

# Proxmox

- [\[PVE\] Création d'une VM Windows 11](#)
- [\[PVE\] Charger des ISO sur un stockage Proxmox](#)
- [\[PVE\] Création d'un réseau NAT sur Proxmox](#)
- [\[PVE\] Importer VM dans Proxmox \(VHDX to VM Proxmox\)](#)
- [\[PVE\] Supprimer l'avertissement de licence](#)
- [\[PVE\] Ajout d'un dépôt de sauvegarde PBS sur PVE](#)
- [\[PBS\] Proxmox Backup Server](#)
  - [\[PBS\] Création d'une tâche de sauvegarde](#)
  - [\[PBS\] Restaurer une sauvegarde](#)
  - [\[PBS\] Initialisation de disque et datastore](#)
- [Optimisation PVE/PBS](#)
  - [\[PVE\] Cache ARC dans RAM à l'utilisation d'un pool ZFS](#)

# [PVE] Création d'une VM Windows 11

Sur Proxmox, en haut a droite "Create VM"

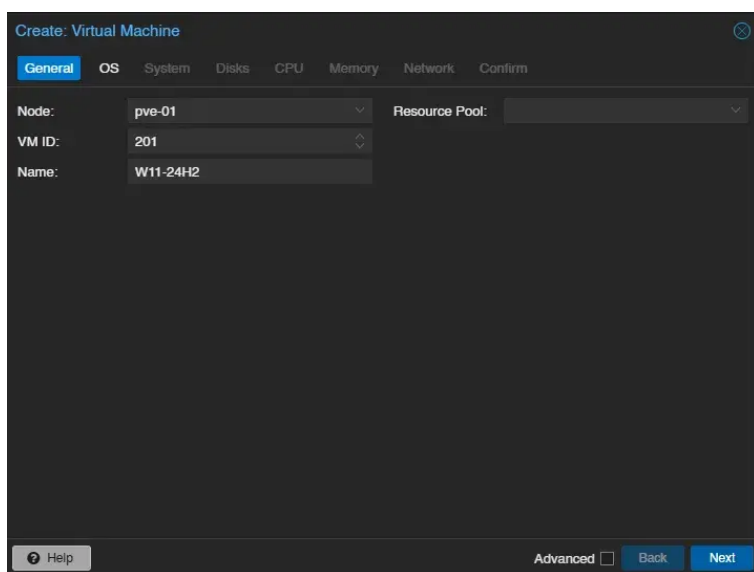
## Général

Node -> Node sur lequel la VM sera déployée

