

Mise en place d'un réseau informatique et configuration de **Proxmox VE**

Projet gsb.org



DUMAS Lucie

Table des matières

Comment mettre en place un réseau informatique ?	3
Configuration Physique du réseau	4
Installation du routeur	5
Installation du système d'exploitation Debian 12 « Bookworm »	5
Configuration du routeur	17
Installation du serveur	20
Installation du système d'exploitation Proxmox VE	20
Procédure de résolution de problèmes	32



Comment mettre en place un réseau informatique ?

Etape par étape, nous apprendrons à mettre en place et à configurer un réseau informatique. Notre réseau sera composé d'un routeur, d'un switch et d'un serveur. Nous commençons donc à définir l'adresse de notre réseau ainsi que les différents sous réseaux :

- Adresse réseau : 10.31.176.0/20
- Adresses sous-réseaux :
 - o LAN : 10.31.176.0/22
 - o Sous-réseau 2 : 10.31.180.0/22
 - o DMZ : 10.31.184.0/22
 - o Sous-réseau 4 : 10.31.188.0/22

Voici un schéma simplifié du réseau que nous allons mettre en place :

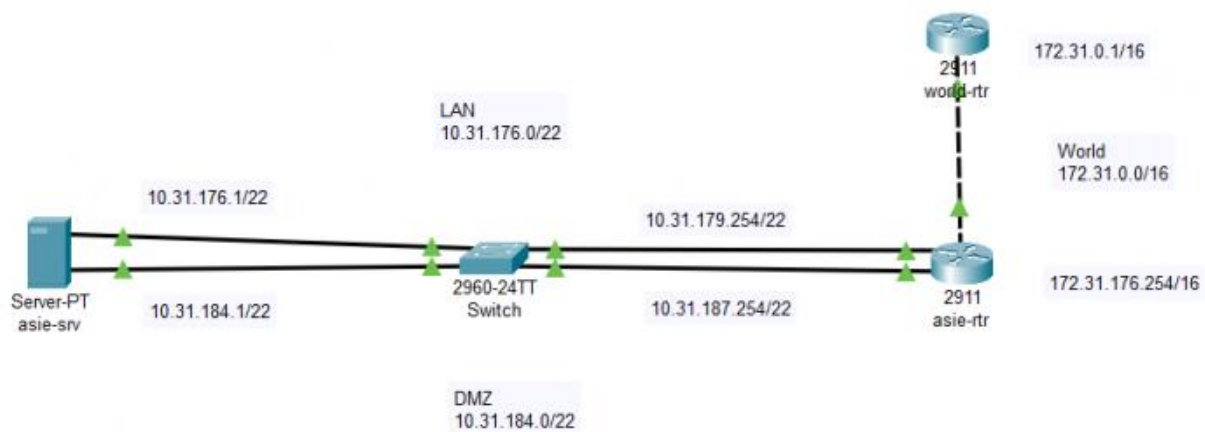


Figure 1 schéma du réseau informatique



Configuration physique du réseau

Nous utilisons des câbles RJ-45 pour relier notre serveur à notre switch, notre switch à notre routeur puis notre routeur au réseau :

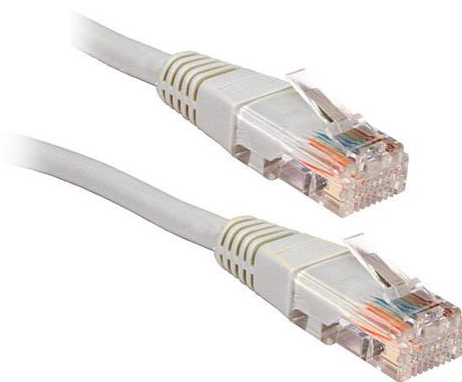


Figure 2 Câble Ethernet RJ-45

Voici les interfaces réseau de notre routeur ainsi que ses correspondances avec ses adresses IP :

- enp2s0 est l'interface branchée au mur : 172.31.176.254/16
- enp4s0 est l'interface du sous-réseau LAN (réseau privé) : 10.31.179.254/22
- enp5s0 est l'interface du sous-réseau DMZ (réseau publique) : 10.31.187.254/22

Voici les interfaces réseau de notre serveur ainsi que ses correspondances avec ses adresses IP :

- vmbr0 est l'interface du sous-réseau LAN (réseau privé) : 10.31.176.1/22
- vmbr1 est l'interface du sous-réseau DNZ (réseau publique) : 10.31.184.1/22

Voici les ports de notre switch :

- Le câble branché sur le port 1 est le câble connecté au routeur sur l'interface enp4s0 (LAN)
- Le câble branché sur le port 2 est le câble connecté au serveur sur l'interface vmbr0
- Le câble branché sur le port 23 est le câble connecté au routeur sur l'interface enp5s0 (DMZ)
- Le câble branché sur le port 24 est le câble connecté au serveur sur l'interface vmbr1



Installation du routeur

Installation du système d'exploitation Debian 12 « Bookworm »

Dans un premier temps, nous allons installer sur notre routeur un système d'exploitation. Pour ce projet, nous avons choisi « Bookworm », un système d'exploitation Debian 12.

