



# Techniques de sécurisation ISR

Par Titouan RICHARD-CARRERE , Alan GUILLET et Marion GUERIN .

# Techniques de sécurisation ISR

## Introduction :

Nous avons travaillé avec :

- VMware
- PNETLAB
- VirtualBox
- GNS3
- Pfsense
- et d'autres modules.

On à tous fait le projet sur nos propres postes, mais principalement sur celui du milieu. L'avancement et la majorité des photos prises ont été faits sur celui-ci. L'ordinateur du milieu étant celui du chef de projet, on à repartis les autres tâches avec le power point par Alan GUILLET, le rapport par Marion GUERIN et Titouan RICHARD-CARRERE.

## SOMMAIRE :

Partie 1 : PnetLab

Partie 2 : Installation et configuration du réseau sous GNS3

Partie 3 : configuration des règles de sécurité

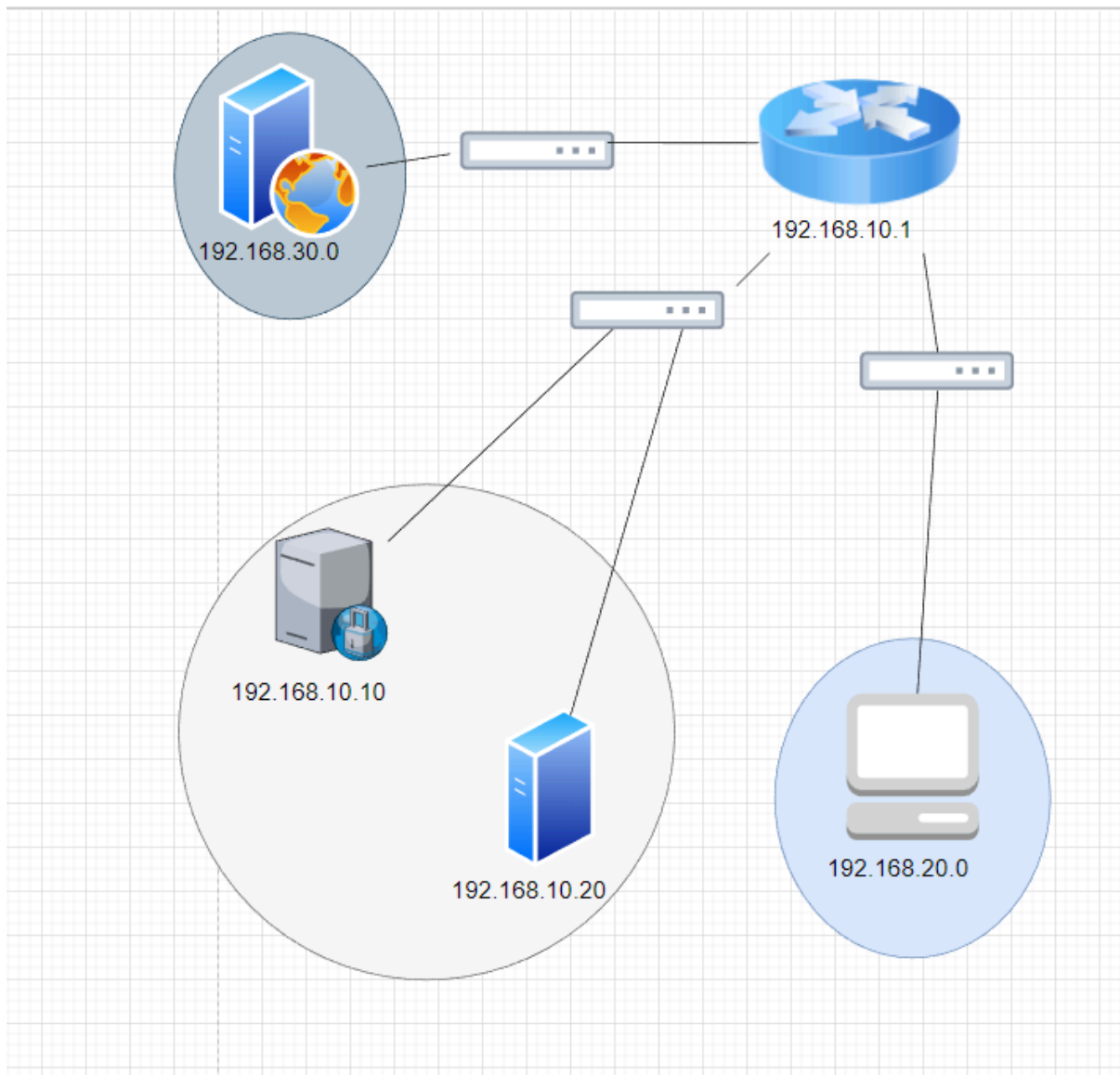
Partie 4 : Erreur rencontré

## Techniques de sécurisation ISR

### Partie 1 :

On commence par installer PNETLAB, via le lien dans le cahier des charges donné. On ouvre alors VMware, d'ici on ouvre une nouvelle machine virtuelle et utilisant le fichier PNETLAB.

