

Aksel

CAUBEL

RT3-App Dev-Cloud

Installation Proxmox

Utilisation de l'Idrac

Creds

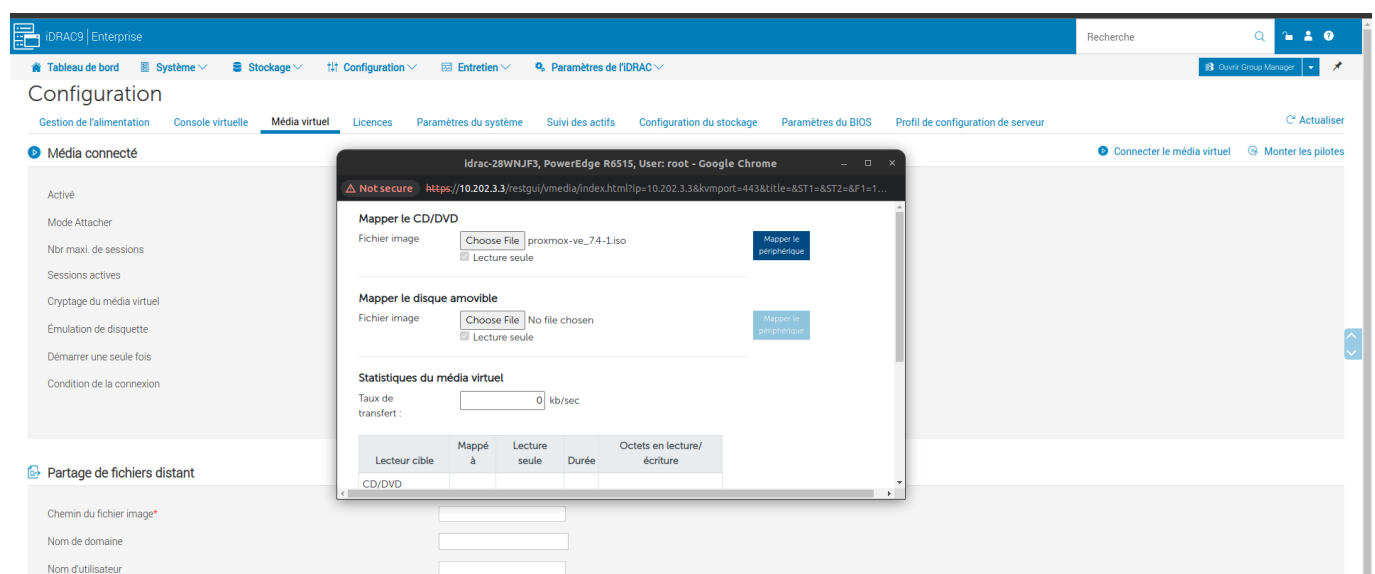
```
user = root  
mdp = root  
ip = 10.202.3.3
```

```
ip Proxmox : 10.202.3.33  
identifiant Proxmox : root  
mot de pass Proxmox : rootroot
```

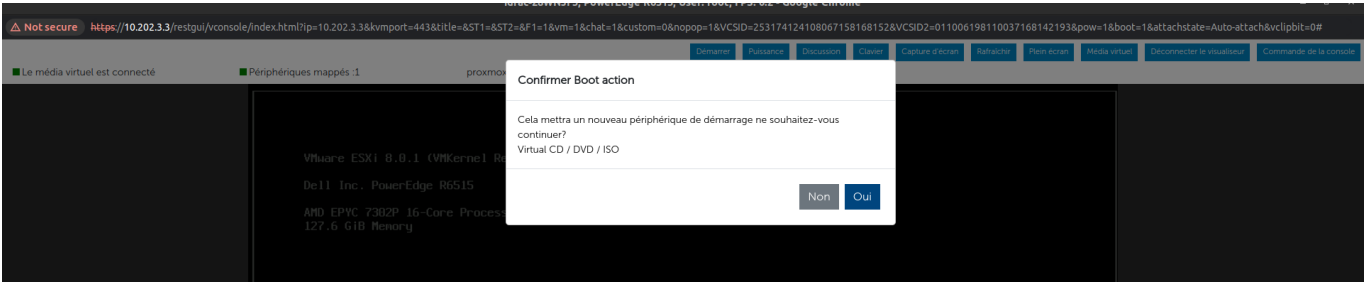
On va venir faire une installation via l'interface Idrac.

Pour ce faire on va entrer dans la partie **configuration** -> **Média Virtuel**. Le but est de faire un mapping de notre OS Proxmox *Utilisation de la version 7.4*

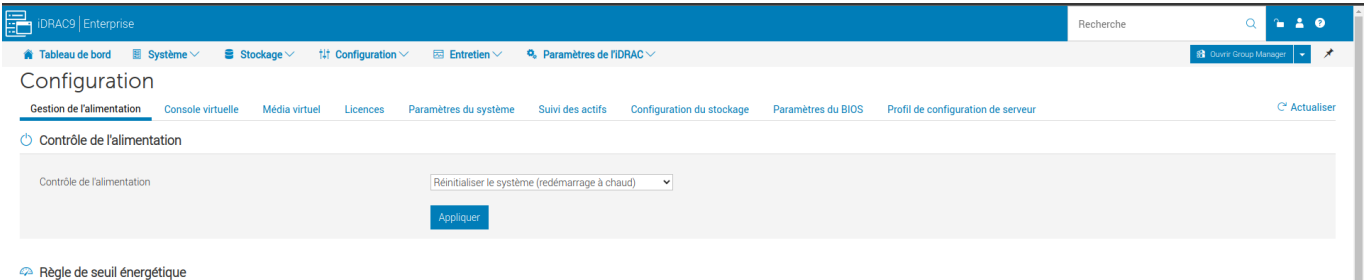
On vient ensuite **Connecter le média virtuel**