Nous allons d'abord télécharger l'image des systèmes d'exploitation sur [le site officiel de Debian](#). Nous téléchargeons l'image i386 de la section autres images (amorçage par le réseau, sur clef USB personnalisée, etc.)

Downloading Debian

This is Debian 12, codenamed *bookworm*, netinst, for 64-bit PC (amd64) [debian-12.2.0-amd64-netinst.iso](#).

Figure 3 Téléchargement de l'image Debian 12

Nous allons ensuite utiliser une clé USB comme clé bootable, c'est-à-dire une clé USB qui nous permet d'utiliser le système de fichiers stocké dans la clé pour démarrer l'ordinateur plutôt que d'utiliser du matériel.

Pour ce faire, nous utilisons le logiciel Rufus, qui permet de créer des supports bootable (live USB) sur un périphérique externe comme une clé USB. Nous pouvons le télécharger sur le [site officiel du logiciel](#).



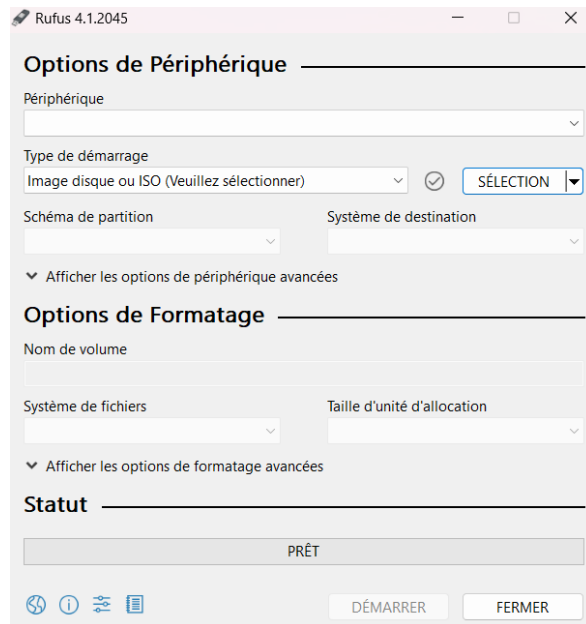


Figure 4 Interface du logiciel Rufus

Nous insérons la clé USB de notre choix dans le périphérique USB de notre ordinateur. Nous sélectionnons ensuite l'image ISO téléchargée au préalable, puis nous démarrons le processus.

ATTENTION : Cette manipulation effacera tout le contenu de la clé USB utilisée. Nous devons nous assurer de sauvegarder les fichiers existants sur un autre périphérique USB

Une fois le périphérique configuré, nous pouvons insérer notre clé USB bootable sur la machine que nous voulons configurer. Pour accéder au menu Boot, nous appuyons sur F11 dès le démarrage de l'ordinateur. Cette touche est susceptible de changer en fonction du modèle d'ordinateur utilisé.



Figure 5 Menu Boot F11



Après avoir sélectionné l'option Boot Device et avoir démarré notre clé USB, nous suivrons pas à pas les directives ci-dessous pour installer nos systèmes d'exploitation :

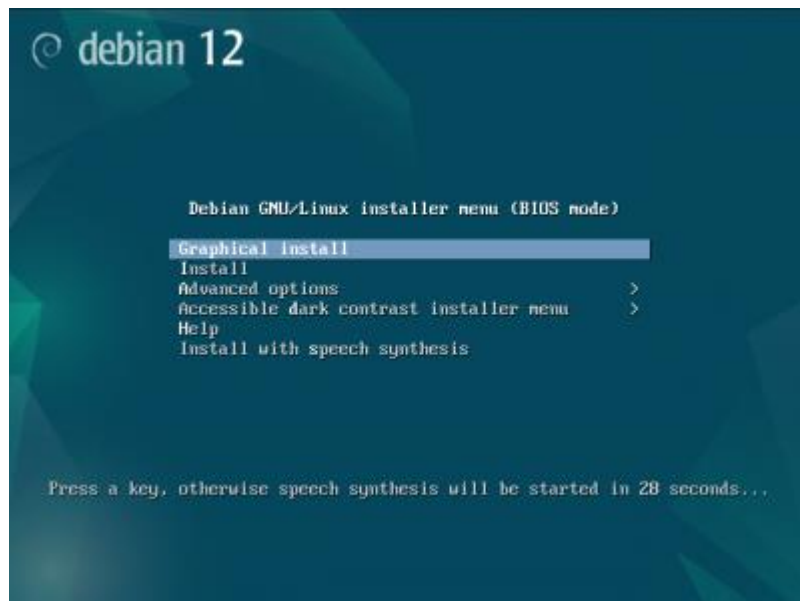


Figure 6 Choix du mode d'installation



Figure 7 Sélection de la langue du système d'exploitation





Figure 8 Sélection du fuseau horaire



Figure 9 Sélection du type de clavier





Figure 10 Chargement et installation des composants nécessaires



Figure 11 Choix du nom de la machine (srv-asie)





