



# Migration de deux serveurs ESXi vers Proxmox

Tunui Franken



---

## Table des matières

<b>1</b>	<b>But de la manœuvre</b>	<b>3</b>
<b>2</b>	<b>Infrastructure existante</b>	<b>3</b>
2.1	Artemis . . . . .	3
2.2	Hera . . . . .	4
<b>3</b>	<b>Problématiques et solutions envisagées</b>	<b>5</b>
<b>4</b>	<b>Feuille de route</b>	<b>5</b>
<b>5</b>	<b>Export des VMs en OVF vers le NAS</b>	<b>5</b>
<b>6</b>	<b>Mise en place de Proxmox sur les hyperviseurs</b>	<b>7</b>
6.1	Installation . . . . .	7
6.2	Paramétrage . . . . .	8
<b>7</b>	<b>Ajout du NAS aux hyperviseurs Proxmox</b>	<b>9</b>
7.1	Ajout du client sur le serveur NFS . . . . .	9
7.2	Ajout du stockage NAS sur l'hyperviseur . . . . .	9
<b>8</b>	<b>Import des VMs sur les hyperviseurs Proxmox</b>	<b>10</b>
<b>9</b>	<b>Ajout des cartes réseau sur les VMs</b>	<b>11</b>
<b>10</b>	<b>Ajout des guest-agent (équivalent des VMware Tools)</b>	<b>11</b>
10.1	VMs Linux . . . . .	11
10.2	VMs Windows . . . . .	12
10.3	Test de la communication entre l'hôte et les guest-agent . . . . .	12
<b>11</b>	<b>Sources</b>	<b>12</b>

# 1 But de la manœuvre

L'infrastructure interne possède deux serveurs ESXi, une pour les tests (**artemis**) et une en production (**hera**). La licence VMware vSphere 6 Hypervisor montre ses limites (pas de migration de VM, pas de clonage ou de déploiement de VM).

Étant un logiciel libre, Proxmox permet cela gratuitement. C'est donc la solution adoptée, il s'agit donc de migrer l'infrastructure existante vers Proxmox.

Pour mettre les deux hyperviseurs en cluster, un NAS sera utilisé comme datastore commun. Ce NAS est pour l'instant utilisé pour stocker des ISOs. Il aura donc à terme une fonction de datastore et stockage pour les ISOs, ainsi que les backups.

L'hyperviseur de test est séparé de l'hyperviseur de production par VLAN. Nous voulons garder cela. Mais pour bénéficier des fonctionnalités que permet Proxmox, il faut que les deux hyperviseurs puissent communiquer. Nous allons donc par la suite acheter un hyperviseur supplémentaire pour le mettre dans le VLAN de test. De cette façon, il y aura deux hyperviseurs de test et un hyperviseur de production.

## 2 Infrastructure existante

### 2.1 Artemis

Artemis est le serveur "No Prod". C'est un Supermicro Super Server.

- 6 CPUs à 2.00GHz, 12 processeurs logiques, 1 socket à 6 cœurs, avec hyperthreading
- 31.89 Go de RAM
- Réseau :
  - nom d'hôte : `artemis.brain-networks.net`
  - @IP : `vmk0` : 172.17.0.130, `fe80::ec4:7aff:feaa:1c80`, Port Group DMZ Brain Admin ESXi
  - DNS : 172.17.0.150
  - passerelle : 172.17.0.1
  - IPv6 activé
  - 2 adaptateurs hôtes
  - `vSwitch0`, sur les deux NICs physiques (`vmnic0` et `vmnic1`) :
    - DMZ Prod Brain avec 0 VMs, ID de VLAN : 17
    - DMZ Brain avec 13 VMs, ID de VLAN : 0
    - DMZ Brain Admin ESXi port VMkernel `vmk0` (172.17.0.130), ID de VLAN : 0
- Stockage ·
  - 2 adaptateurs physiques, `vmhba0` et `vmhba1` :
    - modèle : Wellsburg AHCI Controller
    - pilote : `vmw_ahci`

- une banque de données : `datastore1`, VMFS5, 924 Go (241.7 livres) sur la partition 3 du disque ATA local

Artemis contient 13 VMs.

## 2.2 Hera

Hera est le serveur “Prod”. C’est un HP ProLiant DL360 G7.