On a lancé la machine VMware avec PNETLAB, on a configuré celle ci, cependant on à rencontré une issue, on n'arrivait pas à lancer la virtualisation alors on la lancé sans et ça marchait, ensuite, à partir de l'adresse IP bridge que la machine nous à donné on a accéder au site PNETLAB afin de faire des simulations. On à également fait un schéma



modélisant l'architecture de notre réseau initial.

## Techniques de sécurisation ISR

### Partie 2 : Installation et configuration du réseau sous GNS3

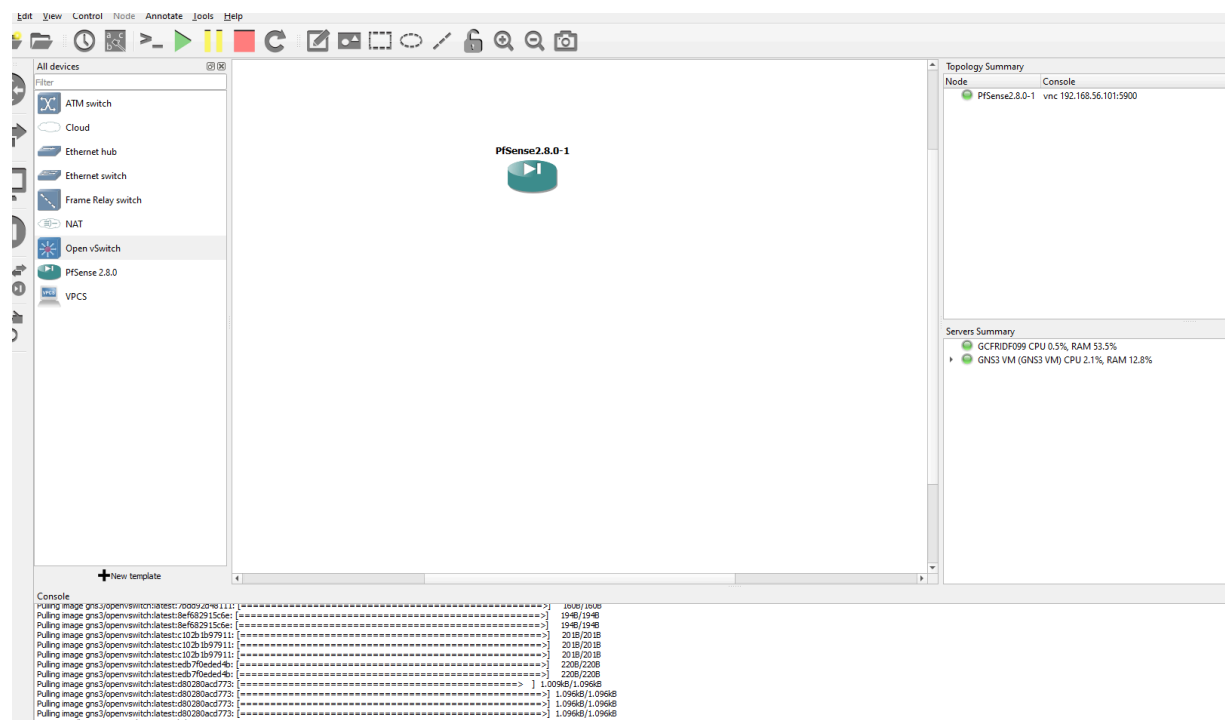
On passe sur GNS3 comme PNETLAB ne marche pas à cause de la virtualisation. On a réussi à lancer la machine sans problème, on a ouvert le logiciel GNS3, tout marchait, cependant quand on installe des templates on rencontrait des problèmes. Principalement lié à la virtualisation et au KVM, dont on a résolu une partie :

