

Firewalling sous BSD - PF

Ivan KURZWEG

20 avril 2009

Table des matières

1	Firewalls	2
1.1	Définitions, principes	2
1.2	Machines Firerwalls	2
1.3	Principes	3
2	OpenBSD pf	3
2.1	Fonctionnalités	3
2.2	Activation sous FreeBSD	4
2.3	Contrôle de PF	5
2.4	Logging	5
3	Configuration de PF	6
3.1	Listes et macros	6
3.1.1	Listes	6
3.1.2	Macros	7
3.2	Les tables	7
3.2.1	Déclaration	7
3.3	Options de normalisation et d'optimisations	8
3.4	Translations d'adresses et redirections	9
3.4.1	Translation d'adresses	9
3.4.2	Redirection de ports	9
3.5	Filtrage	10
3.5.1	Politique par défaut	10
3.5.2	Règles de filtrage	10
4	TP	11
4.1	Environnement	11
4.2	Schéma général du réseau	11
4.3	Travail à faire	11
5	Un exemple de fichier pf.conf complet pour un serveur DNS	12

6	Références	14
6.1	Références	14

Table des figures

1	Schéma du réseau	11
---	----------------------------	----

Résumé

Introdction (cours et TP) sur le firewall PF d'OpenBSD ... Not an How-TO !!
 On prendra soin de consulter le man de pfctl, et de suivre l'excellente documentation PF sur le site www.openbsd.org. Ne sont pas abordées les techniques de gestion de bande passante (AltQ).

This is my network. It is mine or technically my employer's, it is my responsibility and I care for it with all my heart there are many other networks a lot like mine, but none are just like it. I solemnly swear that I will not mindlessly paste from HOWTOs.

—Cantique du SyAdmin, Peter N. M. Hansteen

1 Firewalls

1.1 Définitions, principes

- Un pare-feu est un élément du réseau informatique, logiciel et/ou matériel, qui a pour fonction de faire respecter la politique de sécurité du réseau, celle-ci définissant quels sont les types de communication autorisés ou interdits. ([Wikipedia](#))
- Pf : *OpenBSD's Packet Filter*, a été intégré dans le système base d'OpenBSD 3.0 en Décembre 2009. Le développement en urgence de PF a été causé par un problème de licence sur l'ancien système de filtrage de paquets, IPFilter.
- *Statefull* : L'état peut être conservé pour un trafic spécifique. Les firewall sont configurés pour autoriser certains trafics (par exemple, le trafic sortant). L'état du trafic est mémorisé pour chaque extrémité de la connexion logique (aussi bien pour le protocole tcp (orienté connexion) que pour les protocoles udp, ou icmp) et des règles implicites sont dynamiquement chargées/retirées du firewall selon l'état de la connexion.
- *Netfilter/Iptables* : système de filtrage de paquet sous Linux, incluant firewalling statefull, nat, pat, ...

1.2 Machines Firerwalls

On trouve principalement deux types de matériels :

- Machines Unix, mais attention
 - aux installations par défaut, qui activent des services potentiellement dangereux ...

- à l'existence de compte sans mot de passe.
- Matériels dédiés (type Cisco ...), mais
 - attention aux coûts !

1.3 Principes

Les firewalls interviennent généralement sur plusieurs plans :

- Les paquets : les firewalls interviennent généralement sur la Couche 3 IP. Ainsi, ils peuvent analyser les différents champs des paquets IP.
- Les protocoles : les fonctionnements de UDP, ICMP et de TCP imposent aux firewalls de manipuler les paquets et de stocker éventuellement un état des transactions (statefull)
- Les connexions : lors de la connexion d'un client sur un serveur, les échanges doivent être garantis par le firewall, si le trafic est autorisé
- Les ports : en jouant sur les ports, les firewalls filtrent le trafic, et sont capables de rediriger des flux entrants ou sortants sur des cibles différentes.
- Les adresses : selon les adresses sources et destinations (unicast, broadcast, network ou multicast)