VM ID -> Spécifier un ID précis, ou laisser Proxmox incrémenter l'ID

Name -> Nom de la VM déployer

Ressource Pool -> Permet d'assigner la VM a un pool de gestion



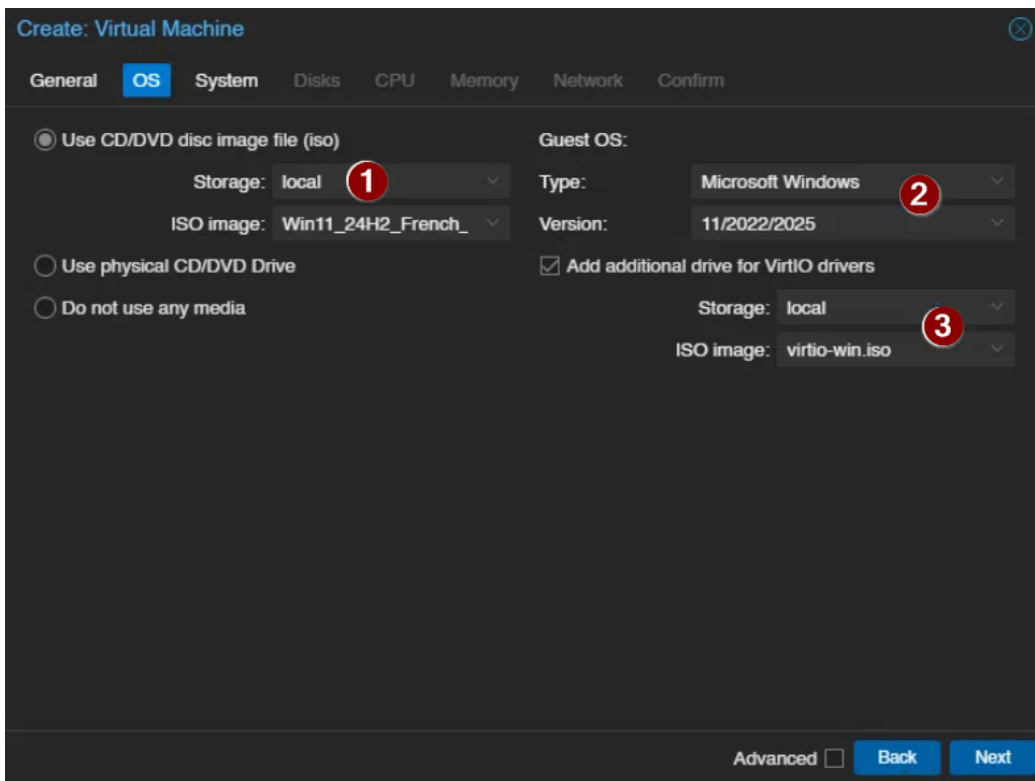
-> Next

## OS

Use CD/DVD disc image file (ISO) -> Choisissez l'image ISO d'installation de Windows 11

Guest OS -> Sélectionnez le type de système d'exploitation invité correspondant à Windows 11, à savoir le type "Microsoft Windows" puis la version "11/2022/2025".

Cochez l'option "Add additional drive for VirtIO drivers" et sélectionnez l'image ISO avec les pilotes VirtIO. Ceci va connecter un second lecteur CD/DVD à la VM pour vous donner accès facilement au pilotes



-> Next

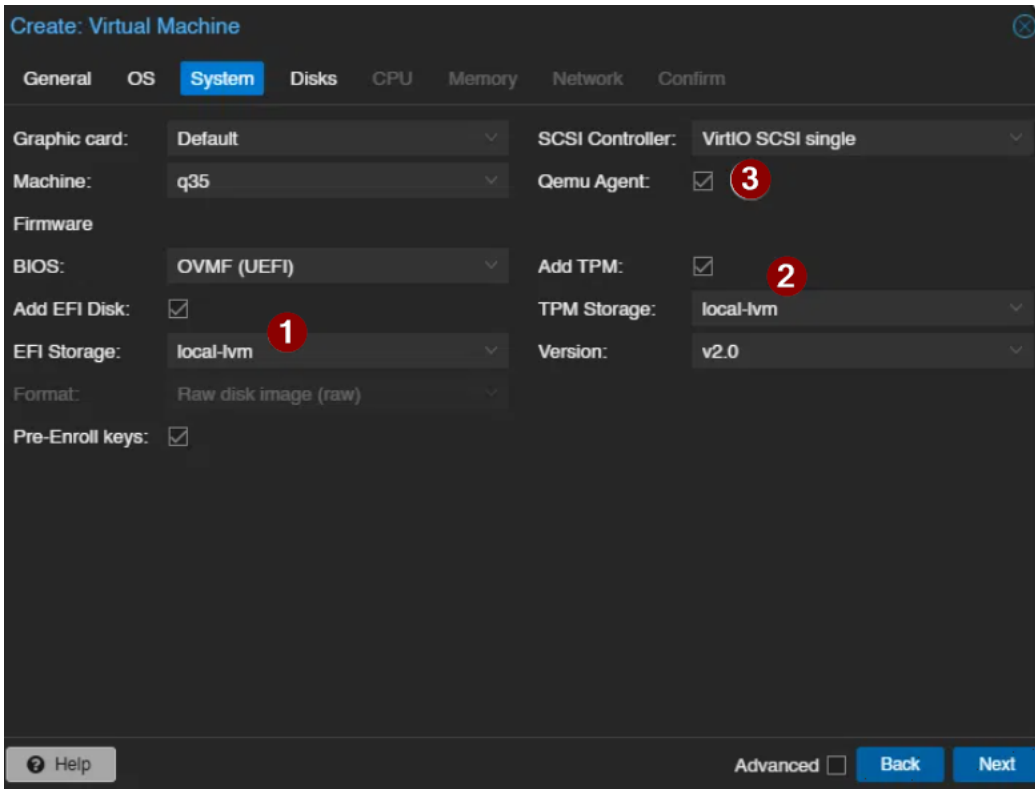
### System

Cette étape est important pour une VM sous Windows 11.