Figure 14 Choix de l'identifiant utilisateur



Figure 15 Configuration de l'horloge





Figure 76 Choix de la méthode de partitionnement

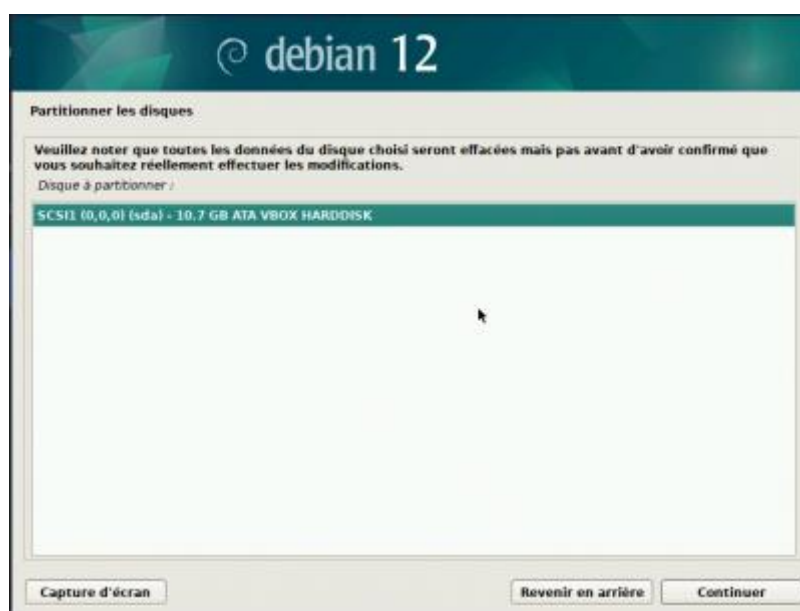


Figure 87 Choix du disque à partitionner





Figure 98 Choix des partitions



Figure 109 Résumé de la table des partitions





Figure 20 Application des changements



Figure 21 Installation du système





Figure 112 Analyse des médias supplémentaires



Figure 123 Choix du pays du miroir de l'archive Debian





Figure 134 Choix du miroir de l'archive Debian



Figure 145 Choix du mandataire HTTP (vide)



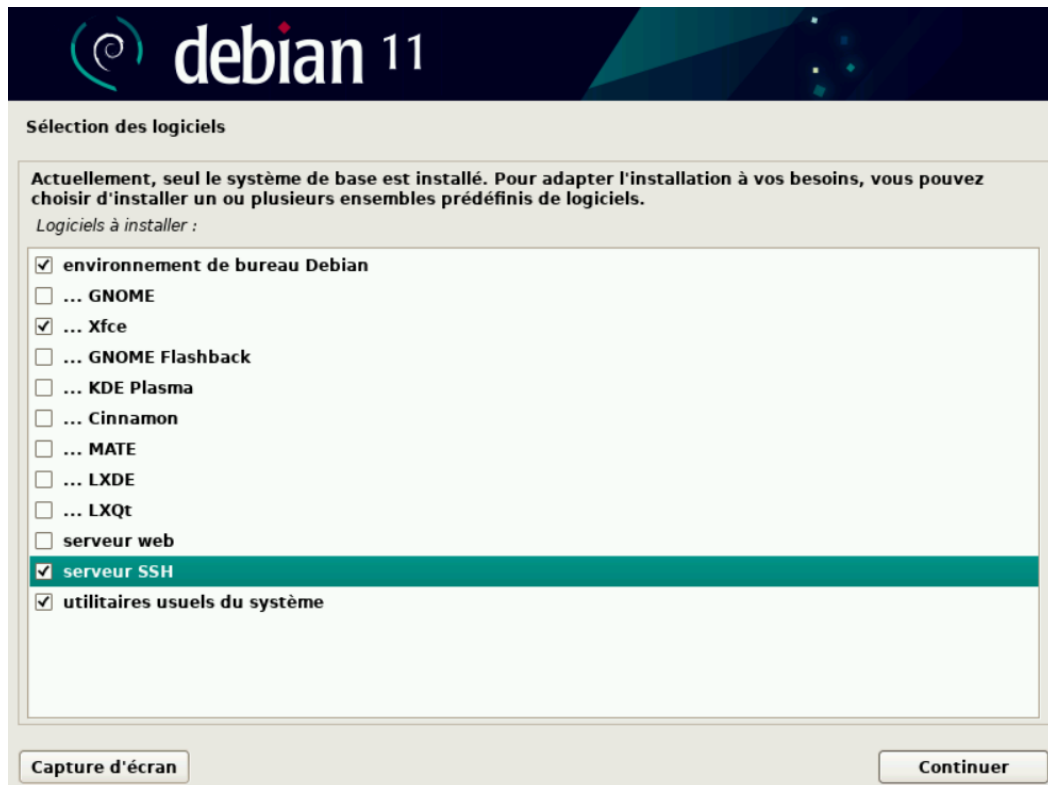


Figure 156 Choisir les logiciels à installer (décocher environnement de bureau Debian, Xfce)