- par exemple pour le KVM sur pfsense on a eu de l'aide de la part de l'intervenant qui nous a dit de mettre la commande suivante avec les machines qui demandait du Linux, la commande est : `-machine accel=tcg`

**`-no-kvm` (removed in 5.2)**

The `-no-kvm` argument was a synonym for setting `-machine accel=tcg`.

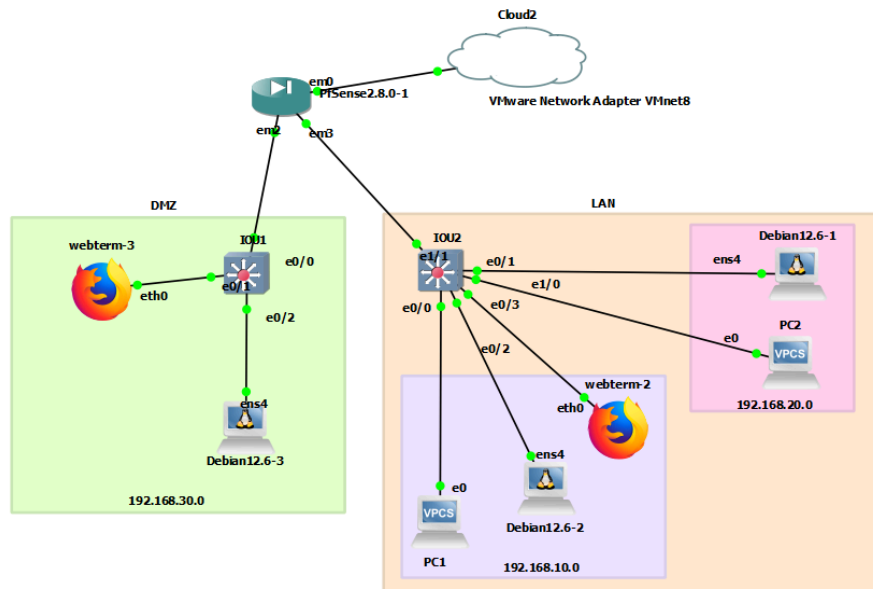
Après une recherche sur internet on peut comprendre que l'utilité de cette commande est de forcer le CPU à faire du software au lieu de virtualisation comme les ordinateurs qu'on utilisait ne prenaient pas cette option.



On retrouve également d'autres erreurs récurrentes comme une erreur de requête du client vers le serveur. Dans ces cas-là, on se contente de fermer complètement GNS3 et de rouvrir le projet.

