
- Coûts de maintenance.

Les machines du marché sont de plus en plus puissantes :

- Processeurs en général sous utilisés : charge moyenne d'un processeur comprise entre 9% et 12%, peu d'IO sur les services, banalisation des processeurs quadri-core et des unités multiprocesseurs.

La virtualisation apporte :

- Partage du matériel d'une machine par plusieurs systèmes d'exploitation indépendants (isolés, encapsulés) :
  - o Répondre aux contraintes de sécurité (séparation des services, isolation).
  - o Répondre aux « nouveaux » besoins de maintenance (administration, déploiement, amélioration de la qualité).
  - o Répondre aux besoins de tests, expérimentations, délégation.
- Tout en stabilisant la taille du parc informatique et les coûts associés :
  - o Diminuer le TCO (Total Cost Ownership).
  - o Diminuer la consommation électrique et la nécessité de refroidissement.

**Fonctions avancées des HV :**

**Le redimensionnement dynamique** des VM à chaud.

**La mobilité de machine virtuelle** (vMotion chez VMware, LiveMotion chez Xen, Live Migration chez Microsoft) autorise à la déplacer d'un serveur physique à un autre au travers du réseau sans l'arrêter. La virtualisation du stockage procure des fonctions de création et d'administration de disques virtuels.

**La virtualisation du réseau**, aussi bien celle des interfaces que des équipements eux-mêmes (commutateurs virtuels, pare-feu virtuels).

**La haute-disponibilité (HA)** assure le déplacement et le redémarrage automatisé d'un serveur physique source vers un serveur physique cible d'une VM qui vient de mourir pour n'importe quelle raison.

**L'équilibrage de charge** consiste à orchestrer le placement des VM dans un cluster pour optimiser le fonctionnement de l'ensemble.

**Le miroir de VM**, est une fonction récente, mise en place à des fins de résilience/tolérance de pannes. Deux VM fonctionnent de façon étroitement synchronisée sur deux serveurs physiques ou

au sein du même serveur, de telle façon que le crash de l'une n'a aucune conséquence sur la continuité du service (Remus pour Xen et Fault Tolerance chez VMware).

#### 4. Les différents acteurs du marché :

**Xen** : support de Windows impossible.

**Citrix XenServer** : a été bâti sur Xen et supporté le Windows. Administré via XenCenter.

**VMware ESXi et VMware vCenter Server** : le premier du marché. Son inconvénient est son prix.

**Microsoft Hyper-V** : utilisé depuis Windows Serveur 2003. Basé sur la technologie Xen et ne fonctionne que sur une plateforme matérielle 64 bits.

**Proxmox VE** : construit autour d'une Debian. Fonctionne sur l'architecture 64 bits. Elle supporte le clustering et la haute disponibilité (HA). Elle utilise le stockage DAS (Direct Attached Storage). Migration des VM à chaud.

#### 5. Bénéfices de la virtualisation :

Utilisation accrue des ressources disponibles.

Disponibilité accrue :

- Haute disponibilité.
- Simplification de la mise en œuvre des PRA (Plan de reprise d'activité – ce qui va se passer quand un matériel qui est tombé se remet en route).

Flexibilité accrue :

- Rapidité de mise à disposition. La création, le clonage, la migration, tout cela est très rapide à mettre en place.

Optimisation des coûts d'exploitation :

- Outils d'administration intégrés.
- Indépendance des plates-formes matérielles.

#### 6. Les enjeux électriques :

1,25 millions de ces équipements se trouvent en service en 2010 en France :

- une grosse partie de ces serveurs se trouve dans des PME ou sur des sites secondaires, seulement un petit tiers se trouverait dans un Datacenter, là où l'effet de la virtualisation-consolidation se fait le plus sentir sur la consommation électrique.

Un calcul approximatif montre que la virtualisation assurerait une réduction drastique de la consommation d'énergie : jusqu'à 95% en utilisant des machines capables d'héberger 20 VM.

Le résultat qu'on voulait avoir n'a pas été atteint car le nombre de machines physiques n'est pas resté constant. Les machines utilisateurs, transformées en VM, demandent de plus en plus de serveurs.