Configuration du routeur

Une fois le routeur installé, nous devons maintenant le configurer. Pour cela, nous commençons par supprimer le paquet téléchargé par défaut « network-manager », puis nous installons les paquets suivants :

```
apt remove --purge network-manager
apt update && apt upgrade
apt install iptables tcpdump net-tools vim nano inetutils-ping sudo less cron wget logrotate
netcat-traditional ntpdate dnsutils rsyslog traceroute nmap rsync
```

Nous configurons maintenant le fichier `/etc/network/interfaces`. Dans ce dernier, nous configurerons nos différentes interfaces avec leurs adresses IP et leurs masques, ainsi que leur passerelle par défaut :

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



```

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# Interface Réseau Routeur Asie <====> Routeur Monde
auto enp2s0
iface enp2s0 inet static
    address 172.31.176.254/16
    gateway 172.31.0.1

# Interface Réseau Routeur Asie <====> Réseau LAN (privé)
auto enp4s0
iface enp4s0 inet static
    address 10.31.179.254/22

# Interface Réseau Routeur Asie <====> Réseau DMZ (public)
auto enp5s0
iface enp5s0 inet static
    address 10.31.187.254/22

```

Dans le fichier rc.local que nous créerons, nous configurerons une règle de pare-feu, une adresse DNS et nous activerons le routage de la machine :

```
nano /etc/rc.local
```

```

#!/bin/bash -e

# Règle de pare-feu disant au routeur de remplacer les adresses sources du réseau par son
adresse IP personnelle
iptables -t nat -A POSTROUTING -s 10.31.176.254/20 -j MASQUERADE

# Activation du routage
echo 1 > /proc/sys/net/ipv4/ip_forward

# Choix du DNS
echo "nameserver 8.8.8.8" > /etc/resolv.conf

exit 0

```



Pour démarrer le service, nous devons utiliser successivement ces commandes :

```
# Droits d'exécution du fichier rc.local
```

```
chmod +x /etc/rc.local
```

```
# Activer le fichier rc-local.service
```

```
systemctl enable rc-local.service
```

```
# Lancer le fichier rc-local.service
```

```
systemctl start rc-local.service
```

```
# Connaître le comportement du fichier rc-local.service (détection d'erreurs)
```

```
systemctl status rc-local.service
```



Installation du serveur

Qu'est-ce que Proxmox VE ?

Proxmox Virtual Environment (VE) est une plateforme de virtualisation open source qui combine deux technologies majeures : la virtualisation de serveurs basée sur conteneurs (LXC) et la virtualisation matérielle complète (KVM - Kernel-based Virtual Machine).

C'est une solution de virtualisation puissante et polyvalente qui combine différentes technologies pour offrir une virtualisation efficace, une gestion simplifiée et des fonctionnalités avancées pour les environnements informatiques virtualisés.

Installation du système d'exploitation Proxmox VE

Pour installer le système d'exploitation sur le routeur, nous commençons par télécharger l'image de la dernière version stable Proxmox sur [le site officiel de Proxmox](https://proxmox.com/en/downloads) :





Proxmox VE 8.0 ISO Installer

	Version	File Size	Last Updated	Download
	8.0-2	1.11 GB	June 22, 2023	
	SHA256SUM	e2b27648a8a91c0da1e8e718882a5ff87a8f054c4dd7e0ea1d8af85125d82812		

Figure 167 Téléchargement de l'ISO de Proxmox VE 8.0



Nous créons ensuite une clé bootable avec cette image en utilisant le logiciel Rufus comme réalisée précédemment. Nous sélectionnons notre clé comme périphérique de démarrage dans le menu boot, puis suivons les étapes des captures d'écran une à une :