1 - Le type de BIOS "OVMF (UEFI)" est automatiquement sélectionné. Vous devez simplement choisir l'emplacement pour le disque EFI.

2 - La puce TPM est aussi activée par défaut comme nous avons sélectionné Windows 11 à l'étape précédente. Vous devez vous assurer que ce soit le cas, choisir un emplacement de stockage et aussi vérifier que c'est bien une puce TPM 2.0 qui est choisie.

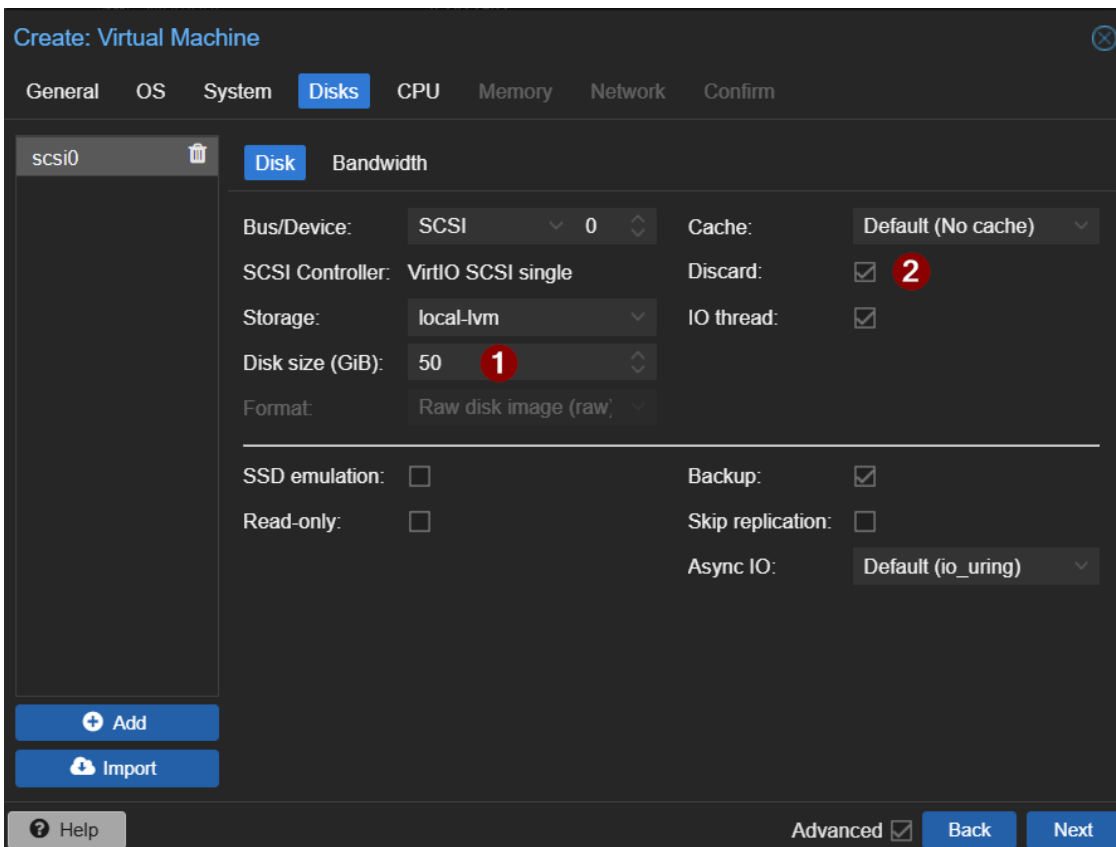
3 - Cochez l'option "Qemu Agent". L'agent QEMU pour Windows est un petit logiciel installé dans la VM qui permet à Proxmox d'interagir avec le système invité (par exemple pour effectuer un arrêt propre).