## Techniques de sécurisation ISR

On commence vraiment le travail à partir d'ici,QQ  
on définit notre nouvelle architecture réseau avec cette topologie



Maintenant il faut que nous configurons les switches pour avoir les VLAN ainsi que les câbles ethernet pour les ajouter aux VLAN créé ( on configurera la DMZ plus tard ) :

VLAN	nom	statut	ports
1	default	active	Et1/2, Et1/3
10	VLAN_ADMIN	active	Et0/0, Et0/2, Et0/3
20	VLAN_CLIENT	active	Et0/1, Et1/0
30	teste	active	

commande utilisée

- conf t
- vlan {numéro du vlan}
- name {nom du vlan}
- exit
- interface {interface}
- switchport mode access
- switchport access vlan {numéro du vlan}
- no shutdown
- exit
- exit
- wr

On a également attribué statiquement les adresses IP aux machines. Comme on n'avait pas encore fait de DHCP on ne pouvait le faire que statiquement.

## Techniques de sécurisation ISR

On a également mis l'interface trunk sur le câble qui lit le switch au routeur pour pouvoir envoyer les paquets au pfsense. L'idée était de pouvoir faire du routage inter-VLAN.

```
IOU2(config-if)#switchport trunk encapsulation ?
dot1q      Interface uses only 802.1q trunking encapsulation when trunking
isl        Interface uses only ISL trunking encapsulation when trunking
negotiate   Device will negotiate trunking encapsulation with peer on
            interface

IOU2(config-if)#switchport trunk encapsulation do
IOU2(config-if)#switchport trunk encapsulation dot1q
IOU2(config-if)#switchport mode trunk
IOU2(config-if)#switchport trunk allowa
IOU2(config-if)#switchport trunk allo
IOU2(config-if)#switchport trunk allowed vlan 10-30,60
IOU2(config-if)#on sh
IOU2(config-if)#on shutdown
^
% Invalid input detected at '^' marker.

IOU2(config-if)#no shutdown
IOU2(config-if)#
*Oct 30 10:01:36.714: %LINK-3-UPDOWN: Interface Ethernet1/1, changed state to up
IOU2(config-if)#
*Oct 30 10:01:38.719: %LINEPROTO-5-UPDOWN: Line protocol on Interface Ethernet1/1, changed state to up
IOU2(config-if)#
```

Commande utilisé pour configurer l'interface trunk :

- conf t
- int {interface}
- switchport trunk encapsulation dot1q
- switchport mode trunk
- switchport trunk allowed VLAN ...
- no shutdown
- exit

Pour vérifier la configuration nous voulions tenter de ping nos deux machines entre eux, donc pour cela nous avons configuré notre routeur pour faire du routage inter-vlan .  
pour ce faire nous avons simplement déclarer les vlan que nous avons précédemment créer sur notre switch dans notre routeur,  
précisons que pour configurer nos vlans sur pfsense nous nous sommes servie du webtern de la DNZ car les machines sur les VLAN ne pouvais plus accéder au pfsense justement car nous avions pas encore définit les vlan sur le pfsense

Cependant nous avons malheureusement rencontré un contretemps lié à un problème de configuration réseau. En effet, les paquets étaient bloqués au niveau du routeur,

## Techniques de sécurisation ISR

Après 2 heures de recherche nous avons fini par comprendre que c'était car nous avions attribué une adresse IP à l'interface LAN. Cette configuration a provoqué un conflit : le même câble était utilisé à la fois pour le trunk et pour le LAN, ce qui entraînait la présence de la même interface et des adresses IP identiques. Ainsi, lorsque les paquets quittent le routeur pour rejoindre le switch, ils ne savaient pas s'ils devaient être acheminés vers le réseau LAN ou vers les VLANs, ce qui bloquait la communication.

une fois l'erreur corrigé cela donnait cette configuration

Interface	Network port
WAN	em0 (0c:f6:da:22:00:00)
LAN	em3 (0c:f6:da:22:00:03) <span>Delete</span>
VLAN_ADMIN	VLAN 10 on em3 - lan (ADMIN) <span>Delete</span>
VLAN_CLIENT	VLAN 20 on em3 - lan (CLIENT) <span>Delete</span>
DMZ	em2 (0c:f6:da:22:00:02) <span>Delete</span>
Available network ports:	em1 (0c:f6:da:22:00:01) <span>+ Add</span>