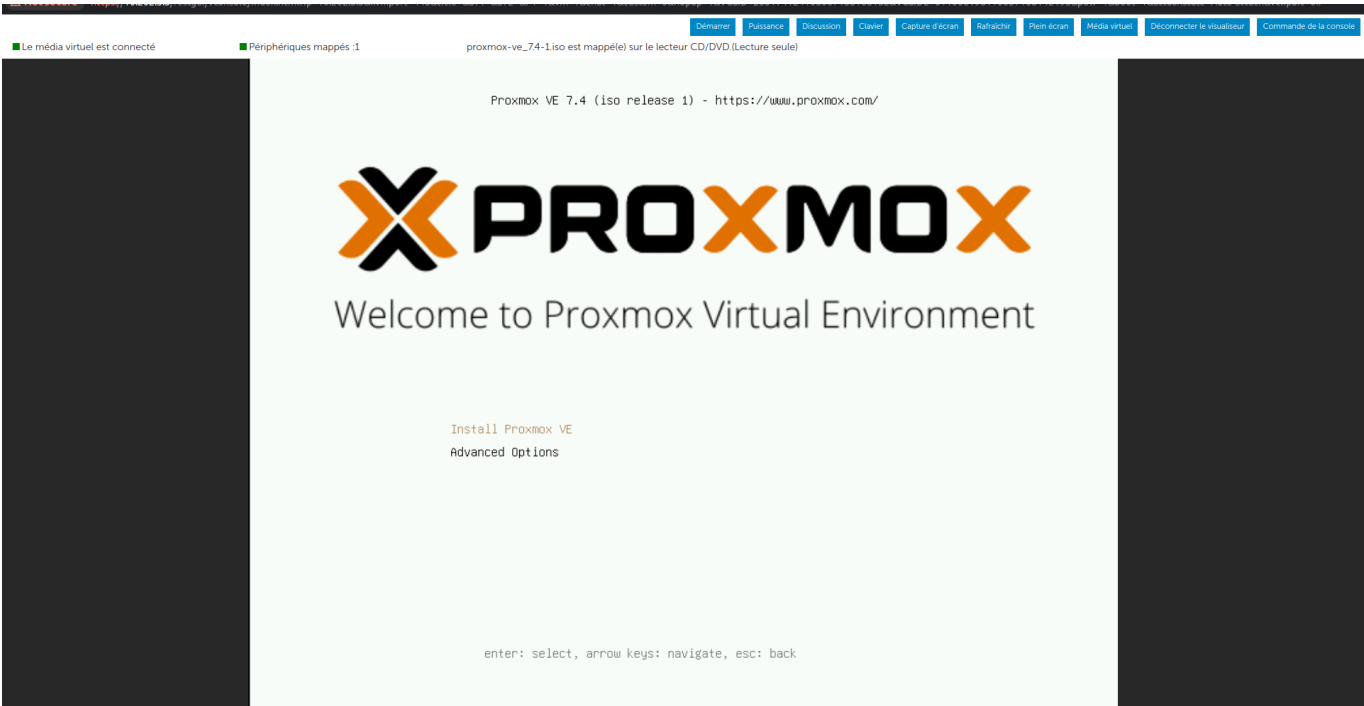
Une fois l'iso connecté on va choisir proxmox, on vient dans la console virtuelle dans **démarrer**->**Boot action**->**CD/DVD/ISO** pour qu'au prochain démarrage l'on puisse réaliser l'installation de **proxmox**.



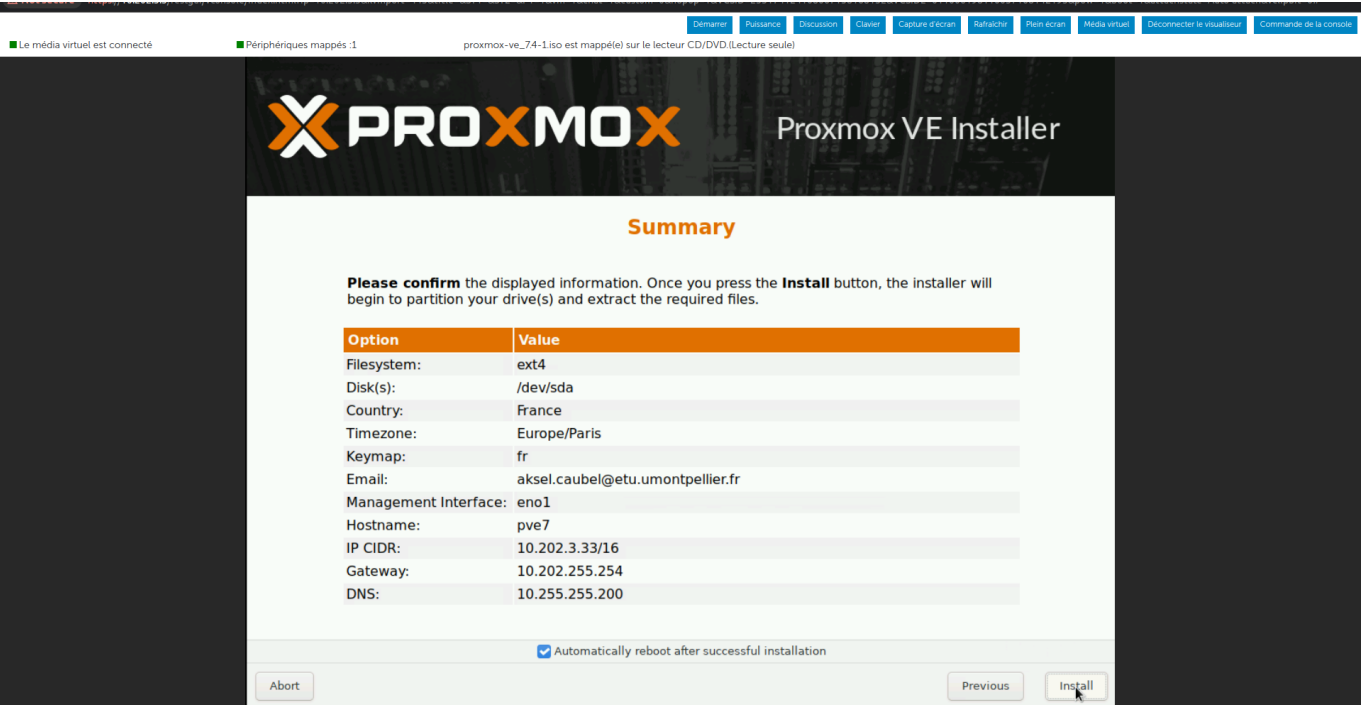
Pour faire le redémarrage a chaud a distance, on va revenir sur l'interface *Idrac* dans **configuration** > **Gestion de l'alimentation** et ensuite dans la partie *Contrôle de l'alimentation* choisir l'option **Réinitialiser le système (redémarrage à chaud)**



Maintenant nous pouvons commencer a suivre les instructions de Proxmox :



Une fois les instructions suivit on retrouve cette configuration dans notre cas.