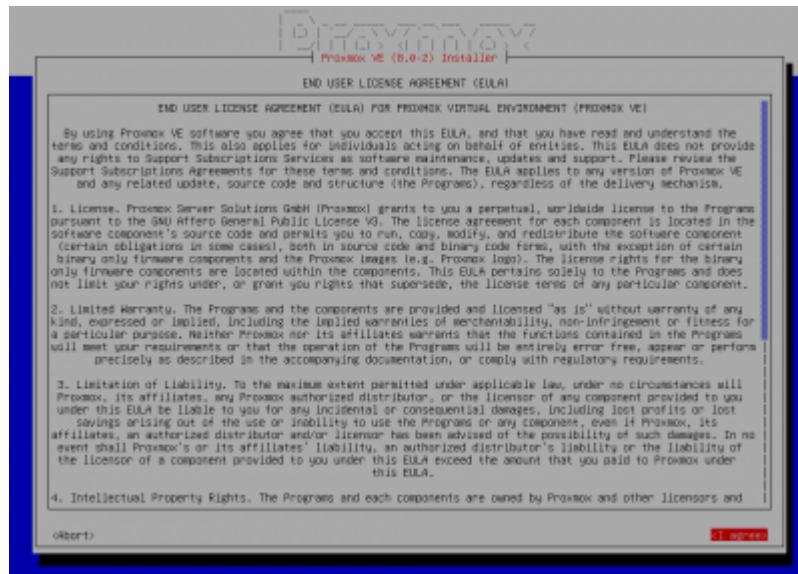


Figure 178 Conditions d'utilisation de Proxmox VE

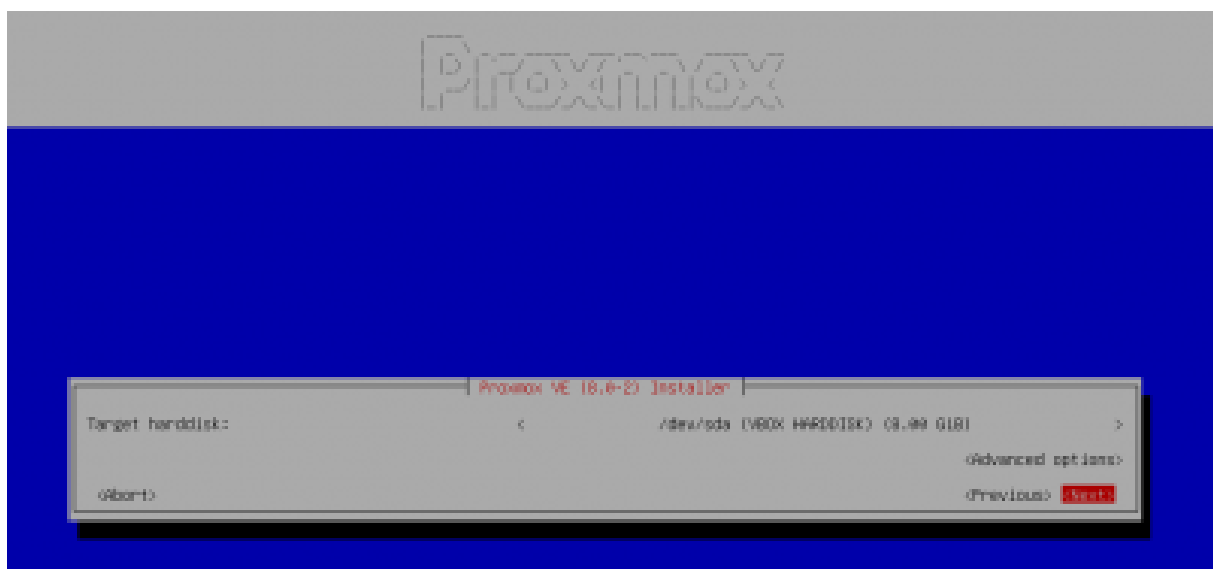


Figure 189 Choix du disque accueillant le système d'exploitation





Figure 30 Choix des informations de langue et de localisation



Figure 31 Choix du mot de passe administrateur et de l'adresse mail



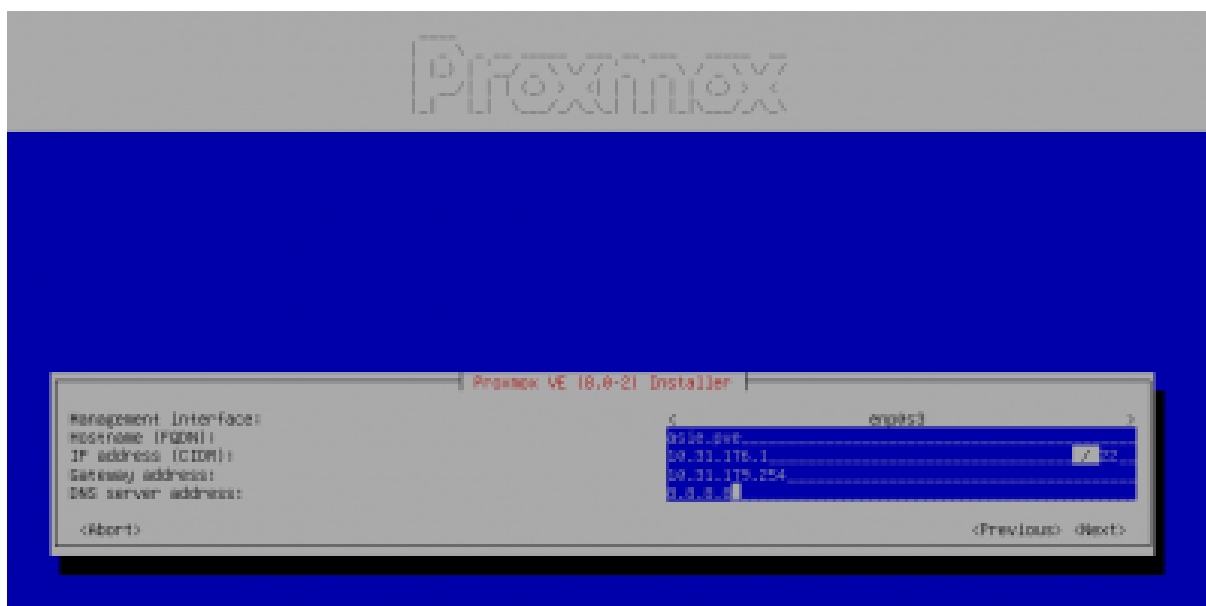


Figure 32 Choix de l'interface, du nom d'hôte, de l'adresse IP, du masque, de la passerelle par défaut et du DNS