Save

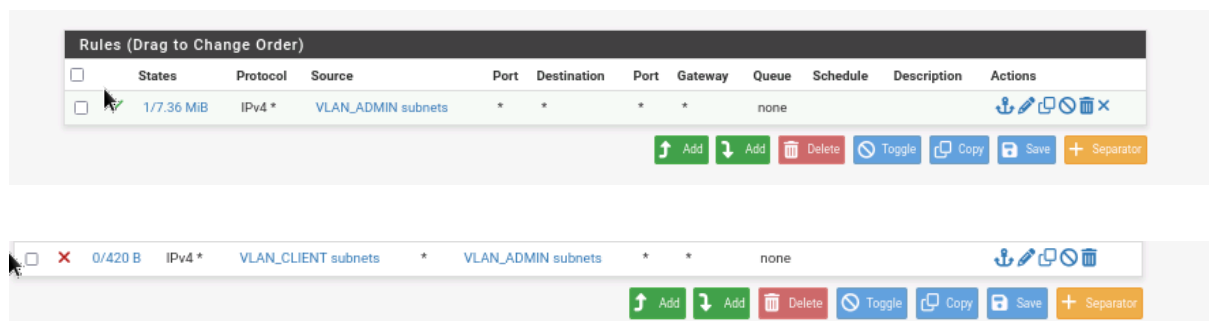
```
WAN (wan)      -> em0      -> v4/DHCP4: 192.168.146.142/24
LAN (lan)      -> em3      ->
VLAN_ADMIN (opt1) -> em3.10 -> v4: 192.168.10.1/24
VLAN_CLIENT (opt2) -> em3.20 -> v4: 192.168.20.1/24
DMZ (opt3)     -> em2      -> v4: 192.168.30.1/24
```

## Techniques de sécurisation ISR

### Partie 3 : configuration des règles de sécurité

Maintenant que nos appareils sont correctement configurés et que le trafic réseau est fonctionnel, nous passons à la configuration de notre firewall pour plus de sécurité :

Nous avons commencé par mettre 2 règles de filtrage afin que les machines du VLAN\_ADMIN puissent joindre les machines du VLAN\_CLIENT mais pas l'inverse.