L'interface graphique est maintenant disponible sur le port [8006](#)

Mise en place de GOAD sur Proxmox

Mise en place de l'architecture

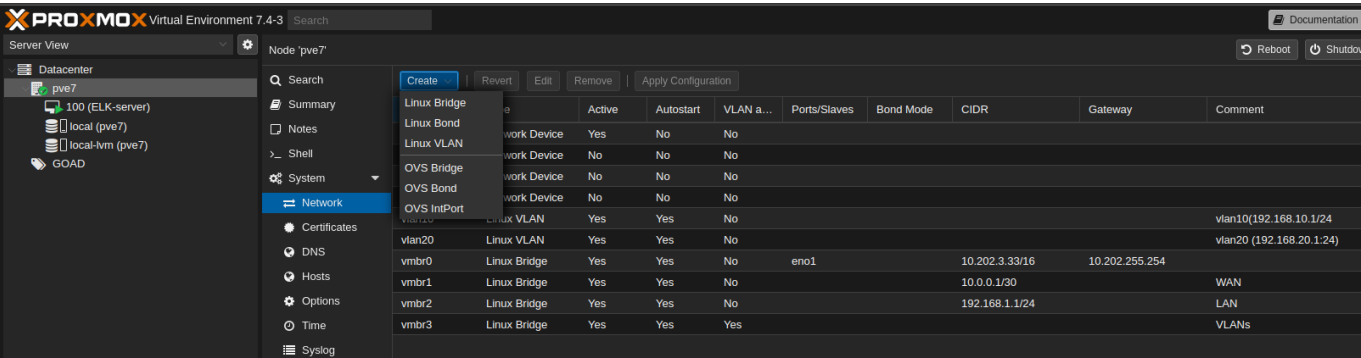
Source d'instruction

La configuration initial donner nous demande crée des interfaces réseaux supplémentaire :

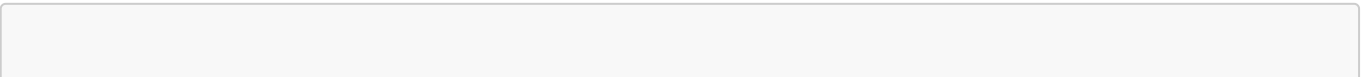
- 3 Bridge Linux
- 2 VLAN Linux

Pour ce faire, dans la partie **Datacenter** (volet de gauche) on va aller dans notre **Node** ici appelé **pve7** puis aller dans l'onglet **Système** -> **Network**.

Pour la création des bridges / VLANs, tous va se faire dans l'onglet **Create** :



Voici un extrait des prérequis :



The network we will build will be in multiple part :

- 10.0.0.0/30 (10.0.0.1-10.0.0.2) : this will be the WAN network with only 2 ips, one for proxmox host, and the other one for pfsense
- 192.168.1.1/24 (192.168.1.1-192.168.1.254) : this will be the LAN network for the pfsense and the provisioning machine
- 192.168.10.1/24 (192.168.10.1-192.168.10.254) : VLAN1 for the GOAD's vm
- 192.168.20.1/24 (192.168.20.1-192.168.20.254) : VLAN2 for future projects
- 10.10.10.0/24 (10.10.10.0-10.10.10.254) : openvpn for vpn users (will be manage by pfsense later)

Création d'un Bridge :

Edit: Linux Bridge

Name: vmbr3 Autostart: ☒

IPv4/CIDR: VLAN aware: ☒

Gateway (IPv4): Bridge ports:

IPv6/CIDR: Comment: VLANs

Gateway (IPv6):

Advanced ☐ OK Reset

Création d'un VLAN :

Create: Linux VLAN

Name: vlan10 Autostart: ☒

IPv4/CIDR: Vlan raw device: vmbr3

Gateway (IPv4): VLAN Tag: 10

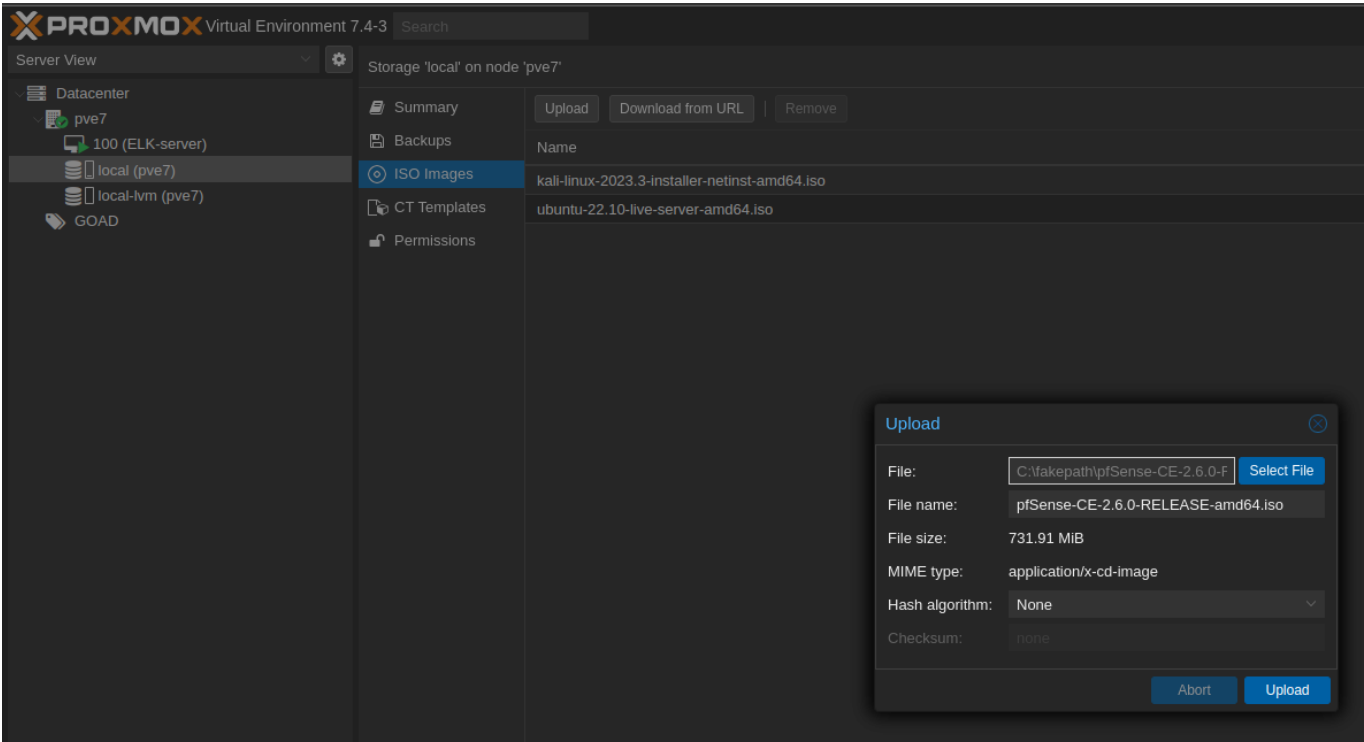
IPv6/CIDR: Comment: vlan10 (192.168.10.1/24)