2 OpenBSD pf

2.1 Fonctionnalités

- c'est un stateful firewall (décisions basées sur l'état d'une connexion logique)
- il utilise la notion de macros (variables à développer), ce qui permet de paramétrer le jeu de règles ; ce qui facilite la maintenance et améliore la portabilité ;
- il utilise des tables (de hachage) qui permettent de stocker de large plage d'adresses, tout en assurant un temps d'accès constant ;
- il implémente le NAT (ré-écriture des adresses sources en sortie) et la redirection (ré-écriture des adresses destination en entrée) ;
- il opère dans le mode de la « dernière correspondance gagnante », mais propose un mode de court-circuit avec le mot clé quick ;
- il propose une fonction de normalisation de trafic (scrub) qui annule toutes ambiguïtés dans les paquets reçus et à transmettre aux destinataires (en entrée ou en sortie) ;
- il supporte nativement plusieurs méthodes implémentant le contrôle de bande passante (par gestion de files de priorités selon différents algorithmes)
- il dispose d'un mécanisme implicite d'ordonnement des règles ;
- il dispose d'un excellent mécanisme de log et de statistiques.

2.2 Activation sous FreeBSD

Si PF est disponible directement sous OpenBSD, sa mise en oeuvre sous FreeBSD implique généralement la recompilation du noyau, ou le chargement dynamique du module pf :

Example 2.1 Chargement manuel du module pf sous FreeBSD

```
# kldload pf.ko
```

Example 2.2 Chargement automatique du module pf sous FreeBSD, fichier `/boot-loader.conf`

```
pf_load="YES"  
pflog_load="YES"
```

Example 2.3 Modification du noyau

```
device      pf  
device      pflog  
device      pfsync  
  
options     ALTQ  
options     ALTQ_CBQ  
options     ALTQ_RED  
options     ALTQ_RIO  
options     ALTQ_HFSC  
options     ALTQ_PRIQ
```

Dans les deux cas, il est nécessaire d'intervenir sur le fichier `/etc/rc.conf` pour activer PF au démarrage de la machine, et charger les jeux de règles :

Example 2.4 Fichier `rc.conf`

```
pf_enable="YES"           # Enable PF (load module if required)
pf_rules="/etc/pf.conf"  # rules definition file for pf
pf_flags=""              # additional flags for pfctl startup
pflog_enable="YES"       # start pflogd(8)
pflog_logfile="/var/log/pflog" # where pflogd should store the logfile
pflog_flags=""           # additional flags for pflogd startup

gateway_enable="YES"     # Enable as LAN gateway
```

2.3 Contrôle de PF

PF possède un puissant outil de manipulation des règles, des tables et de collecte de statistiques, **pfctl**.

```
pfctl [ [-AdeghmNnOqRrvz] [-a anchor] [-D macro= value] [-F modifier] [-f
file] [-i interface] [-K host | network] [-k host | network] [-o [level]] [-p device] [-s
modifier] [-t table -T command [address ...]] [-x level]]
```

Example 2.5 Utilisations de la commande `pfctl`

```
# pfctl -d //désactiver pf
# pfctl -e //activer pf
# pfctl -f /etc/pf.conf -n //tester le fichier de configuration
# pfctl -f /etc/pf.conf -v //charger en mode verbeux le fichier de conf
# pfctl -sr //afficher le jeux de règles en cours
# pfctl -s info //statistiques de filtrage
# pfctl -t banned -T show //afficher le contenu de la table banned
# pfctl -t banned -T add 192.168.0.12/32 //ajouter un hôte à la table ←
banned
```

2.4 Logging

Lorsque les directives de “log” sont activés, les paquets voulus sont envoyés vers la pseudo interface `pflog0`. Cette pseudo interface réseau est alors directement atteignable via les classiques outils d’analyse réseau :