### 7. Conclusion :

La virtualisation permet de tirer parti de la puissance toujours croissante des processeurs.

Ses bénéfices directs ou indirects en termes de consolidation, de réduction de coûts, de flexibilité, de contribution à une plus grande fiabilité et à une résilience simplifiée des infrastructures font désormais l'unanimité.

Autres axes de virtualisation : poste de travail, réseaux, stockage.

Le débat pour les utilisateurs n'est donc plus dans le principe d'adoption de la virtualisation mais dans les modalités et l'étendue de cette adoption.

### Pour commencer sur ProxMox v4 :

Pour accéder au serveur ProxMox, qui est l'hyperviseur, il faut se connecter de manière sécurisée sur : <https://192.168.1.150:8006> (Un https, l'adresse IP du serveur, le numéro de port 8006 qui correspond au numéro de port qui attend une demande d'un client.)

Par défaut, le Realm est en « Linux PAM standard authentication », et la langue est en Anglais.

Pour se connecter :

**Utilisateur : root**

**Mot de passe : Salle105,**

Les 4 nœuds sont représentés sur le côté, et nommés ici pve1, pve2, pve3, pve4. Ils sont en verts s'ils fonctionnent, et rouge si ce n'est pas le cas. En ouvrant un nœud, on peut voir toutes les machines virtuelles qui ont été créées. Pour identifier de manière unique une machine virtuelle, on lui attribue un numéro unique : un VM ID.

On a différentes solutions pour stocker les images disques des machines virtuelles. On peut les stocker en local, ou créer des espaces de stockage qui ne sont pas localement sur le nœud mais sur un autre partage sur le réseau, par exemple sur le NAS.

On peut visualiser pour chaque nœud l'activité du serveur. Voir son matériel, voir le pourcentage d'utilisation du CPU, la mémoire utilisées, le disque dur, etc... Il y a également un graphique avec toutes ces informations. Ici, le serveur pve4 est différent des autres, il est plus puissant.

On peut gérer les nœuds par l'interface administrateur en ouvrant une console shell par exemple, car le noyau est une machine Linux.

## Créer une machine virtuelle sur un ProxMox v4 :

Il faut d'abord choisir son nœud et accéder à son adresse IP. On sélectionne l'onglet « Créer VM ».

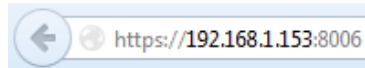
- On choisit ensuite le nœud (ici, **pve4**), le VM ID (qui doit être un numéro unique pour tout le cluster), un nom (compréhensible).
- On paramètre ensuite l'OS en choisissant un type d'OS pour avoir une aide à la configuration.
- Puis on sélectionne une iso, qui est partagée sur le NAS.
- On paramètre le disque dur, avec sa bus (IDE, SATA, VIRTIO, SCSI). Ici, la bus VIRTIO va améliorer les performances grâce aux pilotes VIRTIO sous Linux (pour Windows, il faudra préinstaller les pilotes VIRTIO avant de l'utiliser). On choisit de stocker l'image de notre machine virtuelle en local ou sur le stock du NAS. On choisit ensuite la taille du disque (10 GB ici). Le format de l'image sera laissé par défaut, c'est-à-dire en qcow2 (sinon, vmdk).
- La configuration du CPU n'est pas nécessaire.
- Pour configurer la mémoire, soit on bloque un espace mémoire de taille fixe, ou allouer une mémoire dynamiquement. Ici, on laissera **512 MB** en fixe.
- Le dernier onglet est la carte réseau, et doit être configurée en « Accès par pont », en choisissant la bonne carte réseau. Et le modèle de la carte sera VirtIO, paravirtualisé (à la place de Intel E1000 – standard –, ou Realtek, etc...).
- Une confirmation est demandée, et la VM est créée.

## Procédure pour une machine virtuelle sur ProxMox v4 :

Pour notre machine virtuelle, on utilisera le **nœud pve4** à l'adresse **https://192.168.1.153:8006** et de numéro **VM ID 209**. L'adresse IP utilisée par notre VM sera **192.168.1.208** et sera nommée **D8DETROZ**.