Gateway (IPv6):

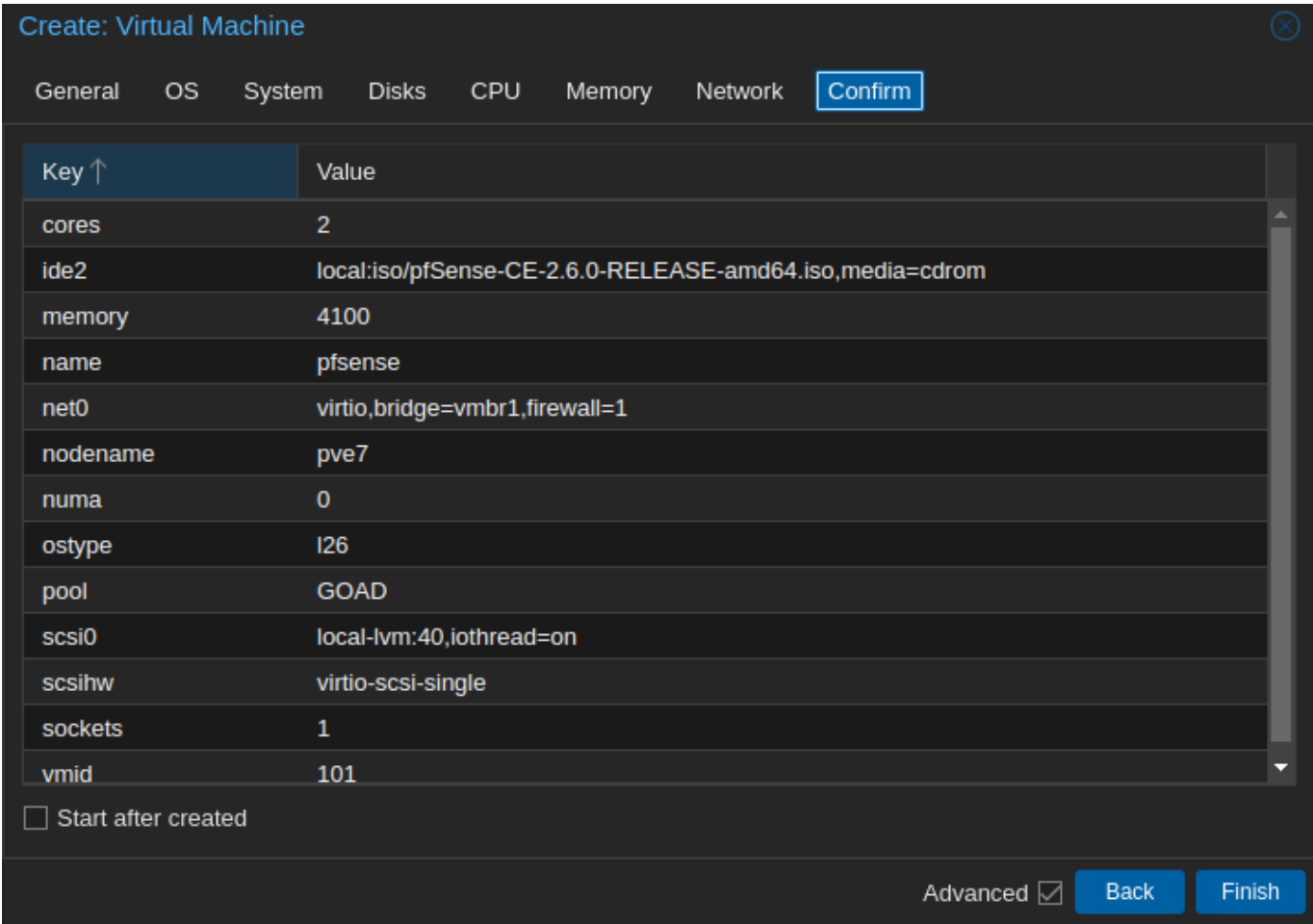
Either add the VLAN number to an existing interface name, or choose your own name and set the VLAN raw device (for the latter ifupdown1 supports vlanXY naming only)