- 12 CPUs à 2.40GHz, 24 processeurs logiques, 2 sockets à 6 cœurs, avec hyperthreading
- 131.99 Go de RAM
- Réseau :
  - nom d’hôte : `ESX-1.brain-networks.net`
  - @IP :
    1. `vmk0` : 172.17.1.130, `fe80::461e:a1ff:fe59:f118`
    2. `vmk1` : 172.18.1.130, `fe80::250:56ff:fe6b:f532`
    3. `vmk2` : 172.16.20.1, `fe80::250:56ff:fe63:acff`
  - DNS : 172.17.0.150
  - passerelle : 172.17.1.1
  - IPv6 activé
  - 4 adaptateurs hôtes
  - vSwitch0, sur les NICs physiques `vmnic0` et `vmnic3` :
    - VM DMZ Brain avec 3 VMs, ID de VLAN : 0
    - VM DMZ Prod avec 4 VMs, ID de VLAN : 17
    - DMZ Prod Brain administration port VMkernel `vmk0` (172.17.1.130), ID de VLAN : 17
  - vSwitch Interco NAS, sur le NIC physique `vmnic1` :
    - `Interco_NAS` port VMkernel `vmk1` (172.18.1.130), ID de VLAN : 0
  - vSwitch Management Secours, sur le NIC physique `vmnic2` :
    - `MGMT Secours` port VMkernel `vmk2` (172.16.20.1), ID de VLAN : 0
- Stockage ·
  - 1 adaptateur physique, `vmhba0` :
    - modèle : Smart Array P410i
    - pilote : `nhpsa`
    - node world wide name : `5001438018076c40`
  - une banque de données : `datastore1`, VMFS6, 830.5 Go (538.83 livres) sur la partition 3 du disque HP local
  - un NAS en NFS, 5.36 To (4.98 livres)

Hera contient 6 VMs.

### 3 Problématiques et solutions envisagées

1. **Les datastores de chaque hyperviseur sont locaux, on ne peut donc pas faire de migration ou de load balancing.**

Pour profiter du fait d'avoir deux hyperviseurs, il faut disposer d'un stockage partagé, donc accessible sur le réseau. Le NAS existant, qui pour l'heure ne sert que pour les ISOs, peut assumer ce rôle. Pour cela il faut une bande passante conséquente, et une redondance de liens.

Nous allons donc agréger deux interfaces du NAS en LACP (802.3ad). Nous installerons ensuite les VMs directement sur le NAS, ce qui les rendra accessibles par les deux hyperviseurs.

2. **Il faut assurer une séparation entre les VMs de production et les VMs de test.**

L'architecture actuelle met les deux hyperviseurs dans des VLANs séparés : une DMZ\_Prod pour les VMs de production et une DMZ\_Brain pour les VMs de test (Lab). L'idée de faire communiquer les hyperviseurs pour permettre des migrations et des déploiements en partageant un espace de stockage (NAS) va à l'encontre de cette séparation.

Nous allons donc dans un premier temps migrer les hyperviseurs dans l'architecture actuelle, chacun dans un VLAN séparé. Il faudra par la suite acheter un serveur physique pour ajouter un hyperviseur dans le VLAN Lab.

3. **Pour les ESXi, les datastores sont formatés en VMFS pour permettre l'écriture simultanée de plusieurs VMs.**

VMFS n'existe pas sous Proxmox, c'est un format propriétaire VMware. Écrire dans des datastores partagés formatés en NFS ou locaux formatés en `ext4` ne semble pas poser de problème.

### 4 Feuille de route

N'ayant que deux serveurs et aucun autre pouvant jouer un rôle d'hôte temporaire, les étapes de migration seront les suivantes sur chaque serveur :

- export des VMs en OVF vers le montage NFS
- écrasement de la machine hébergeant le serveur ESXi pour installer Proxmox
- paramétrage de Proxmox pour retrouver la configuration de l'ESXi
- importation des VMs précédemment exportées

### 5 Export des VMs en OVF vers le NAS

On va commencer par créer un nouveau dossier dans le NAS pour recevoir les fichiers OVF exportés. Pour cela, naviguer dans l'explorateur Windows : `\\172.17.0.30\esx_prod`, les identifiants sont dans le Keepass.

Nous créons à côté du dossier ISOs le dossier `exported`. Dans le dossier nouvellement créé nous créons deux dossiers : `artemis` et `hera`.

Nous allons utiliser `ovftool` sur notre poste pour exporter les OVF directement dans le NAS.

On commence par l'installer sur notre poste depuis le site de VMware.

On peut maintenant retirer les ISOs qui seraient attachés à nos VMs puis utiliser les commandes suivantes :

```
ovftool vi://root@172.17.0.130/<nom-VM> \\172.17.0.30\esx_prod\exported\artemis
ovftool vi://root@172.17.1.130/<nom-VM> \\172.17.0.30\esx_prod\exported\hera
```

Si on obtient un OVA, ne pas oublier d'extraire l'archive.