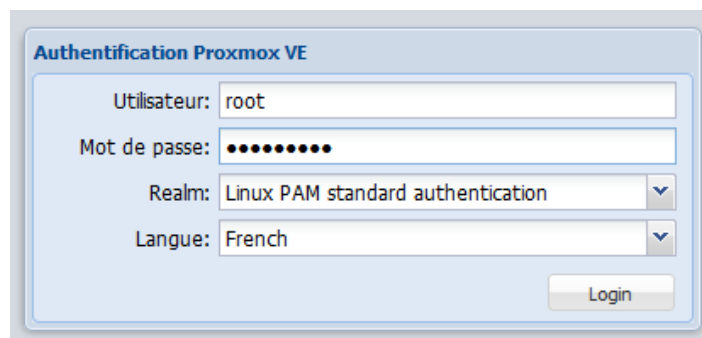
	Noeud	Adresse IP	VMID	Adresse IP Debian 8.2	Nom de la VM
Sébastien	PVE4	192.168.1.153	209	192.168.1.208	D8DETROZ

Pour se connecter, on accède donc à l'adresse :

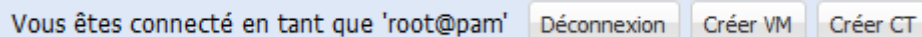


Et les identifiants suivants :

**Utilisateur : root**  
**Mot de passe : Salle105,**

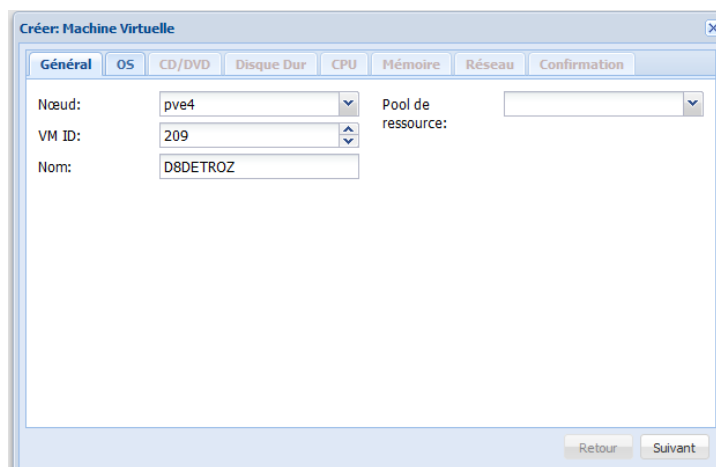


Pour créer une machine virtuelle, on utilise ensuite l'onglet en haut à droite de l'écran, nommé « **Créer VM** » :

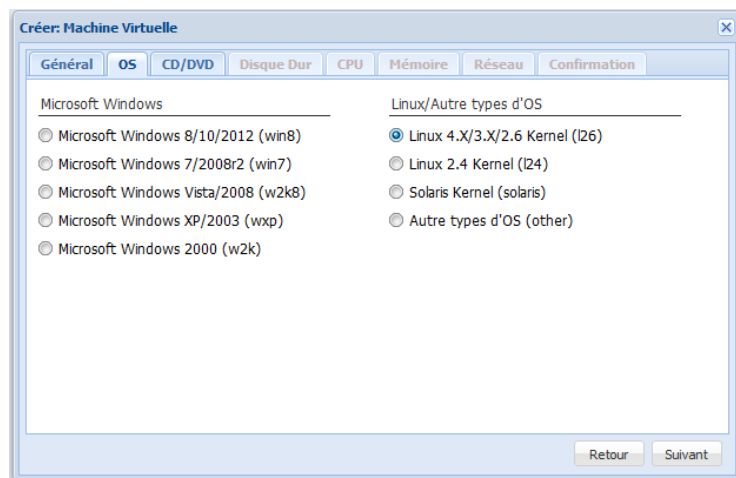


Pour la configuration générale, on utilise donc les informations citées plus haut, c'est-à-dire :

- *Nœud* : pve4
- *VM ID* : 209
- *Nom* : D8DETROZ



On présélectionne le type de notre OS a installé :



On sélectionne ensuite le fichier ISO qui contient notre OS :