? Help Advanced ☐ Create

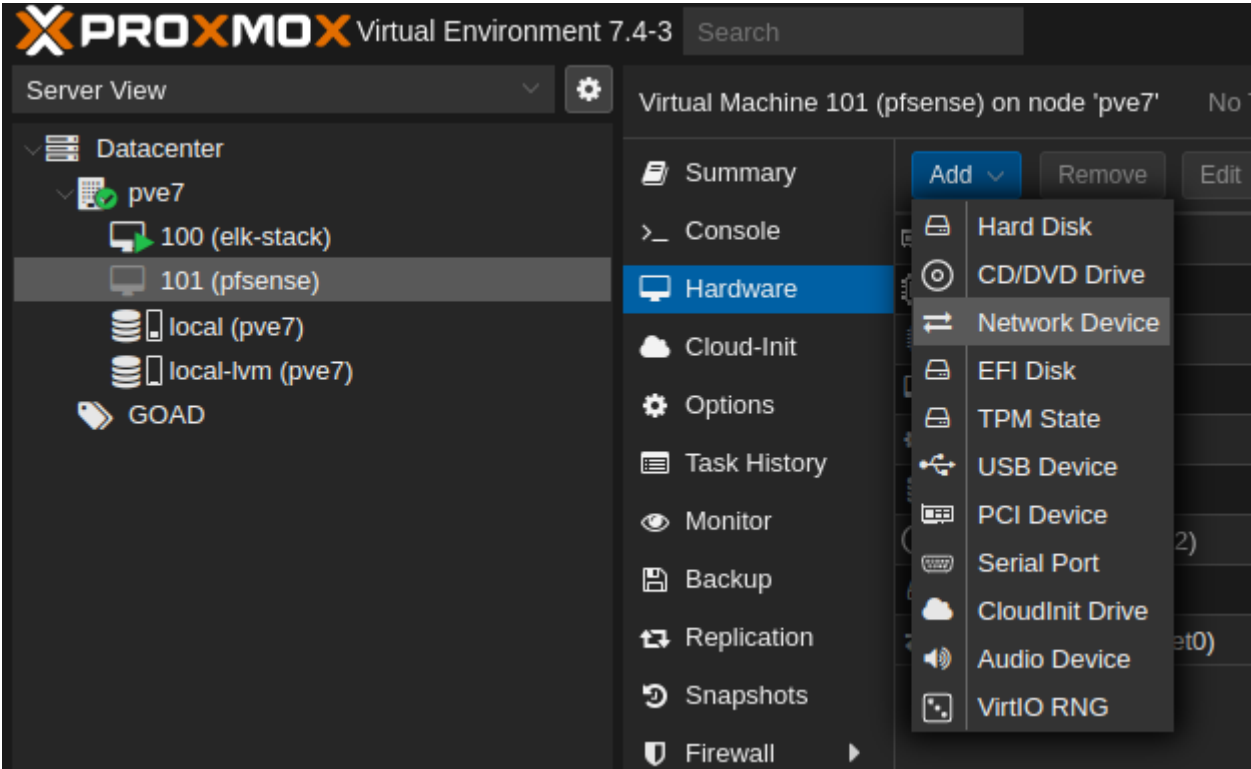
Par la suite il nous est demandé de faire l'installation d'une ISO PFSense. On va pouvoir procéder ainsi :



On va ensuite pouvoir crée notre première VM en commençant par *PfSense* **Ne pas démarrer la VM a sa création:**



Une fois que la VM est crée avec la configuration ci-dessus, on va venir lui rajouter des interfaces réseaux que nous avons précédement crée de cette manière :



Le résultat attendu est d'avoir :

| | |
|-----------------------|--|
| Network Device (net0) | virtio=EE:C9:AF:42:CC:C9,bridge=vibr1,firewall=1 |
| Network Device (net1) | virtio=26:9D:C0:46:60:E6,bridge=vibr2,firewall=1 |
| Network Device (net2) | virtio=A2:2E:3C:27:A5:D4,bridge=vibr3,firewall=1 |

Maintenant que *PfSense* est configuré on peut démarrer la machine.

Entrez dans la console depuis *Proxmox*

Suivez le guide d'installation jusqu'à l'option **reboot**

Configuration réseau

VLAN(s)

On ne souhaite pas configurer de VLAN :

```
Do VLANs need to be set up first?
If VLANs will not be used, or only for optional interfaces, it is typical to
say no here and use the webConfigurator to configure VLANs later, if required.
Should VLANs be set up now [y/n]? n
```

Interfaces

Précédemment nous avons attribuée les **devices** réseaux vtnet{1,2,3}. **Attention, dans PfSense le compteur est revenue a partir de 0. Nous aurons alors vtnet1 -> vtnet0 et ainsi de suite.**

```
Enter the WAN interface name or 'a' for auto-detection
(vtnet0 vtnet1 vtnet2 or a): vtnet0

Enter the LAN interface name or 'a' for auto-detection
NOTE: this enables full Firewalling/NAT mode.
(vtnet1 vtnet2 a or nothing if finished): vtnet1

Enter the Optional 1 interface name or 'a' for auto-detection
(vtnet2 a or nothing if finished): vtnet2

The interfaces will be assigned as follows:

WAN   -> vtnet0
LAN   -> vtnet1
OPT1  -> vtnet2

Do you want to proceed [y/n]? y
```

Les choix fait précédemment nous menerons a la configuration suivante :

```
*** Welcome to pfSense 2.6.0-RELEASE (amd64) on pfSense ***

WAN (wan)      -> vtnet0      ->
LAN (lan)      -> vtnet1      -> v4: 192.168.1.1/24
OPT1 (opt1)    -> vtnet2      ->

0) Logout (SSH only)          9) pfTop
1) Assign Interfaces          10) Filter Logs
2) Set interface(s) IP address 11) Restart webConfigurator
3) Reset webConfigurator password 12) PHP shell + pfSense tools
4) Reset to factory defaults   13) Update from console
5) Reboot system              14) Enable Secure Shell (sshd)
6) Halt system                15) Restore recent configuration
7) Ping host                  16) Restart PHP-FPM
8) Shell

Enter an option: 2
```

Configuration Réseau

```

Available interfaces:

1 - WAN (vtnet0 - dhcp, dhcp6)
2 - LAN (vtnet1 - static)
3 - OPT1 (vtnet2)

Enter the number of the interface you wish to configure: 1

Configure IPv4 address WAN interface via DHCP? (y/n) n

Enter the new WAN IPv4 address. Press <ENTER> for none:
> 10.0.0.2

Subnet masks are entered as bit counts (as in CIDR notation) in pfSense.
e.g. 255.255.255.0 = 24
     255.255.0.0   = 16
     255.0.0.0     = 8

Enter the new WAN IPv4 subnet bit count (1 to 32):
> 30

For a WAN, enter the new WAN IPv4 upstream gateway address.
For a LAN, press <ENTER> for none:
> 10.0.0.1

Configure IPv6 address WAN interface via DHCP6? (y/n) n

Enter the new WAN IPv6 address. Press <ENTER> for none:
>
Disabling IPv4 DHCPD...
Disabling IPv6 DHCPD...

Do you want to revert to HTTP as the webConfigurator protocol? (y/n) y

Please wait while the changes are saved to WAN...
  Reloading filter...
  Reloading routing configuration...
  DHCPD...
  Restarting webConfigurator...

The IPv4 WAN address has been set to 10.0.0.2/30

Press <ENTER> to continue.