Le réseau actuel n'ayant pas encore été migré, le NAS est seulement joignable par un lien 100 Mbps. Cela ralentit considérablement le transfert. Parce que les transferts échouent et ont parfois besoin de continuer en dehors des horaires de travail, nous allons utiliser un script PowerShell :

```
cd 'C:\Program Files\VMware\VMware OVF Tool'

$LOG_FILE = "C:\Users\${env:USERNAME}\Documents\copy_to_nas.txt"

echo '' > $LOG_FILE

Net Use \\${NAS_IP}\esx_prod\exported\artemis /user:${NAS_USER} ${NAS_PASSWD}

.\ovftool.exe --overwrite "vi://${ESX_USER}:${ESX_PASSWD}@${ESX_IP}/${VM_NAME}"
  \\${NAS_IP}\esx_prod\exported\artemis *>> $LOG_FILE
.\ovftool.exe --overwrite "vi://${ESX_USER}:${ESX_PASSWD}@${ESX_IP}/${VM_NAME}"
  \\${NAS_IP}\esx_prod\exported\artemis *>> $LOG_FILE
.\ovftool.exe --overwrite "vi://${ESX_USER}:${ESX_PASSWD}@${ESX_IP}/${VM_NAME}"
  \\${NAS_IP}\esx_prod\exported\artemis *>> $LOG_FILE
#...

echo DONE >> $LOG_FILE
(Get-Date | Out-String).trim() >> $LOG_FILE
```

Il faut noter que l'export avec `ovftool` n'inclut pas les cartes réseau, qu'il faudra créer à la main sur les VMs importées dans Proxmox (voir 9). Pour chaque VM exportée, dans l'interface ESXi, nous notons donc les adresses MAC des cartes réseau. Nous ajoutons ces informations dans un fichier texte que l'on met dans chaque dossier d'export VM (sur le NAS) pour noter le nombre de cartes réseau à créer. L'adresse MAC sera réutilisée, notamment pour les licences se basant sur l'adresse MAC.

## 6 Mise en place de Proxmox sur les hyperviseurs

### 6.1 Installation

Proxmox sera installé directement, en écrasant l'ancien hyperviseur présent, parce qu'on ne peut pas faire d'auto-migration avec VMFS5.

Au jour de l'écriture de la présente procédure, la dernière version de Proxmox VE est la version 6.4.

On flashe une clé USB avec l'ISO et suit la procédure d'installation une fois booté sur la clé :

```
Install Proxmox VE
```

On accepte la licence et on commence le partitionnement. En cliquant sur `Options`, on vérifie que le système de fichiers est bien `ext4`, comme nous ne faisons pas de RAID logiciel.

On choisit la localisation et l'horaire, ce qui fournira un miroir proche et permettra de choisir un clavier `azerty`.

Il faut ensuite choisir le mot de passe `root` et un email, qu'on n'oubliera pas d'ajouter dans le `Keepass`.

La dernière étape consiste à ajouter les informations de réseau. Vérifier :

— **Artemis** :

Il y a deux cartes physiques, on va en utiliser une pour le management et une pour la connexion avec le NAS.

— Management :

- @IP : 172.17.0.130/24
- Passerelle : 172.17.0.1
- DNS : 172.17.0.1

— Interconnexion NAS :

- @IP : 172.17.0.131/24
- Passerelle : 172.17.0.1
- DNS : 172.17.0.1

À l'installation, Proxmox ne configure que l'interface de management. On peut ajouter les autres adresses IP plus tard.

— **Hera** :

Il y a quatre cartes physiques, mais nous en avons pour l'instant besoin que d'une. On va seulement utiliser la carte à 100 Mbits pour le management.

- @IP : 172.17.1.130/24
- Passerelle : 172.17.1.1
- DNS : 172.17.1.1

Pour Hera, il va falloir ajouter le VLAN 17 dans la configuration du réseau car on est dans un VLAN tagué. Il faudra aussi penser à ajouter le tag sur les VMs à

l'important, même s'il faudra enlever les tags une fois le réseau migré.