Figure 33 Confirmation des informations





Figure 34 Installation de Proxmox VE

Configuration du serveur

Lorsque notre système d'exploitation est installé, nous pouvons accéder à l'interface graphique de Proxmox VE depuis l'adresse <https://10.31.176.1:8006>. Toutes nos configurations se feront à partir de cette interface web. Nous nous connectons avec le compte root :

A screenshot of the Proxmox VE Login form. The form has a title 'Proxmox VE Login'. It contains four input fields: 'User name:', 'Password:', 'Realm:', and 'Language:'. The 'Realm:' field is set to 'Linux PAM standard authentication' and the 'Language:' field is set to 'English - English'. There is a 'Save User name:' checkbox and a 'Login' button at the bottom right.

Figure 35 Interface de connexion de Proxmox VE



Grâce à Proxmox, nous pouvons voir la fiche technique du serveur en cliquant dans le menu à droite sur « PVE », puis « Summary » :

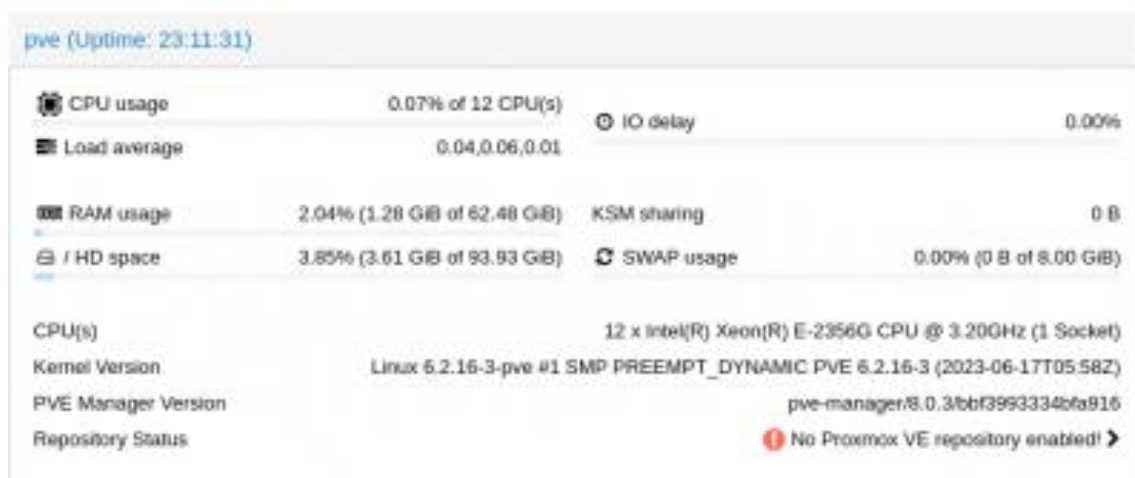


Figure 36 Fiche technique du serveur

Pour attribuer plusieurs interfaces virtuelles à notre carte réseau, nous devons créer des ponts virtuels. Ces derniers permettent de segmenter notre réseau, et donc de pouvoir créer plusieurs interfaces dans un serveur ne possédant qu'une seule carte graphique

Pour configurer les adresses IP de nos interfaces, nous cliquons sur notre serveur, puis dans l'onglet « Network », et enfin sur le bouton « Create » -> Linux Bridge. Nous créons deux ponts :

- vmbr0, l'interface du sous-réseau LAN qui aura pour adresse IP 10.31.176.1/22 et sur laquelle on définira la passerelle par défaut (10.31.179.254)
- vmbr1, l'interface du sous-réseau DMZ qui aura pour adresse IP 10.31.184.1/22

Name	Type	Active	Autostart	VLAN a	Ports/Slaves	Bond Mode	CIDR	Gateway
eno1	Network Device	Yes	No	No				
eno2	Network Device	Yes	No	No				
vmbr0	Linux Bridge	Yes	Yes	No	eno1		10.31.176.1/22	10.31.179.254
vmbr1	Linux Bridge	Yes	Yes	No	eno2		10.31.184.1/22	

Figure 37 Bridges vmbr0 et vmbr1

Nous pouvons vérifier la configuration de notre serveur en ouvrant un shell et en entrant la commande suivante :

```
ifconfig
```



```

vmbr0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.31.176.1 netmask 255.255.252.0 broadcast 0.0.0.0
    inet6 fe80::a94:efff:fea0:5049 prefixlen 64 scopeid 0x20<link>
    ether 08:94:ef:a0:50:49 txqueuelen 1000 (Ethernet)
    RX packets 186644 bytes 67151241 (64.0 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 143938 bytes 103864053 (99.0 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

vmbr1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.31.184.1 netmask 255.255.252.0 broadcast 0.0.0.0
    inet6 fe80::a94:efff:fea0:504a prefixlen 64 scopeid 0x20<link>
    ether 08:94:ef:a0:50:4a txqueuelen 1000 (Ethernet)
    RX packets 227 bytes 19783 (19.3 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 33 bytes 7838 (7.6 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