Example 2.6 Analyse temps réel de l'activité de PF :

```
# tcpdump -netvli pflog0

rule 9/(match) [uid 0, pid 2208] block in on em0: 172.17.10.31.1900 > ←
    239.255.255.250.1900: udp 318 (ttl 127, id 51797, len 346)
rule 1/(match) [uid 0, pid 2208] block in on em0: 172.17.8.6.138 > ←
    172.17.8.127.138: udp 201 (ttl 128, id 6662, len 229)
rule 1/(match) [uid 0, pid 2208] block in on em0: 172.17.8.6.138 > ←
    172.17.8.127.138: udp 201 (ttl 128, id 6662, len 229, bad cksum 0!)
....
```

Cette facilité d'analyse rend PF bien plus agréable à utiliser que Netfilter/Iptables sous Linux ! En effet, par la

3 Configuration de PF

D'une manière générale, les règles de PF sont définies dans un seul fichier de configuration, par défaut le fichier `/etc/pf.conf`. Ce fichier est composé de plusieurs sections :

- Les définitions en utilisant les macros, les tables et les listes
- les règles d'optimisation et de contrôle de bande passante
- les règles de NAT
- les règles de filtrage

3.1 Listes et macros

3.1.1 Listes

Une liste permet de manipuler des valeurs de même types (liste d'adresses, liste de ports, ...). Elle sont repérées par des accolades.

Example 3.1 Exemples de listes

```
block out on fxp0 from { 192.168.0.1, 10.5.32.6 } to any
trusted = "{ 192.168.1.2 192.168.5.36 }"
pass in inet proto tcp from { 10.10.0.0/24 $trusted } to port 22
```

3.1.2 Macros

Les macros sont des variables définies par l'utilisateur. En stockant des valeurs, elles facilitent la lecture du fichier de configuration et sa maintenance.

Example 3.2 Exemples de macros

```
host1 = "192.168.1.1"
host2 = "192.168.1.2"
all_hosts = "{" $host1 $host2 "}"
ext_if = "fxp0"
block in on $ext_if from any to any
```

3.2 Les tables

Une table permet le regroupement d'adresses au sein d'une même entité. Les spécificités des tables par rapport aux listes standard sont les suivantes :

- Utilisation directe et multiple dans les règles, comparable à une macro
- Les recherches d'une adresse sont beaucoup plus rapides (utilise moins de mémoire et de temps CPU qu'une simple liste)
- Existence qui va au-delà du fichier de configuration où elles sont définies puisqu'elles résident ensuite en mémoire ce qui permet de leur appliquer des modifications au niveau de leur contenu. Ces modifications sont immédiatement prises en compte par Packet Filter sans avoir à recharger les règles ou à procéder à une quelconque manipulation.

3.2.1 Déclaration

Le contenu d'une table est sensiblement identique à celui que l'on peut fournir comme adresse source ou de destination dans les règles en temps normal. Il peut prendre les différentes formes suivantes :

- Une adresse IP (version 4 comme 6), exemples : 192.168.0.1, 66.80.97.134
- Une adresse en notation CIDR (version 4 comme 6) : 192.168.0.0/24
- Un nom de machine qui sera automatiquement remplacé par les adresses IP correspondantes (versions 4 et 6) à condition de pouvoir effectuer cette résolution (ne pas oublier de construire des règles visant à autoriser le trafic DNS)
- Le nom d'une interface (*sis1*, *lo0*, etc), qui sera substituée par la liste des adresses qui lui sont attribuées
- Le nom d'une interface suivie de `:network`, `:broadcast`, `:peer` ou `:0` correspondant respectivement à l'adresse réseau, l'adresse de broadcast, l'adresse d'un pair sur un lien point à point ou l'adresse principale affectée à une carte réseau (ne tient pas compte des alias)

Example 3.3 Exemples de tables

```
table <dmz_hosts> const { \  
    172.17.0.1/32,    \  
    172.17.0.239/32, \  
    172.17.0.240/32  \  
}  
table <banned> persist
```

La manipulation des tables est possible en utilisant la commande **pfctl** (voir chapitre précédent).