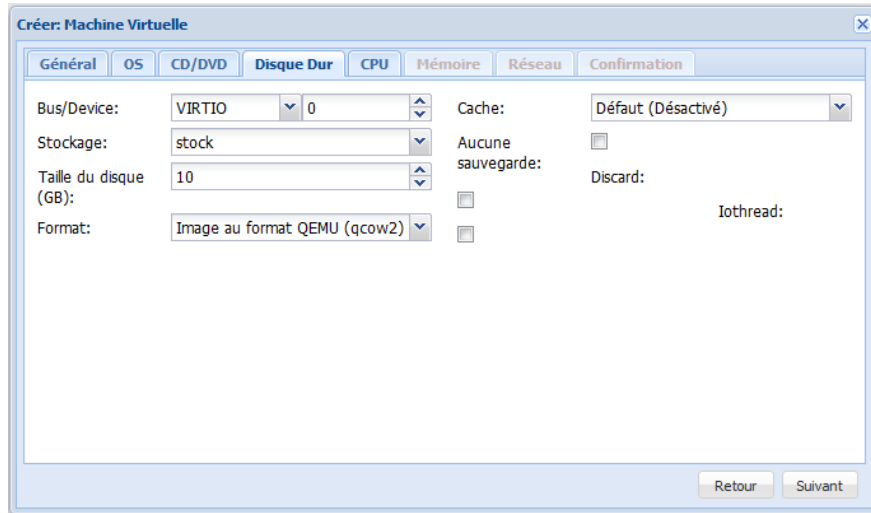
On créer ensuite le disque dur de notre OS :

*Bus/Device* : Le type VIRTIO permet une meilleure optimisation.

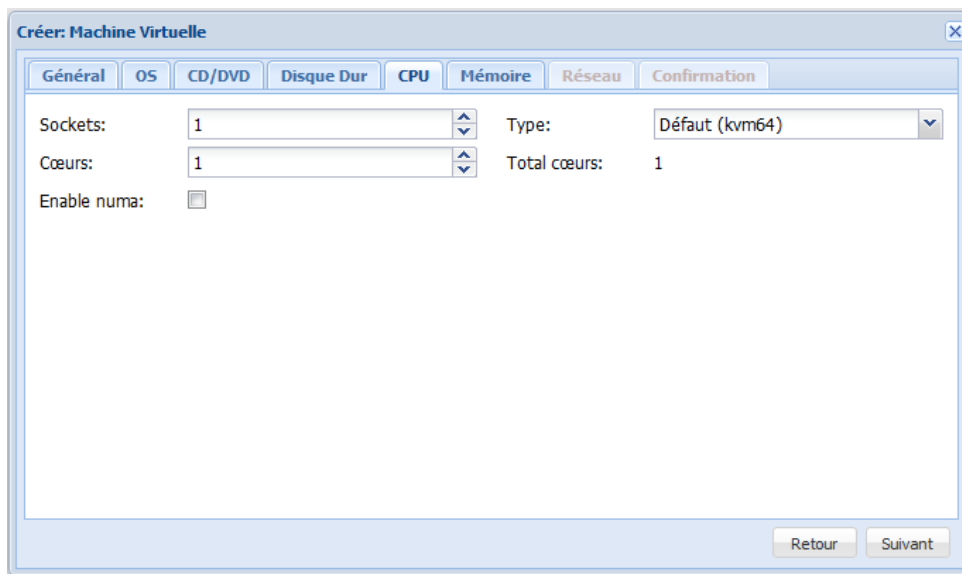
*Stockage* : L'image de notre disque dur sera stockée dans le fichier prévu à cet effet, le « stock ».

*Taille du disque* : 10 GB suffiront pour une distribution Linux.

*Format* : L'image sera au format qcow2 (cf. cours sur la virtualisation).