```

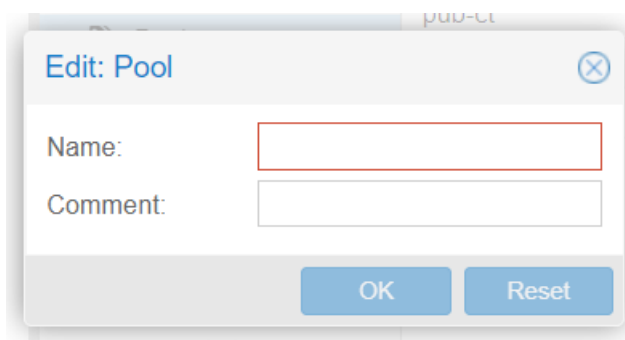
Figure 38 Résultat de la commande ifconfig

Création des pools

Les pools sont des groupes qui contiendront l'ensemble de nos machines virtuelles et de nos conteneurs. Nous en créons 5 comme suit :

- pub-ct : nos conteneurs publics
- priv-ct : nos conteneurs privés
- pub-vm : nos machines virtuelles publiques
- priv-vm : nos machines virtuelles privées
- template : un template pour nos conteneurs et nos machines virtuelles

Dans un premier temps, nous nous plaçons dans la vue des pools en cliquant sur « Server View » et en sélectionnant « Pool View ». Dans le menu des permissions, nous sélectionnons « Pool », puis « Create ». Nous entrons le nom de notre pool puis nous validons.



The image shows a web-based dialog box titled "Edit: Pool". It has a close button in the top right corner. Inside the dialog, there are two text input fields. The first is labeled "Name:" and the second is labeled "Comment:". Below these fields, there are two buttons: "OK" and "Reset". The dialog is overlaid on a background that shows a list of pools, with "pub-ct" visible.



Figure 39 Création de la pool

Une fois toutes les pools créées, nous pouvons les apercevoir dans l'encadré prévu à cet effet :

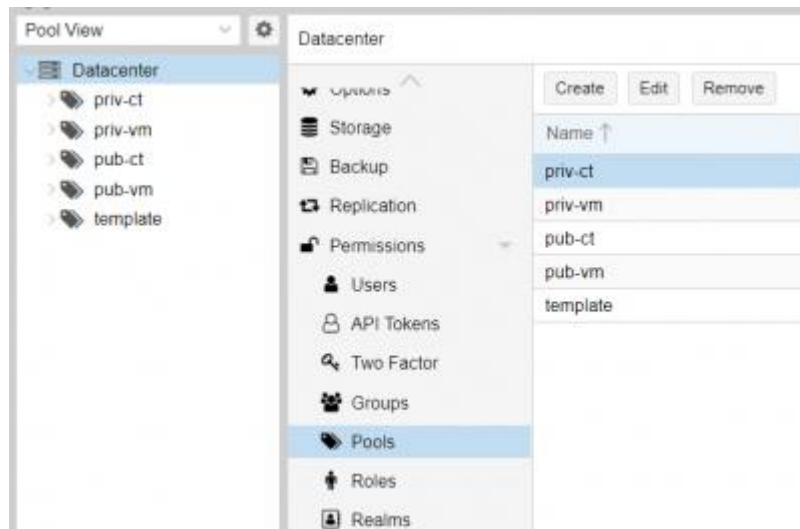


Figure 40 Création des pools terminée

Création du conteneur template

Pour créer un conteneur template, nous allons tout d'abord télécharger l'image Debian 12. Pour ce faire, nous nous replaçons en « Server View », puis sélectionnons « local (pve) ». Dans les menus disponibles, nous sélectionnons « CT Templates », puis nous cliquons sur le bouton « Templates » et sélectionnons la version debian-12-standard.



Figure 41 Téléchargement de l'ISO Debian 12

Nous pouvons maintenant créer notre conteneur en cliquant sur le bouton « Create CT ». Voici les différentes informations que nous devons entrer :



General :

- Node : pve
- Hostname : template-ct
- Resource Pool : template
- Password : password

Template :

- Storage : local
- Template : debian-12-standard

Disks :

- Storage : local-lvm
- Disk size : 5Go

CPU :

- Cores : 1

Memory :

- Memory : 1024MiB
- Swap : 1024MiB

Network :

- Name : eth0
- Bridge : vmbf0
- IPv4 : 10.31.176.2
- Gateway : 10.31.179.254

DNS :

- DNS domain : 8.8.8.8
- DNS servers : 8.8.4.4
-

Create: LXC Container

General

Template

Disks

CPU

Memory

Network

DNS

Confirm

Node:

pve

CT ID:

116

Hostname:

template-ct

Unprivileged container:

☒

Nesting:

☒

Resource Pool:

template

Password:

.....

Confirm password:

.....