-> Next

## Disks

- 1 - Disk size (Gib) -> Choisir la taille du disk virtuel (50Go mini pour un Windows 11)
- 2 - Cocher l'option "Discard" pour une gestion du provisionnement dynamique sur les SSD

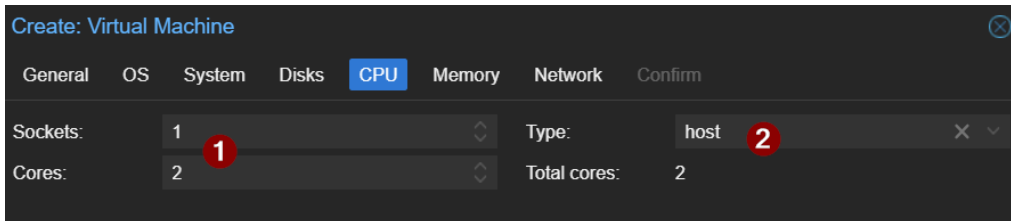


-> Next

## CPU

1 - "Sockets" et "Cores" -> Choisir le nombre de socket alloué et le nombre de cœurs virtuels (Attribuer au moins 2 cœurs à la machine virtuelle.)

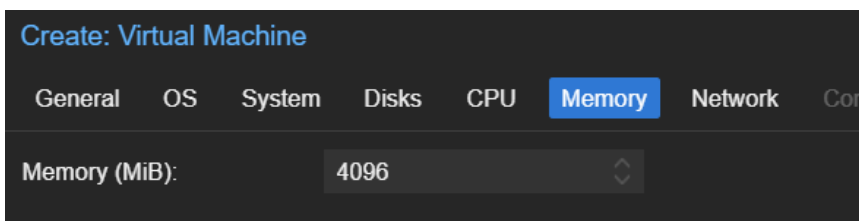
2 - Sélectionner le type de processeur "Host". Le CPU de type Host pour une VM Windows 11 permet d'exposer toutes les fonctionnalités du processeur physique, ce qui garantit la compatibilité avec les exigences matérielles de Windows 11 (et certaines applications).



-> Next

## Memory

Memory -> Attribuez au moins 4 Go de RAM (mémoire vive) à cette machine virtuelle. Là encore, c'est un prérequis de Windows 11



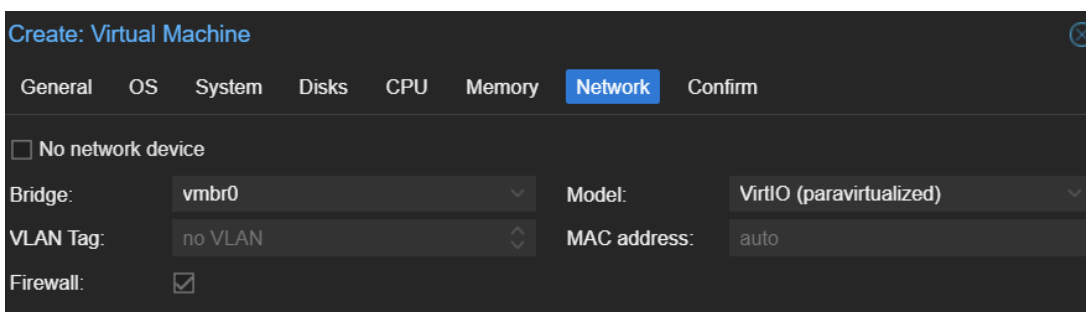
-> Next

## Network

Bridge -> Permet de connecter la VM a un réseau. Le réseau vmbr0 (créé par défaut).

Model -> Le modèle de carte réseau, par défaut "VirtIO (paravirtualized)". à ces limites selon le besoin en bande passante au-delà de 10Gobit/s

VLAN Tag -> Permet de tagger la caret réseau de la VM sur un VLAN précis



-> Next

Vérifiez votre configuration et cliquez sur le bouton "**Finish**" pour finaliser la création de la VM.



# [PVE] Charger des ISO sur un stockage Proxmox

Lien téléchargement Windows 11

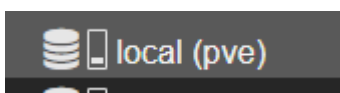
<https://www.microsoft.com/fr-fr/software-download/windows11>

Lien téléchargement Pilote VirtIO pour Windows

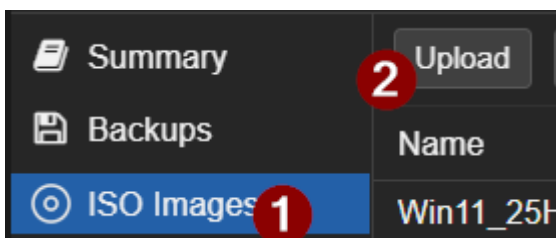
<https://fedorapeople.org/groups/virt/virtio-win/direct-downloads/stable-virtio/virtio-win.iso>