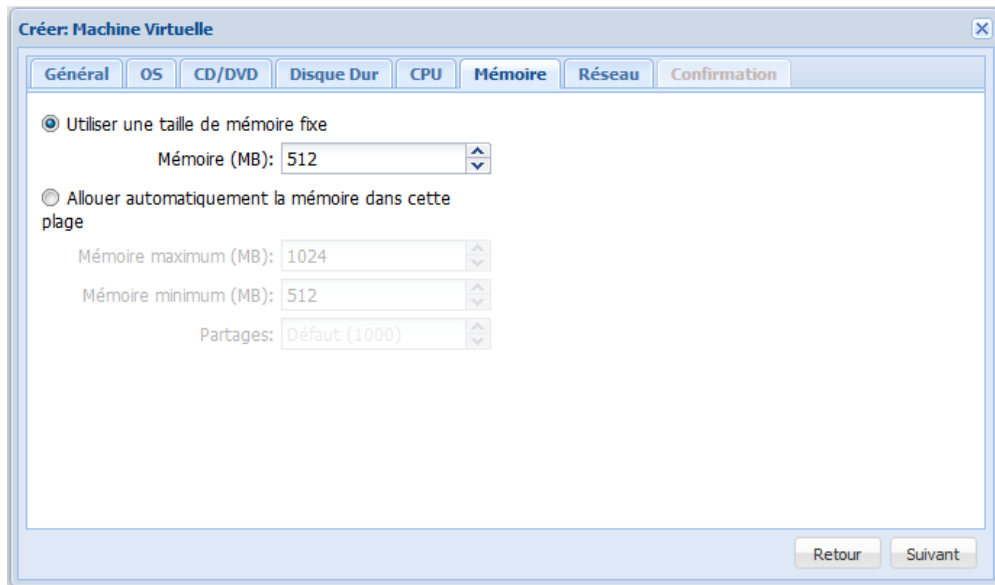


La configuration du processeur reste par défaut :

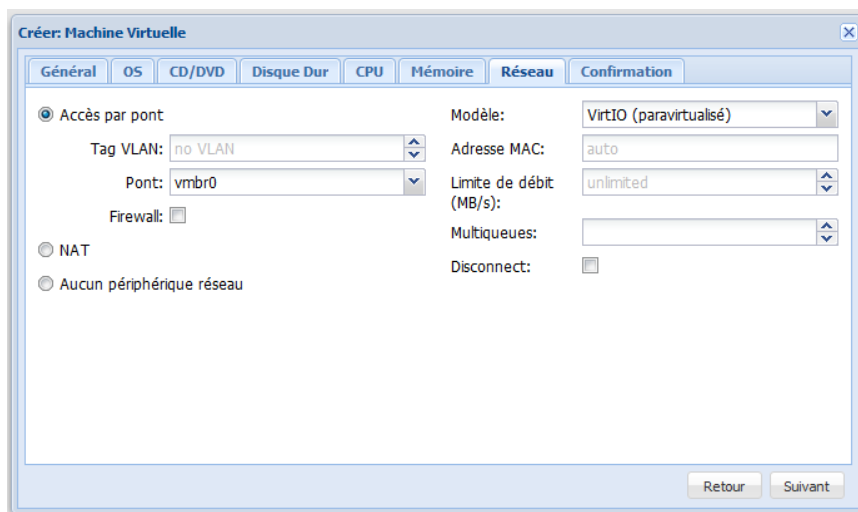


Au niveau de la mémoire, 512 MB suffiront pour notre Debian. Une taille fixe peut être utilisée, car nous n'aurons pas besoin de dépasser ce chiffre.