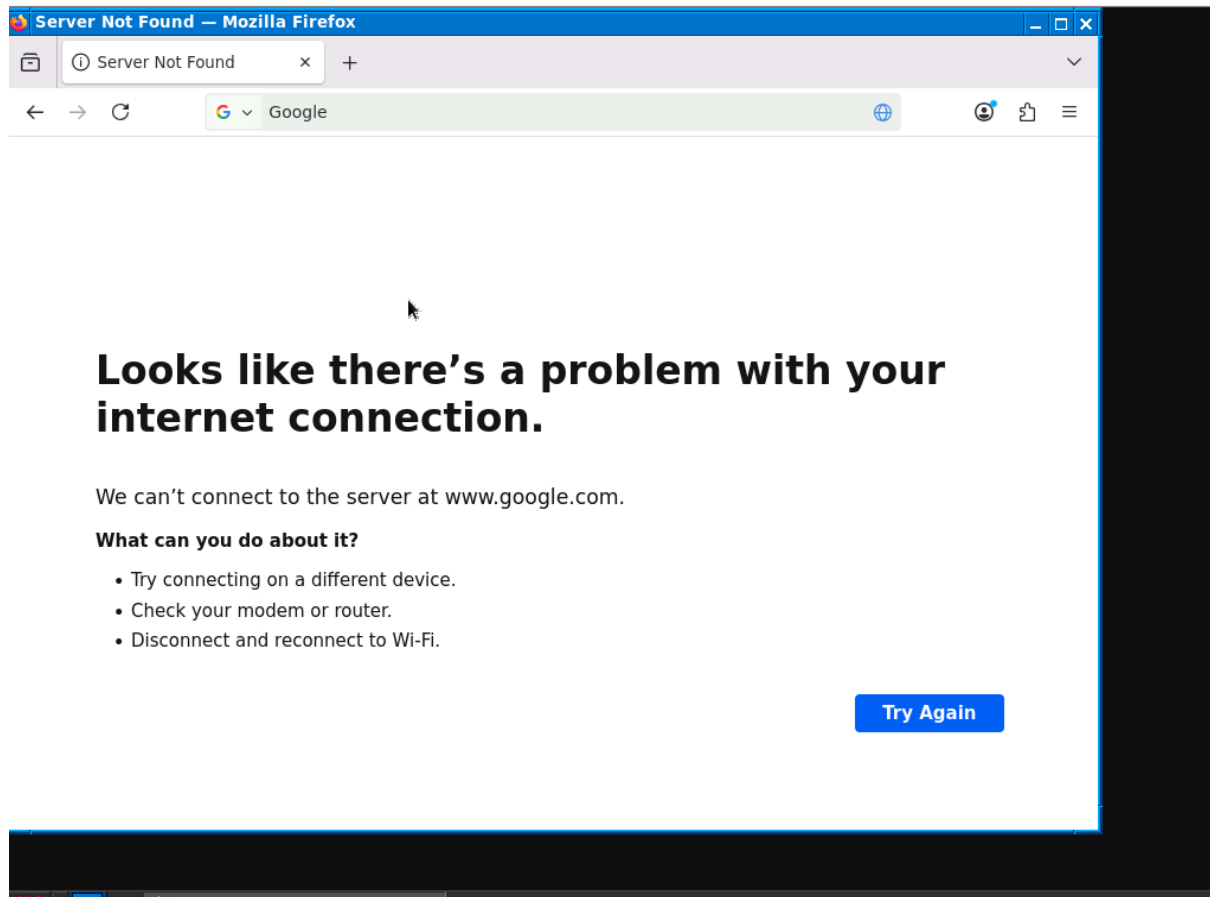
Désormais si on essaie de ping la machine admin à partir de la machine client on obtient :

```
PC2> ping 192.168.10.11
192.168.10.11 icmp_seq=1 timeout
```

Donc maintenant que les communications entre les différents VLANs sont configurés, on va essayer d'accéder à Internet.



Voilà le résultat :



Après avoir regarder notre configuration réseau, nous remarquons que notre webtern n'a pas d'adresse IP défini :

## Techniques de sécurisation ISR

```
File Edit Tabs Help
root@webterm-3:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
10: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UNKNOWN group default qlen 1000
    link/ether 02:42:d8:60:71:00 brd ff:ff:ff:ff:ff:ff
    inet6 fe80::42:d8ff:fe60:7100/64 scope link
        valid_lft forever preferred_lft forever
root@webterm-3:~#
```

Nous allons donc faire notre configuration réseau, avec les commandes suivantes :

- ip address add 192.168.30.50/24 dev eth0
- ip route add default via 192.168.30.1

```
root@webterm-3:~#
root@webterm-3:~# ip address add 192.168.30.50/24 dev eth0
root@webterm-3:~# ip route add default via 192.168.30.1
root@webterm-3:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
10: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UNKNOWN group default qlen 1000
    link/ether 02:42:d8:60:71:00 brd ff:ff:ff:ff:ff:ff
    inet 192.168.30.50/24 scope global eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::42:d8ff:fe60:7100/64 scope link
        valid_lft forever preferred_lft forever
```

Nous réessayons et nous obtenons toujours l'erreur précédente. nous allons vérifier le contenu du fichier /etc/resolv.conf. nous voyons qu'il est vide alors nous ajoutons la ligne :

```
nameserver 8.8.8.8
```

Cette ligne indique à la machine du serveur DNS quel DNS elle doit utiliser et nous observons alors que désormais nous pouvons accéder à internet.