```

Nous aurons alors le résultat de configuration suivant :

```

WAN (wan)      -> vtnet0      -> v4: 10.0.0.2/30
LAN (lan)      -> vtnet1      -> v4: 192.168.1.1/24
OPT1 (opt1)    -> vtnet2      ->

```

Une fois la configuration générique faite, on va venir faire une configuration plus précise pour l'interface LAN en faisant :

- Un changement d'adresse IP -> 192.168.1.2/24
 - Sans mettre de passerelle
 - Pas d'IPv6
- Un serveur DHCP (pool : 192.168.1.100 <-> 192.168.1.254)


```

0) Logout (SSH only)          9) pfTop
1) Assign Interfaces          10) Filter Logs
2) Set interface(s) IP address 11) Restart webConfigurator
3) Reset webConfigurator password 12) PHP shell + pfSense tools
4) Reset to factory defaults  13) Update from console
5) Reboot system              14) Enable Secure Shell (sshd)
6) Halt system                15) Restore recent configuration
7) Ping host                  16) Restart PHP-FPM
8) Shell

```

Enter an option: 2

Available interfaces:

```

1 - WAN (vtnet0 - static)
2 - LAN (vtnet1 - static)
3 - OPT1 (vtnet2)

```

Enter the number of the interface you wish to configure: 2

Enter the new LAN IPv4 address. Press <ENTER> for none:

> 192.168.1.2

Subnet masks are entered as bit counts (as in CIDR notation) in pfSense.

```

e.g. 255.255.255.0 = 24
      255.255.0.0   = 16
      255.0.0.0     = 8

```

Enter the new LAN IPv4 subnet bit count (1 to 32):

> 24

For a WAN, enter the new LAN IPv4 upstream gateway address.

For a LAN, press <ENTER> for none:

>

Enter the new LAN IPv6 address. Press <ENTER> for none:

>

Do you want to enable the DHCP server on LAN? (y/n) y

Enter the start address of the IPv4 client address range: 192.168.1.100

Enter the end address of the IPv4 client address range: 192.168.1.254

The IPv4 LAN address has been set to 192.168.1.2/24

You can now access the webConfigurator by opening the following URL in your web browser:

<http://192.168.1.2/>

Press <ENTER> to continue.

KUM Guest - Netgate Device ID: e39fa5b38524b178733c

*** Welcome to pfSense 2.6.0-RELEASE (amd64) on pfSense ***

```

WAN (wan)      -> vtnet0      -> v4: 10.0.0.2/30
LAN (lan)      -> vtnet1      -> v4: 192.168.1.2/24
OPT1 (opt1)    -> vtnet2      ->

```

Configuration suite en GUI

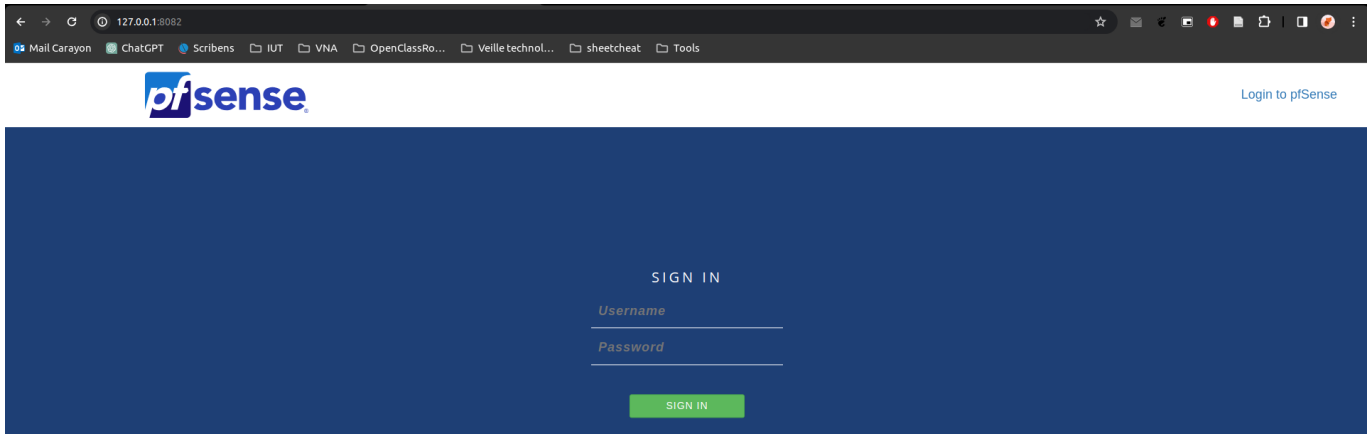
Afin d'avoir accès à l'interface graphique sur notre poste nous devons faire un *port-forwarding* de l'host 192.168.1.2:80 vers notre machine avec un port client quelconque (*ici le 8082*)

Pour ce faire on viens faire un **ssh-L**

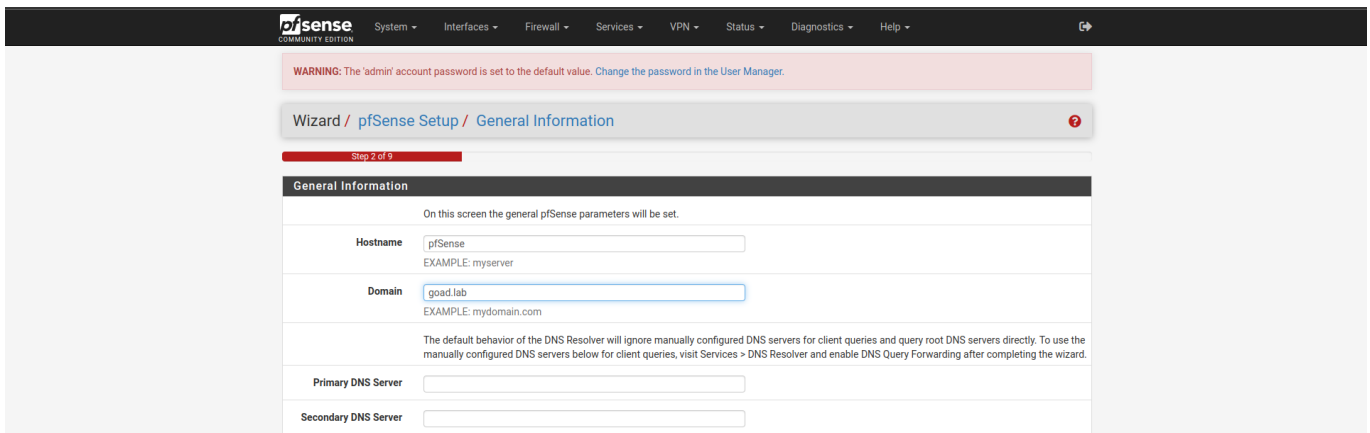
```
ssh-L 8082:192.168.1.2:80 root@10.202.3.33 #Ip proxmox
```