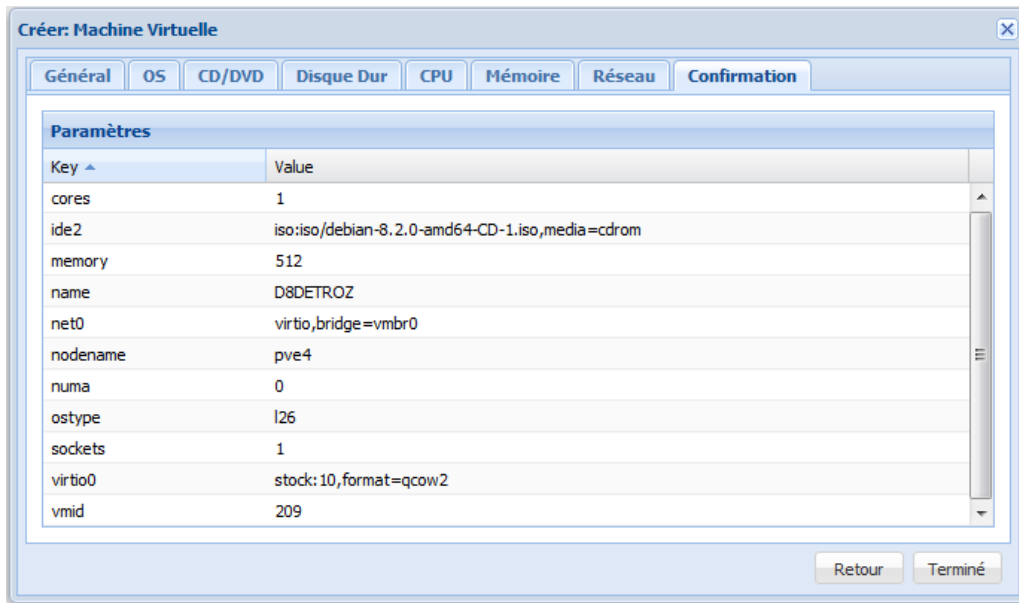




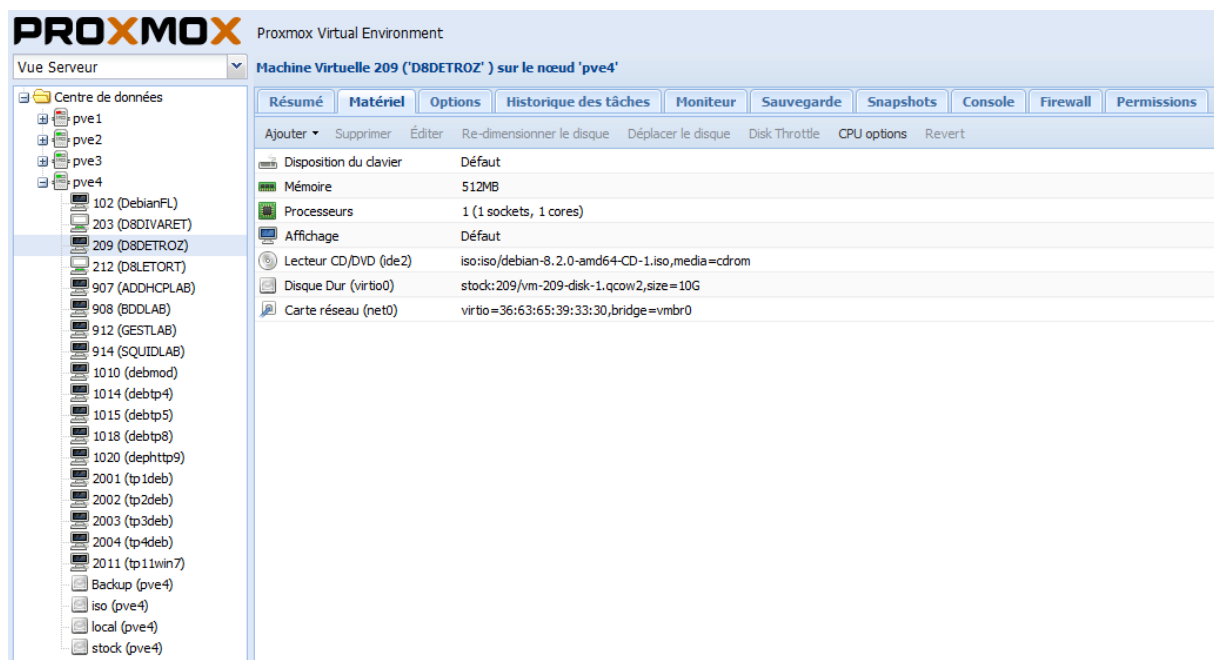
Au niveau du réseau, on laisse une configuration en « Accès par pont » sur la seule carte réseau disponible, soit « vmbro ». Le modèle de cette carte est en VirtIO (paravirtualisé).