## Techniques de sécurisation ISR

Maintenant qu'on a réussi configurer notre routage inter-vlan sécurisé et qu'on peut accéder à internet il nous faut configurer le NAT pour pouvoir accéder à internet

Ces deux règles stipulent comment le réseau de la DMZ accède à internet

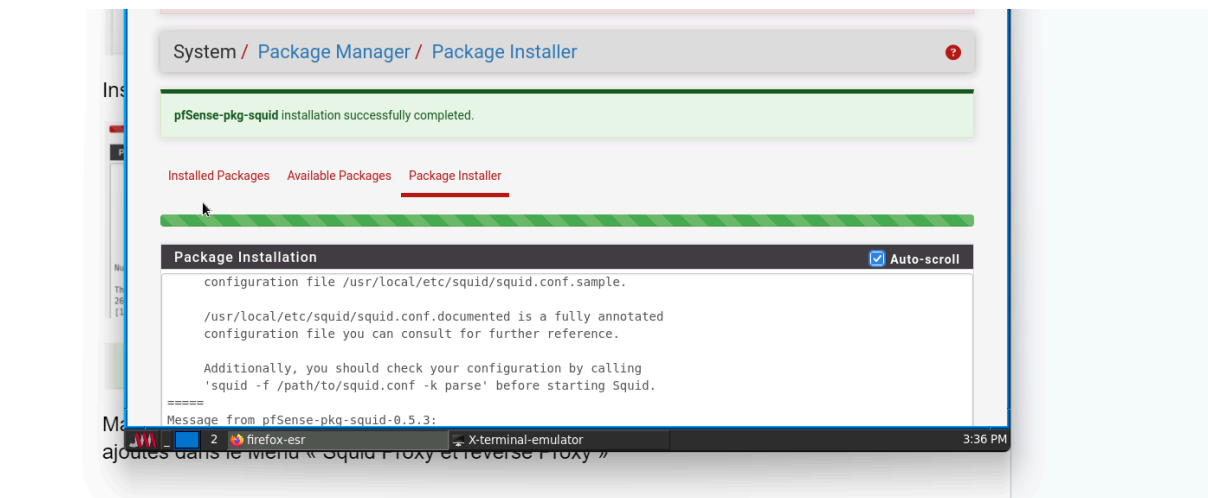
	Interface	Protocol	Source Address	Source Ports	Dest. Address	Dest. Ports	NAT IP	NAT Ports	Description	Actions
<input type="checkbox"/>	<input checked="" type="checkbox"/>	DMZ	TCP	DMZ subnets	*	! This Firewall (self)	80 (HTTP)	127.0.0.1	3128	
<input type="checkbox"/>	<input checked="" type="checkbox"/>	DMZ	TCP	DMZ subnets	*	! DMZ address	443 (HTTPS)	127.0.0.1	3129	

Add Add Delete Toggle Save Separator

Maintenant que nous avons configuré la dmz pour accéder à internet et que nos vlan communiquent entre eux de manière sécurisé. nous allons faire le proxy qui servira d'intermédiaire.