## Upload local

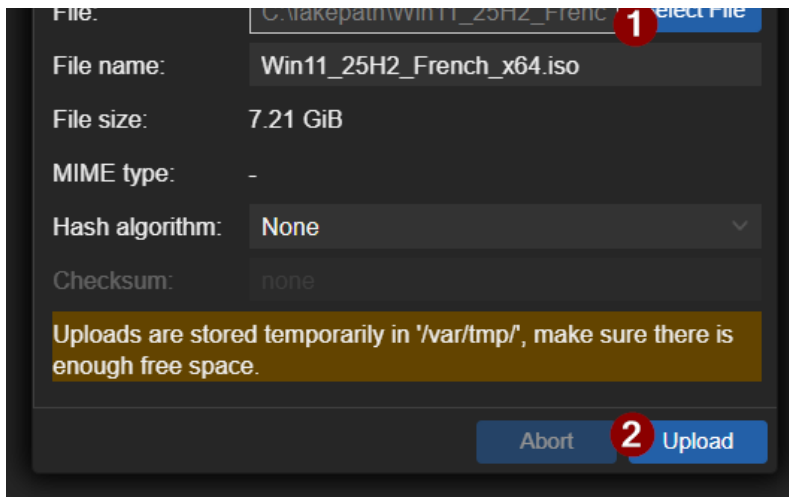
Cliquer sur le stockage souhaitez (Exemple avec le stockage local de base)



Cliquer sur "ISO Images" 1 puis "Upload" 2



Choisir le fichier à charger via "Select File" puis "Upload"



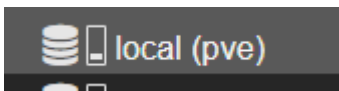
Le fichier est maintenant sur le

stockage local

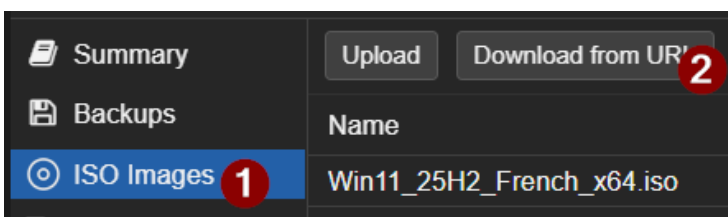
Name	Date	Format	Size
Win11_25H2_French_x64.iso	2026-04-09 08:59:50	iso	7.75 GB

## Upload via URL

Cliquer sur le stockage souhaité (Exemple avec le stockage local de base)



Cliquer sur "ISO Images" 1 puis "Download from URL" 2



Coller l'URL direct du téléchargement de l'iso puis validé avec "Query URL" 1, le nom du fichier ISO doit se remplir dans le champs "File name" 2 finir par "Download" 3

## Download from URL



URL:  **1** Query URL

File name:  **2**

File size: 7.89 GiB

MIME type: application/octet-stream

Hash algorithm:

Verify certificates:

Checksum:

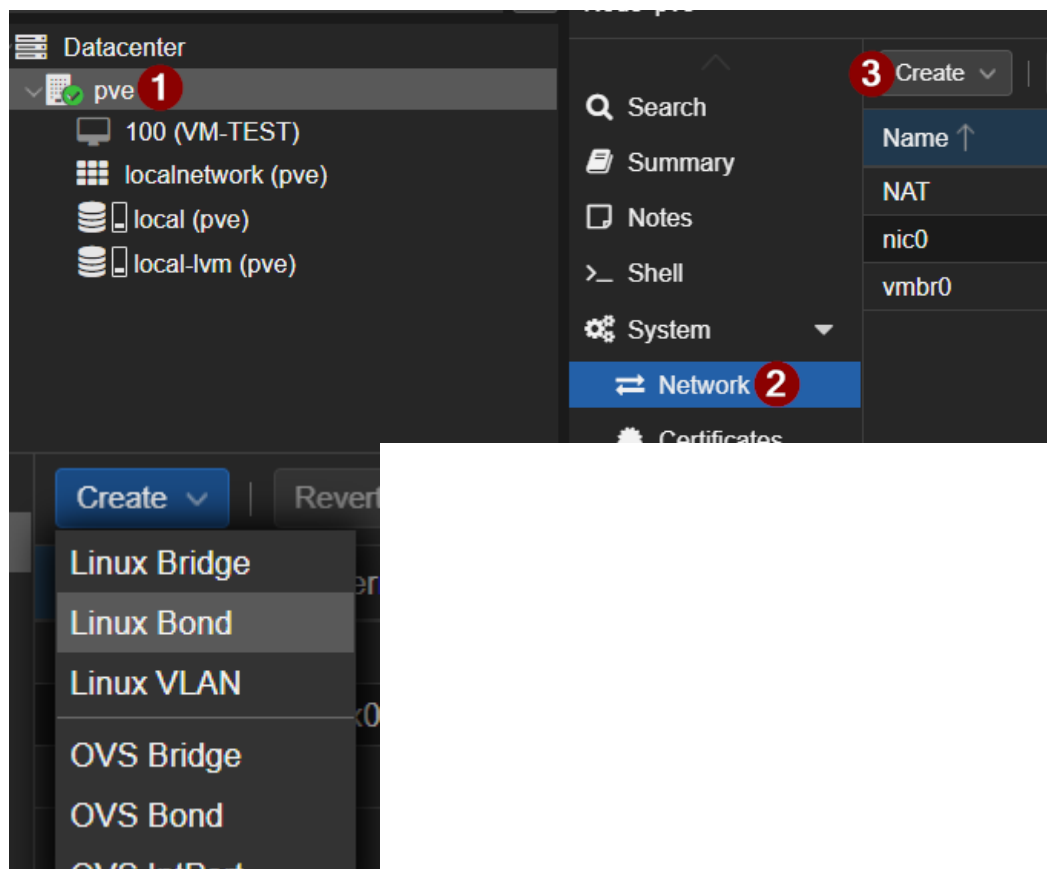
Decompression algorithm:

Advanced  **3** Download

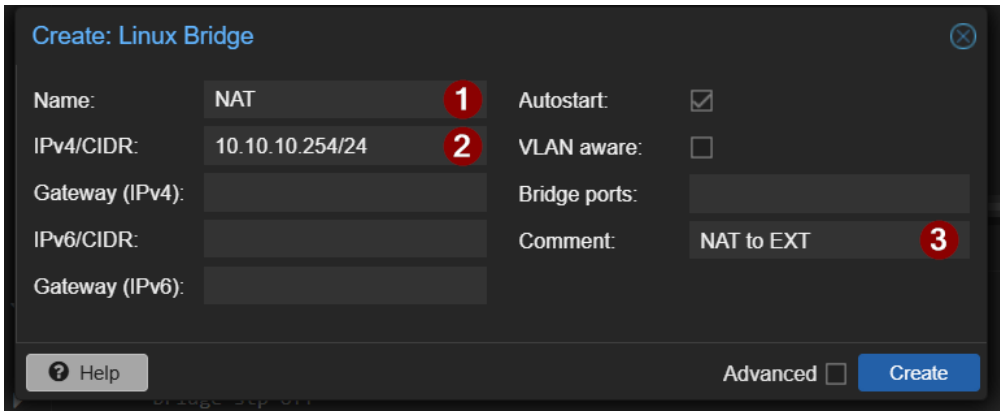
# [PVE] Création d'un réseau NAT sur Proxmox

## Création d'une nouvelle interface Linux Bridge

- 1 -> Sur votre node, ici PVE
- 2 -> "Network"
- 3 -> "Create" -> "Linux Bridge"

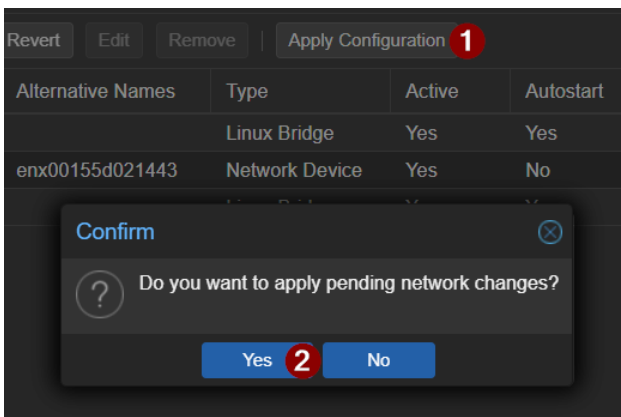


- 1 -> Nommée votre Linux Bridge
- 2 -> IPv4/CIDR -> Assigné une IP à votre interface Linux Bridge  
Cette IP deviendra la passerelle de vos VM
- 3 -> Ajouter une notation dans l'espace "Comment"  
Validé par "Create"