Sur `/etc/network/interfaces` :

```
auto lo
iface lo inet loopback

iface ens3f0 inet manual

iface ens3f0.17 inet manual

auto vubr0v17
iface vubr0v17 inet static
    address 172.17.1.130
    netmask 255.255.255.0
    gateway 172.17.1.1
    bridge_ports ens3f0.17
    bridge_stp off
    bridge_fd 0

auto vubr0
iface vubr0 inet manual
    bridge-ports ens3f0
    bridge-stp off
    bridge-fd 0

iface ens3f1 inet manual

iface ens4f0 inet manual

iface ens4f1 inet manual
```

Après un redémarrage, nous pouvons nous rendre sur `https://<ip-hyperviseur>:8006/`

## 6.2 Paramétrage

On commence par se connecter avec les identifiants ajoutés lors de l'installation.

### 6.2.1 Vérification des sources apt

Proxmox installe des sources apt en fonction de la souscription. En effet Proxmox offre un support payant et optionnel. Si les sources installées correspondent à la version payante, tous les apt update échoueront.

Par défaut, le dépôt activé est deb `https://enterprise.proxmox.com/debian/pvebuster pve-enterprise`, dans le fichier `/etc/apt/sources.list.d/pve-enterprise.list`. Ce dépôt correspond à la version payante. Dans notre cas nous allons le désactiver en le commentant dans le fichier :



```
root@artemis:~# cat /etc/apt/sources.list.d/pve-enterprise.list
#deb https://enterprise.proxmox.com/debian/pve buster pve-enterprise
```

On peut maintenant rajouter le dépôt no-subscription. Proxmox recommande de l'ajouter dans le fichier `/etc/apt/sources.list` directement :

```
root@artemis:~# cat /etc/apt/sources.list
deb http://ftp.fr.debian.org/debian buster main contrib
deb http://ftp.fr.debian.org/debian buster-updates main contrib

# pve-no-subscription repository, disable pve-enterprise if using this one
deb http://download.proxmox.com/debian/pve buster pve-no-subscription

# security updates
deb http://security.debian.org buster/updates main contrib
```

Il faut ensuite bien-sûr mettre à jour le cache ce qui nous confirmera par la même occasion que les dépôts fonctionnent. On en profite pour faire une mise à jour :

```
root@artemis:~# apt update && apt full-upgrade
```

## 7 Ajout du NAS aux hyperviseurs Proxmox

### 7.1 Ajout du client sur le serveur NFS

Sur le NAS (Buffalo), on doit ajouter le client correspondant à l'hyperviseur qui va monter le partage NFS. Pour cela, dans la partie NFS, onglet NFS Rules, cliquer sur Add. On peut alors choisir le dossier à partager (on sélectionne `esx_prod`). Dans la partie Hosts, on ajoute l'adresse IP de l'hyperviseur. On met `Read & write` en privilège et on active la synchronisation.

On peut maintenant passer au montage du NAS dans l'hyperviseur.

### 7.2 Ajout du stockage NAS sur l'hyperviseur

Pour monter le NAS sur les hyperviseurs Proxmox, pas besoin de toucher au fichier `/etc/fstab`.

On doit d'abord trouver le chemin de l'export. Pour cela, on utilise la commande suivante :

```
# pvesm nfsscan <ip-du-nas>
```

On peut alors modifier le fichier `/etc/pve/storage.cfg`.

```
nfs: nas
    path /mnt/pve/nas
    server 172.17.0.30
    export /mnt/array1/esx_prod
```

```
options vers=3,soft
content iso,vztmpl,images
```

On peut aussi le faire en graphique. Pour cela, dans **Datacenter** de la colonne de gauche, cliquer sur **Storage**, puis **Add**. Il suffit de renseigner les informations du NAS comme dans le fichier texte `/etc/pve/storage.cfg`.

## 8 Import des VMs sur les hyperviseurs Proxmox

Nous ouvrons la ligne de commande de Proxmox VE et nous utilisons la commande suivante pour créer de nouvelles VMs avec les informations des fichiers OVF.

```
qm importovf <vmid> </chemin/nas/fichier.ovf> <stockage>
```

Pour obtenir le prochain VMID disponible, on peut utiliser `pvesh get /cluster/nextid`.

La commande d'import peut donc être :

```
qm importovf $(pvesh get /cluster/nextid) /mnt/pve/nas/exported/artemis/vm-
example/vm-example.ovf> nas
```

Cette commande est à répéter pour chaque VM à importer.

Comme les cartes réseaux ne sont pas incluses dans les exports OVF, il faudra configurer le réseau à la main dans un second temps (voir 9).