Interface WEB

User: admin / passwd : pfsense



Après connexion appuyer sur **Next** deux fois pour arriver sur cette page :

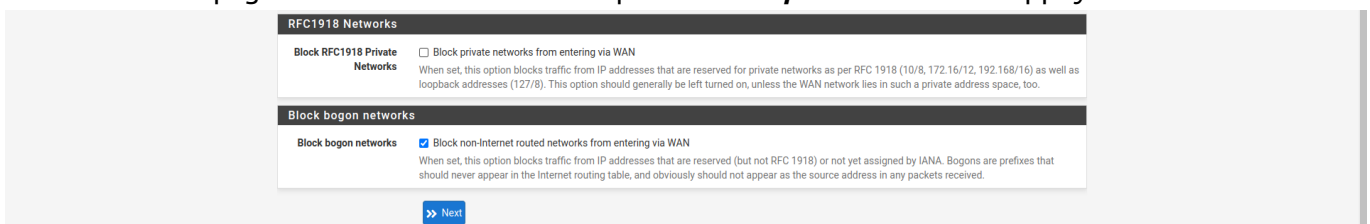


Changer le Domain présent pour **goad . lab**

Pour la configuration **NTP** vous pouvez le laisser par défaut et ensuite entrez **NEXT**.

L'interface WAN **doit être** laissée par défaut.

Sur cette même page vous devais enlever le bloque **RFC1918 private network**. Appuyer sur **NEXT**.




Laissez l'interface LAN comme il vous est affichée. **NEXT**

Changez le mot de passe admin (ici on a choisit la sécurité 😊 => passwd = admin)

Dans l'onglet **System/Advanced/Networking** en bas de page dans la partie **Network Interfaces** on va venir cocher la première case **Hardware Checksum Offloading**

| Network Interfaces | |
|---|--|
| Hardware Checksum Offloading | <input checked="" type="checkbox"/> Disable hardware checksum offload Checking this option will disable hardware checksum offloading. Checksum offloading is broken in some hardware, particularly some Realtek and some specific NICs. This will take effect after a machine reboot or re-configuration. |
| Hardware TCP Segmentation Offloading | <input checked="" type="checkbox"/> Disable hardware TCP segmentation offload Checking this option will disable hardware TCP segmentation offloading (TSO) which may impact performance with some specific NICs. This will take effect after a machine reboot. |
| Hardware Large Receive Offloading | <input checked="" type="checkbox"/> Disable hardware large receive offload Checking this option will disable hardware large receive offloading (LRO). This will take effect after a machine reboot. |
| hn ALTQ support | <input checked="" type="checkbox"/> Enable the ALTQ support for hn NICs. Checking this option will enable the ALTQ support for hn NICs. The ALTQ support will handle traffic. This will take effect after a machine reboot. |
| ARP Handling | <input type="checkbox"/> Suppress ARP messages This option will suppress ARP log messages when multiple interfaces reside on the same network. |
| Reset All States | <input type="checkbox"/> Reset all states if WAN IP Address changes This option resets all states when a WAN IP Address changes instead of only the states related to the changed IP address. |

 Save

Lors de la sauvegarde de configuration, acceptez le **Reboot**

SetUP Fire-Wall PFSense

On vient ajouter une règle pour accepter le trafic **HTTP***(80)***** :

Source

Source

☐ Invert match

Single host or alias

10.0.0.1

Display Advanced

The **Source Port Range** for a connection is typically random and almost never equal to the destination port. In most cases this setting must remain at its default value, any.

Destination

Destination

☐ Invert match

LAN address

Destination Address

Destination Port Range

HTTP (80)

From

Custom

HTTP (80)

To

Custom

Specify the destination port or port range for this rule. The "To" field may be left empty if only filtering a single port.

Et l'on vient bloquer en dernier tous le reste du traffic.

Floating

WAN

LAN

Rules (Drag to Change Order)

| | States | Protocol | Source | Port | Destination | Port | Gateway | Queue | Schedule | Description | Actions |
|--------------------------|-------------|----------|----------------------------------|------|-------------|-----------|---------|-------|----------|----------------------|---------|
| <input type="checkbox"/> | ✗ 0 / 7 KiB | * | Reserved Not assigned by IANA | * | * | * | * | * | | Block bogon networks | |
| <input type="checkbox"/> | ✓ 0 / 0 B | IPv4 TCP | 10.0.0.1 | * | LAN address | 80 (HTTP) | * | none | | | |
| <input type="checkbox"/> | ✗ 0 / 0 B | IPv4 TCP | * | * | * | * | * | none | | | |

↑ Add

↓ Add

🗑 Delete

💾 Save

+ Separator

SetUP IpTables

Sur notre connexion **SSH** précédemment crée (*cette pour le port-forwarding*), on va venir en tant que user root faire :

```
# activate ipforward
echo 1 | sudo tee /proc/sys/net/ipv4/ip_forward
# allow icmp to avoid ovh monitoring reboot the host
iptables -t nat -A PREROUTING -i vmbr0 -p icmp -j ACCEPT
# allow ssh
iptables -t nat -A PREROUTING -i vmbr0 -p tcp --dport 22 -j ACCEPT
# allow proxmox web
iptables -t nat -A PREROUTING -i vmbr0 -p tcp --dport 8006 -j ACCEPT
# redirect all to pfsense
iptables -t nat -A PREROUTING -i vmbr0 -j DNAT --to 10.0.0.2
# add SNAT WAN -> public ip
iptables -t nat -A POSTROUTING -o vmbr0 -j SNAT -s 10.0.0.0/30 --to-source MYPUBLICIP_HERE
```

On va également crée une sauvegarde des règles (**Sachant qu'IpTables perd sa configuration a chaque restart**):

```
iptables-save | sudo tee /etc/network/save-iptables
```

Pour que la configuration se mette a jour dès que la machine démarre, on va venir mettre la configuration suivante a la fin du fichier `/etc/network/interfaces`

```
post-up iptables-restore < /etc/network/save-iptables
```

Setup VLAN(s)

Dans l'onglet `Interfaces/Interface Assignments/VLANs` on vient ajouter un VLAN et mettre la configuration suivant :

Interfaces / VLANs / Edit

VLAN Configuration

Parent Interface

vtnet2 (a2:2e:3c:27:a5:d4) - opt1

Only VLAN capable interfaces will be shown.