SSH public key:

Load SSH Key File



Create: LXC Container



General **Template** Disks CPU Memory Network DNS Confirm

Storage: local

Template: debian-12-standard_12.0-1_amd64

Create: LXC Container



General Template **Disks** CPU Memory Network DNS Confirm

rootfs



Storage: local-lvm

Disk size (GiB): 5

Create: LXC Container



General Template Disks **CPU** Memory Network DNS Confirm

Cores: 1

Create: LXC Container



General Template Disks CPU **Memory** Network DNS Confirm

Memory (MiB): 1024

Swap (MiB): 1024

Create: LXC Container



General Template Disks CPU Memory **Network** DNS Confirm

Name: eth0

MAC address: auto

Bridge: vmbr0

VLAN Tag: no VLAN

Firewall: ☒

IPv4: ☒ Static ☐ DHCP

IPv4/CIDR: 10.31.176.2/22

Gateway (IPv4): 10.31.179.254

IPv6: ☒ Static ☐ DHCP ☐ SLAAC

IPv6/CIDR: None

Gateway (IPv6):



Create: LXC Container

General
Template
Disks
CPU
Memory
Network
DNS
Confirm

DNS domain: 8.8.8.8

DNS servers: 8.8.4.4

Create: LXC Container

General
Template
Disks
CPU
Memory
Network
DNS
Confirm

Key ↑	Value
cores	1
features	nesting=1
hostname	template-ct
memory	1024
nameserver	8.8.4.4
net0	name=eth0,bridge=vmbr0,firewall=1,ip=10.31.176.2/22,gw=10.31.179.254
nodename	pve
ostemplate	local:vztmpl/debian-12-standard_12.0-1_amd64.tar.zst
pool	template
rootfs	local-lvm:5
searchdomain	8.8.8.8
swap	1024
unprivileged	1

☐ Start after created

Advanced ☐
Back
Finish

(insérer screens de cette config là)

Création de la machine virtuelle template

Pour créer une machine virtuelle template, nous téléchargeons dans un premier temps l'image Debian. Pour ce faire, nous cliquons sur notre serveur PVE, puis sélectionnons « local (pve) ». Dans les menus disponibles, nous sélectionnons « ISO Image », puis nous cliquons sur le bouton « Upload »



Descriptif des paquets

Voici un descriptif de chaque paquet que nous avons pu installer au cours de nos configurations :

- iptables : iptables permet d'écrire des règles de pare-feu.
- tcpdump : tcpdump permet de capturer les trames d'une interface réseau.
- net-tools : net-tools contient un ensemble d'outils importants pour contrôler le sous-système réseau du noyau Linux.
- vim : vim est un éditeur de texte en ligne de commande, qui offre beaucoup plus de fonctionnalités que l'éditeur nano.
- nano : nano est un éditeur simple de texte en ligne de commande.
- inetutils-ping : inetutils-ping permet de vérifier la communication d'une machine avec la commande ping.
- sudo : sudo permet d'exécuter des commandes avec les droits superutilisateurs.
- less : less est utilisé pour afficher des longs textes grâce au fait de pouvoir revenir sur la page précédente.
- cron : cron est un programme permettant de planifier l'exécution de certaines commandes ou scripts.
- wget : wget permet de télécharger des fichiers via http, https et ftp en ligne de commande.
- logrotate : logrotate permet de limiter la taille des fichiers journaux présents dans /var/log.
- netcat-traditional : netcat est un outil permettant d'ouvrir des connexions TCP et UDP.
- ntpdate : ntpdate permet de synchroniser l'heure de la machine à chaque démarrage.
- dnsutils : dnsutils permet de tester et dépanner des problèmes de DNS. Il inclut par exemple la commande dig qui permet d'interroger des serveurs DNS.
- rsyslog : rsyslog permet de centraliser des logs.
- traceroute : traceroute est une commande permettant de suivre les chemins de paquets de données.
- nmap : nmap est principalement un scanneur de ports, mais peut également détecter l'OS d'une machine, et exécuter des scripts conçus pour tester la sécurité d'un service.
- rsync : rsync permet de copier et de synchroniser des données entre des répertoires locaux ou distants de manière efficace en minimisant les transferts de données.



Procédure de résolution de problèmes

Si le serveur / routeur ne peut pas accéder à Internet :

- Etape 1 : ping l'interface du routeur reliée à notre serveur

1. si ok : passer à l'étape suivante
2. si erreur : vérifier la configuration du routeur et du serveur, vérifier les câbles

- Etape 2 : ping l'interface du routeur relié au mur (routeur professeur)

1. si ok : passer à l'étape suivante
2. si erreur : vérifier la configuration du routeur et de la passerelle par défaut du serveur, vérifier les câbles

- Etape 3 : ping le routeur professeur

1. si ok : passer à l'étape suivante
2. si erreur : vérifier la configuration du routeur et sa passerelle par défaut, vérifier les câbles

- Etape 4 : ping 8.8.8.8

1. si ok : félicitations ! problème résolu !
2. si erreur : vérifier le routeur professeur