Pour les VMs sous Windows, il faut spécifier que l'on veut utiliser UEFI au lieu de BIOS :

```
qm set <vmid> --bios ovmf
```

Toujours pour les VMs sous Windows, nous allons utiliser IDE en attendant d'installer les drivers permettant d'utiliser SCSI :

```
sed -i 's/scsi/sata/g' /etc/pve/qemu-server/<vmid>.conf
```

Nous pouvons donc scripter l'import des VMs de la façon suivante :

```
#!/bin/bash

for vm in /mnt/pve/nas/exported/artemis/*; do
  vmid=$(pvesh get /cluster/nextid)
  vm_name=$(basename "$vm")

  echo "Import de $vm_name en utilisant le VMID $vmid..."
  qm importovf "$vmid" "/mnt/pve/nas/exported/artemis/${vm_name}/${vm_name}.ovf"
  local-lvm
  echo "Terminé"

  # On sélectionne les VMs Windows à la main
```

```
if [[ "$vm_name" = VM-10* ]] || [[ "$vm_name" = VM-45* ]]; then
    echo "$vm_name semble être un VM Windows (VMID $vmid)"
    qm set "$vmid" --bios ovmf && echo "$vm_name mise en UEFI"
    sed -i 's/scsi/sata/g' "/etc/pve/qemu-server/${vmid}.conf" &&
    echo "$vm_name mise en IDE en attendant les drivers SCSI"
fi

echo
done
```

## 9 Ajout des cartes réseau sur les VMs

Nous avons créé un fichier texte par VM, que nous avons mis dans le dossier d'export contenant l'OVF et les disques. Ce fichier texte contient les adresses MAC des cartes réseaux, pour les réutiliser (ce qui est notamment nécessaire pour certaines licences). Il permet aussi de recenser le nombre de cartes réseau à créer, puisque nous le faisons à la main dans l'interface graphique de Proxmox.

Pour ce faire, on clique sur la VM à configurer (dans la liste de gauche). Dans la section **Hardware** de la VM, on peut voir qu'il n'y a pas de carte réseau. Il suffit de sélectionner **Add > Network Device**. On peut laisser le nom du bridge par défaut, `vmbrio`. Le modèle par défaut est **Intel E1000**, on va garder ce choix sauf au cas où la VM ne reconnaîtrait pas cette carte. Pour cela, on fait au cas par cas, en notant que les drivers n'étant pas encore installés, aucune carte ne sera reconnue pour l'instant. Cela est fait dans la section 10.

## 10 Ajout des guest-agent (équivalent des VMware Tools)

### 10.1 VMs Linux

Il faut installer le paquet et activer le service correspondant.

Sur les dérivés de Debian :

```
# apt install qemu-guest-agent
```

Sur les dérivés de Red Hat :

```
# yum install qemu-guest-agent
```

Pour être sûr que tout fonctionne, on va activer le service, ce qui peut déjà être fait selon les distributions :

```
# systemctl enable qemu-guest-agent && systemctl start qemu-guest-agent
```

## 10.2 VMs Windows

Pour les VMs Windows il faut d'abord télécharger l'ISO du driver virtio-win. Nous mettons cet ISO sur le NAS avec les autres ISOs.

Pour installer le driver :

- attacher l'ISO sur la VM
- se rendre dans le Gestionnaire de Périphériques
- chercher *PCI Simple Communications Controller*
- faire un clic droit et *mettre à jour le driver*, sélectionner dans le montage de l'ISO :  
DISQUE:\vioserial\

On peut ensuite installer `qemu-guest-agent`. Pour cela il y a un MSI dans le dossier `guest-agent` dans le montage de l'ISO.

Pour être sûr que l'agent tourne, on va ouvrir une invite PowerShell :

```
PS C:\Users\Administrateur> Get-Service QEMU-GA
```

```
Status Name          DisplayName
-----
Running QEMU-GA       QEMU Guest Agent
```

S'il ne tourne pas, il faut l'activer dans les services.

## 10.3 Test de la communication entre l'hôte et les guest-agent

Si la communication se fait correctement, la commande suivante devrait retourner sans erreur :

```
# qm agent <vmid> ping
```

## 11 Sources

[https://pve.proxmox.com/wiki/Migration\\_of\\_servers\\_to\\_Proxmox\\_VE](https://pve.proxmox.com/wiki/Migration_of_servers_to_Proxmox_VE)

[https://pve.proxmox.com/wiki/Windows\\_10\\_guest\\_best\\_practices](https://pve.proxmox.com/wiki/Windows_10_guest_best_practices)

<https://pve.proxmox.com/wiki/Qemu-guest-agent>

[https://pve.proxmox.com/wiki/Network\\_Configuration](https://pve.proxmox.com/wiki/Network_Configuration)

<https://kb.vmware.com/s/article/2147824>