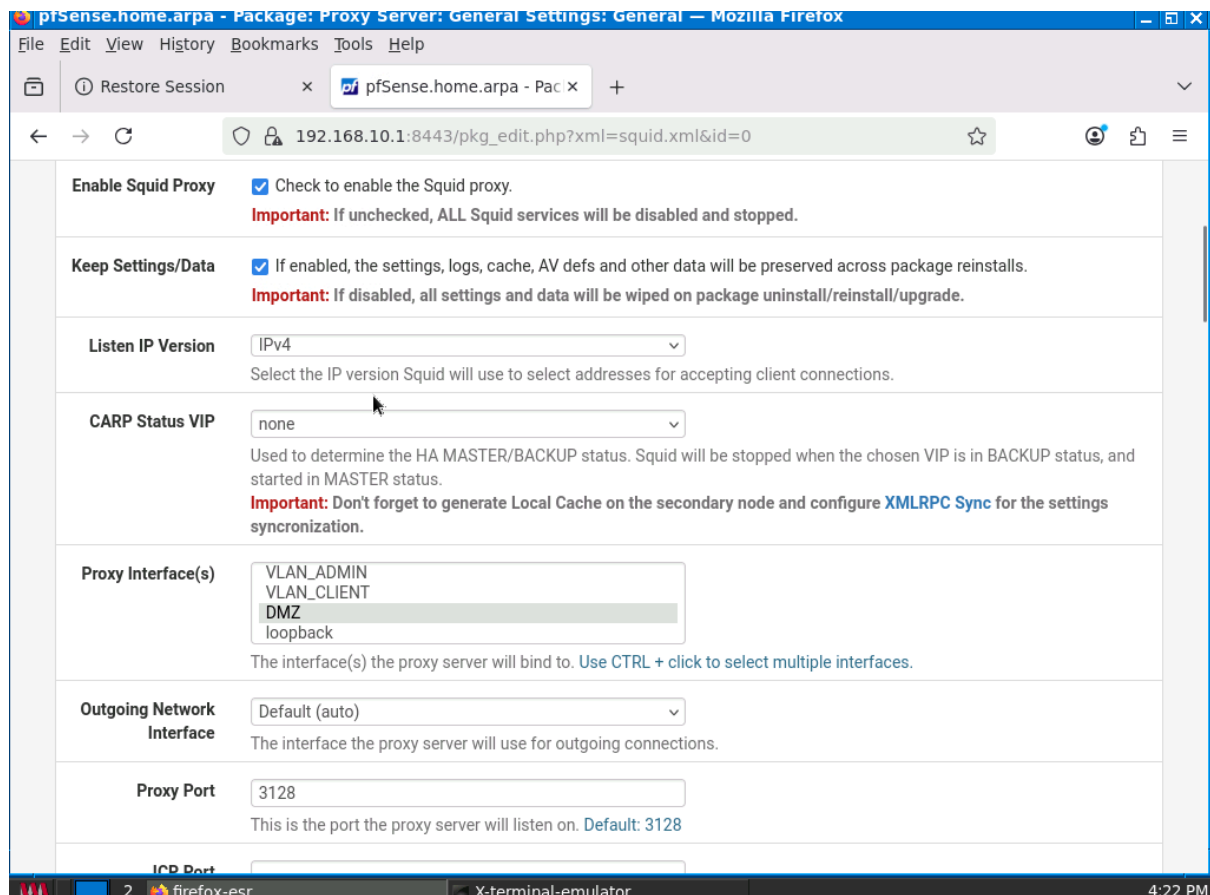
L'objectif de notre proxy ici sera notamment de pouvoir bloquer certains sites.

Nous allons configurer le proxy, pour cela on commence à l'installer par le pfsense :

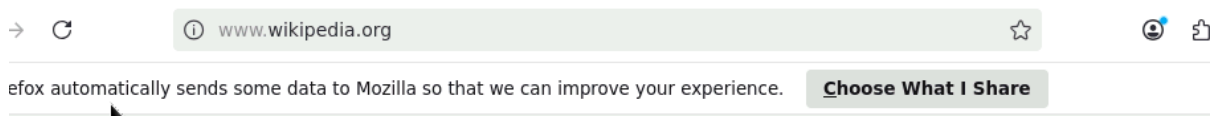


Et on suit les étapes du documents donnés.

## Techniques de sécurisation ISR



enfin pour l'exemple on décide de bloquer le site wikipédia.org



## Hmm. We're having trouble finding that site.

We can't connect to the server at [www.wikipedia.org](http://www.wikipedia.org).

**If you entered the right address, you can:**

- Try again later
- Check your network connection
- Check that Firefox has permission to access the web (you might be connected but behind a firewall)

[Try Again](#)

Voilà , nous avons donc une architecture réseau fonctionnelle et sécurisée. Ce projet nous aura beaucoup appris, notamment car nos connaissances avec les logiciels de topologie réseau comme GNS3 étaient très limitées.

Merci d'avoir lu ce rapport.