VLAN Tag

10

802.1Q VLAN tag (between 1 and 4094).

VLAN Priority

0

802.1Q VLAN Priority (between 0 and 7).

Description

VLAN10

A group description may be entered here for administrative reference (not parsed).





Save



On fait pareil pour le VLAN 20 pour obtenir cette configuration final :

Interfaces / VLANs


Interface AssignmentsInterface GroupsWirelessVLANsQinQsPPPsGREsGIFsBridgesLAGGs

VLAN Interfaces

| Interface | VLAN tag | Priority | Description | Actions |
|---------------|----------|----------|-------------|---|
| vtnet2 (opt1) | 10 | | VLAN10 |   |
| vtnet2 (opt1) | 20 | | VLAN20 |   |

  Add

Une fois les VLANs créés, on va leur assigner une adresse IP. Pour cela on vient dans l'onglet Interface Assignments, on y rajoute le VLAN10 et le VLAN20 :

Interfaces / Interface Assignments 

Interface Assignments

Interface Groups

Wireless

VLANs

QinQs








PPPs


GREs

GIFs



Bridges

LAGGs

| Interface | Network port |
|--------------------------|---|
| WAN | vtnet0 (ee:c9:af:42:cc:c9)  |
| LAN | vtnet1 (26:9d:c0:46:60:e6)   Delete |
| OPT1 | vtnet2 (a2:2e:3c:27:a5:d4)   Delete |
| Available network ports: | VLAN 10 on vtnet2 - opt1 (VLAN10)   Add |

 Save


Et ensuite les configurer en cliquant sur leur nom d'interface :


Interfaces / OPT2 (vtnet2.10)  

General Configuration

| | |
|-------------------------|--|
| Enable | <input checked="" type="checkbox"/> Enable interface |
| Description | <input type="text" value="VLAN10"/> <small>Enter a description (name) for the interface here.</small> |
| IPv4 Configuration Type | <input type="text" value="Static IPv4"/> |
| IPv6 Configuration Type | <input type="text" value="None"/> |
| MAC Address | <input type="text" value="xx:xx:xx:xx:xx:xx"/> <small>The MAC address of a VLAN interface must be set on its parent interface</small> |
| MTU | <input type="text"/> <small>If this field is blank, the adapter's default MTU will be used. This is typically 1500 bytes but can vary in some circumstances.</small> |
| MSS | <input type="text"/> <small>If a value is entered in this field, then MSS clamping for TCP connections to the value entered above minus 40 for IPv4 (TCP/IPv4 header size) and minus 60 for IPv6 (TCP/IPv6 header size) will be in effect.</small> |
| Speed and Duplex | <input type="text" value="Default (no preference, typically autoselect)"/> <small>Explicitly set speed and duplex mode for this interface. WARNING: MUST be set to autoselect (automatically negotiate speed) unless the port this interface connects to has its speed and duplex forced.</small> |


Static IPv4 Configuration

| | |
|-----------------------|---|
| IPv4 Address | <input type="text" value="192.168.10.1"/> / <input type="text" value="24"/> |
| IPv4 Upstream gateway | <input type="text" value="None"/>  Add a new gateway <small>If this interface is an Internet connection, select an existing Gateway from the list or add a new one using the "Add" button. On local area network interfaces the upstream gateway should be "none". Selecting an upstream gateway causes the firewall to treat this interface as a WAN type interface.</small> |

nov. 29 11:28 

Reserved Networks

| | |
|---|--|
| Block private networks and loopback addresses | <input type="checkbox"/> <small>Blocks traffic from IP addresses that are reserved for private networks per RFC 1918 (10/8, 172.16/12, 192.168/16) and unique local addresses per RFC 4193 (fc00::/7) as well as loopback addresses (127/8). This option should generally be turned on, unless this network interface resides in such a private address space, too.</small> |
| Block bogon networks | <input checked="" type="checkbox"/> <small>Blocks traffic from reserved IP addresses (but not RFC 1918) or not yet assigned by IANA. Bogons are prefixes that should never appear in the Internet routing table, and so should not appear as the source address in any packets received. This option should only be used on external interfaces (WANs), it is not necessary on local interfaces and it can potentially block required local traffic. Note: The update frequency can be changed under System > Advanced, Firewall & NAT settings.</small> |

 Save

On configurera de la même manière le VLAN20 en assignant l'adresse IP suivant : **192.168.20.1**.

Attention de ne pas oublier de renseigner le masque de sous-réseau !

Ajout du DHCP Serveur

Configuration de Terraform

Avant de lancer la procédure de création il faut renseigner les variables de connexion pour le serveur **Proxmox** dans le fichier **GOAD/ad/GOAD/providers/proxmox/terraform/variables.tf.template**

Attention, pour que **Terraform** prenne en compte le fichier variables.tf, il faut changer l'extention en enlevant le **.template**. Dans l'optique d'avoir une version de sauvegarde en local on peut faire une copie du fichier avant de faire des modifications.

dans notre cas la configuration correspondra a :

```
variable "pm_api_url" {
  default = "https://10.202.3.33:8006/api2/json"
}

variable "pm_user" {
  default = "root@pam"
}

variable "pm_password" {
  default = "rootroot"
}

variable "pm_node" {
  default = "proxmox-goad"
}

variable "pm_pool" {
  default = "GOAD"
}

variable "pm_full_clone" {
  default = false
}
```

Provisionnement Proxmox via Ansible

[Source d'instruction](#)

configuration :

Afin de mener a bien le provisionning via Ansible on va venir installer les dependencies du projet se trouvant dans le fichier **GOAD/ansible/requirements.yml** via la commande suivante :

```
ansible-galaxy install -r requirements.yml
```

Dans ces requirements on va retrouver par exemple la capacité a utiliser Ansible sur le système Windows.

Pour continuer l'installation avec les scripts d'installation fournis; On vient *set* la variable d'environnement suivant pour

Dans le but d'également mettre les agents des SIEM directement sur le réseau, on va pouvoir mettre en place