1 -> Appliquer la configuration

2 -> Puis validé



## Configurer le NAT sur le réseau

Pour cela il faudra activer l'IP forwarding et utiliser des règles iptables pour mettre en place le NAT via l'IP masquerade.

Se connecter en SSH a son proxmox ou via l'onglet "Shell", puis modifier le fichier de conf réseau

```
nano /etc/network/interfaces
```

En dessous de votre Linux Bridge, ici "NAT", Ajoutée les lignes suivantes :

[Attention, il faudra adapter votre IP réseau en fonction de ce que vous avez choisi à la création du Linux Bridge](#)

```
post-up echo 1 > /proc/sys/net/ipv4/ip_forward
post-up iptables -t nat -A POSTROUTING -s '10.10.10.0/24' -o vbr0 -j MASQUERADE
post-down iptables -t nat -D POSTROUTING -s '10.10.10.0/24' -o vbr0 -j MASQUERADE
```

```
auto NAT
iface NAT inet static
    address 10.10.10.254/24
    bridge-ports none
    bridge-stp off
    bridge-fd 0
    post-up echo 1 > /proc/sys/net/ipv4/ip_forward
    post-up iptables -t nat -A POSTROUTING -s '10.10.10.0/24' -o vnet0 -j MASQUERADE
    post-down iptables -t nat -D POSTROUTING -s '10.10.10.0/24' -o vnet0 -j MASQUERADE
#NAT to EXT
source /etc/network/interfaces.d/*
```

Voici à quoi servent ces trois lignes, dans l'ordre :

- Active le routage IPv4 sur l'hôte Proxmox pour permettre le transfert de paquets entre interfaces réseau (IP forwarding).
- Ajoute une règle NAT qui remplace l'adresse source des paquets provenant du réseau 10.10.10.0/24 par celle de l'interface vnet0 (NAT vers l'extérieur).
- Supprime cette même règle NAT lorsque l'interface est arrêtée, pour garder une configuration propre.

Dans ce cas, le "vnet0", donne accès à Internet, il fera donc l'adapté également en fonction de votre configuration

- Sauvegarder la config et redémarrer le services réseau

```
systemctl restart networking.service
```

# [PVE] Importer VM dans Proxmox (VHDX to VM Proxmox)

Copier le VHDX sur le serveur, via Filezilla

Vérification de l'état de l'image sources .vhdx

```
qemu-img check -r all X.vhdx
```

Conversion en format RAW :

X = fichier sources à convertir

Y = Nom du fichier de destination (vm-'id\_vm'-disk-'numéro\_disk')

```
qemu-img convert -O raw 'X' 'Y'
```

Exemple pour le disk 0 de la vm id 101:

```
« qemu-img convert -O raw vm-hyperv.vhdx vm-101-disk-0.raw »
```

Ajout du disque à la VM dans le pool de stockage.

X = Numéro VM

Y = Nom du disque à importer

Z = pool de stockage

```
qm importdisk 'X' 'Y' 'Z'
```

Exemple pour l'import du disk nommé vm-101-disk-0 dans le pool DATA:

```
« qm importdisk 101 vm-101-disk-0.raw DATA »
```

# [PVE] Supprimer l'avertissement de licence

Se connecter en SSH sur le PVE en question.