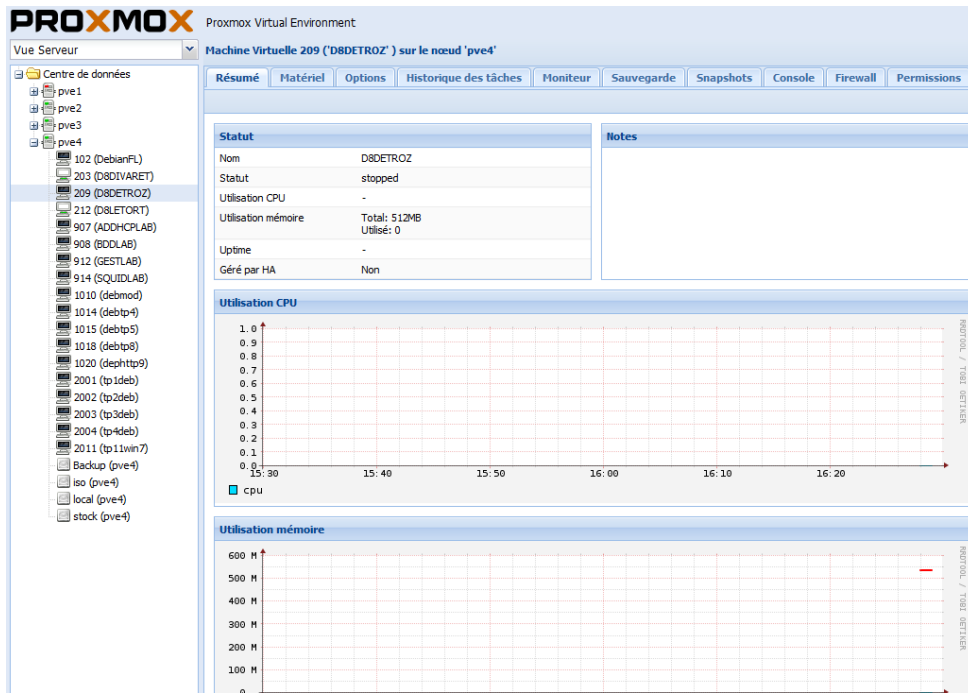
Un récapitulatif apparaît finalement, et la machine virtuelle est créée :



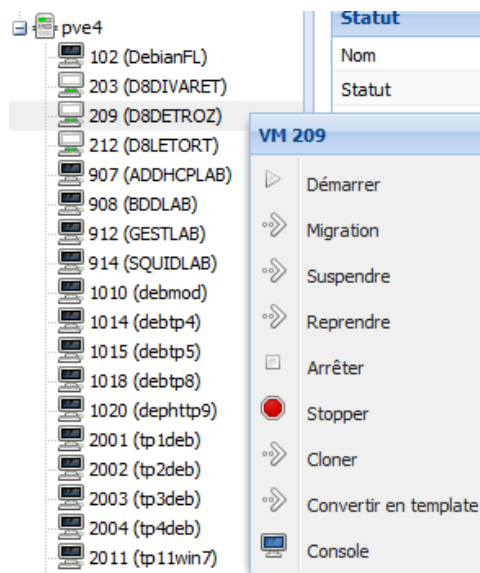
On peut ensuite voir la configuration matérielle de notre machine virtuelle :



Et les ressources qui sont utilisées par notre machine virtuelle :



Pour la démarrer, on fait un clic droit et on choisit « Démarrer ». Pour visualiser notre machine, on clique sur « Console ».



**ATTENTION, LE PAVÉ NUMÉRIQUE NE FONCTIONNE PAS SUR CETTE VIRTUALISATION !**

## Sauvegarde, migration et réplication de machines virtuelles avec Proxmox VE :

<b>Groupe</b>	<b>Noeud</b>	<b>Adresse IP</b>	<b>VMID</b>	<b>Adresse IP Debian 8.2</b>	<b>Nom de la VM</b>
Sébastien Guillaume	PVE2	192.168.1.151	103	192.168.1.203	Debian1DHD
Anthony	PVE2	192.168.1.151	113	192.168.1.213	Debian2DHD