Accéder au répertoire :

```
cd /usr/share/javascript/proxmox-widget-toolkit
```

Créer une copie du fichier :

```
cp proxmoxlib.js proxmoxlib.js.save
```

Utiliser nano pour éditer le fichier :

```
nano proxmoxlib.js
```

Trouver la partie à modifier

Utiliser Ctrl+W et rechercher :

“No valid subscription”

Remplacer “Ext.Msg.show” avec “void”

```
.data.status.toLowerCase() !== 'active') {  
  Ext.Msg.show({  
    title: gettext('No valid subscription'),  
    icon: Ext.Msg.WARNING,
```

```
let res = response.result;  
if (res === null || res === undefined || !res || re  
.data.status.toLowerCase() !== 'active') {  
  void({  
    title: gettext('No valid subscription'),  
    icon: Ext.Msg.WARNING,
```

Redémarrer le service web Proxmox :

```
systemctl restart pveproxy.service
```

# [PVE] Ajout d'un dépôt de sauvegarde PBS sur PVE

Sur l'interface web du PVE dans **Datacenter** > **Storage** :

[image.png](#)

Puis cliquez sur **Add** > **Proxmox Backup Server** et ajoutez toutes les informations requises :

[image.png](#)

Vous pouvez récupérer le Fingerprint de votre serveur PBS depuis le **Dashboard** l'interface web de PBS :

[image.png](#)

# [PBS] Proxmox Backup Server

[PBS] Proxmox Backup Server

# [PBS] Création d'une tâche de sauvegarde

Pour l'exemple, nous allons créer une sauvegarde quotidienne de notre VM **debian-test** en nous rendant sur l'interface web de PVE dans le menu **Datacenter > Backups** puis cliquez sur **Add** :

image.png

Dans l'onglet **Retention** vous pouvez configurer les sauvegardes qui seront conservées de manière journalière, hebdomadaire, mensuelle ou même annuelle :

image.png

Ici, la sauvegarde sera effectuée tous les jours à 4h du matin et nous conserverons 5 sauvegardes journalières, 1 hebdomadaire, 1 mensuelle et 1 annuelle.

Une fois que vous aurez cliqué sur **Create**, votre tâche planifiée de sauvegarde sera programmée.

Vous pouvez cliquer sur **Run Now** pour lancer et tester votre sauvegarde.

[PBS] Proxmox Backup Server

# [PBS] Restaurer une sauvegarde

Si vous avez besoin de restaurer une sauvegarde, vous pouvez vous rendre sur l'interface web PVE, sélectionnez votre VM puis dans le menu **Backup**, cliquez sur **Restore** après avoir sélectionné votre sauvegarde :

[image.png](#)

[PBS] Proxmox Backup Server

# [PBS] Initialisation de disque et datastore

Pour cela, nous partons du principe que notre VM possède déjà d'un deuxième disque vierge dédié.

Rendez-vous sur l'interface web de PBS dans **Administration > Storage / Disks** :

image.png

Ici nous utiliserons le disque **/dev/sdb**

Commencez par créer une table de partition sur le disque en cliquant sur **Initialize Disk with GPT** après avoir sélectionné le disque :

image.png

Ensuite, rendez vous dans **Directory** et cliquez sur **Create: Directory** :

image.png

image.png

Le disque est maintenant formaté, monté et un datastore a automatiquement été créé dessus.

image.png

# Optimisation PVE/PBS

# [PVE] Cache ARC dans RAM à l'utilisation d'un pool ZFS

Lors de l'utilisation d'un pool ZFS pour le stockage des VMs. Le cache ARC utilise une grande partie de la RAM. Pour le limiter il faut :

Accéder au répertoire :

```
cd /etc/modprobe.d
```

Créer un fichier zfs.conf :

```
nano zfs.conf
```

Utiliser nano pour éditer le fichier :

```
nano proxmoxlib.js
```

Ajouter les lignes suivantes :

```
options zfs zfs_arc_min=4294967296
options zfs zfs_arc_max=17179869184

options zfs l2arc_noprefetch=0
```

```
options zfs zfs_arc_min=4294967296
options zfs zfs_arc_max=17179869184
```

```
options zfs l2arc_noprefetch=0
```

Le cache ARC sera limiter à :

4 Go minimum

16 Go maximum

Redémarrer le Noeuds Proxmox :

```
reboot
```

La commande, permet de connaitre l'état du cache ARC:

arc\_summary

```
ARC size (current):                10.1 %    1.6 GiB
  Target size (adaptive):          25.0 %    4.0 GiB
  Min size (hard limit):           25.0 %    4.0 GiB
  Max size (high water):            4:1    16.0 GiB
  Max size (low water) (100%):     10.0 %    1.6 GiB
```