3.3 Options de normalisation et d'optimisations

Dans un fichier de configuration de PF, il peut-être intéressant de spécifier un certain nombre de valeurs, modifiant les réactions du firewall. Sont présentées les plus communes :

- `block-policy` : ce paramètre spécifie si les tentatives de connexion refusées sur des ports provoquent ou non un retour d'erreur (de type *connexion-refused*). Sa valeur peut-être *drop* (valeur par défaut) ou *return*.

```
set block-policy return
```

- `skip` : cette directive permet d'ignorer les règles sur une interface (typiquement la boucle locale).

```
set skip on lo0
```

- `scrub` : il est possible pour PF de normaliser une partie du trafic réseau en réassemblant des paquets fragmentés ou en en éliminant certains. L'utilisation commune de cette valeur est :

```
scrub in all
```

- `antispoof` : cette directive permet de détecter et bloquer les tentatives d'usurpation d'adresse IP, sur les réseaux directement connectés à PF. Il permet par exemple de bloquer des paquets venant du WAN portant des adresses locales.

```
antispoof for $ext_if  
antispoof for $int_if
```

3.4 Translations d'adresses et redirections

3.4.1 Translation d'adresses

Mécanisme indispensable à l'interconnexion de réseaux de différentes classes d'adresses, la translation d'adresse (NAT : Network Address Translation) permet de modifier les adresses sources du trafic, pour masquer par exemple la structure du LAN. Dans l'exemple proposé en TP (cf. Figure 1 schéma réseau), lors d'une connexion d'une machine du LAN (192.168.36/0/24) vers un serveur du WAN, la passerelle/pare-feu transmettrait le flux vers le WAN, en traduisant l'adresse source vers sa propre adresse (172.17.0.1). Le trafic retour est alors automatiquement traduit de manière symétrique, et les paquets remis à la machine du LAN.

Sous PF, il est possible de définir sur quelles interfaces, pour quels flux, la translation d'adresse doit se faire. La syntaxe générale des règles NAT est la suivante :

```
nat [pass] [log] on interface [af] from src_addr [port src_port] to \
  dst_addr [port dst_port] -> ext_addr [pool_type] [static-port]
```

- `pass` : désactive le filtrage pour les paquets traduits
- `interface` : l'interface sur laquelle les paquets sont traduits
- `ext_addr` : l'adresse (ou les adresses) sur laquelle les paquets sont traduits

Exemple 3.4 Exemple de règle NAT

```
ext_if = "re0"
int_if = "re1"
localnet = $int_if:network
nat on $ext_if from $localnet to any -> ($ext_if)
block all
pass from { lo0, $localnet } to any keep state
```

3.4.2 Redirection de ports

La redirection de ports permet de rediriger les flux arrivant sur une interface, un port vers une machine physique, sur un port précis. Dans l'exemple proposé en TP (cf. Figure 1 schéma réseau), le serveur WEB de la DMZ doit être joignable du WAN. IL faut donc rediriger les connexions sur le port HTTP vers le serveur WEB.

Exemple 3.5 Exemple de règle de redirection

```
server = 192.168.1.40
rdr on $ext_if proto tcp from any to $ext_if port 80 -> $server \
  port 80
```

3.5 Filtrage

Les règles de filtrages sont examinées dans l'ordre d'apparition, la dernière règle correspondant au trafic examinée est appliquée. Si aucune règle ne correspond au paquet, la politique par défaut est appliquée. Dans le cas où le mot clef `quick` est passé dans la règle, la règle est immédiatement appliquée, sans parcourir le reste des règles.

3.5.1 Politique par défaut

De manière générale, la politique par défaut d'un firewall est de bloqué tout trafic, puis d'autoriser uniquement ce que l'on souhaite.

Example 3.6 Politique par défaut

```
block in all
block out all
```

3.5.2 Règles de filtrage

La syntaxe générale des règles est la suivante :

```
action [direction] [log] [quick] [on interface] [af] [proto protocol] \
[from src_addr [port src_port]] [to dst_addr [port dst_port]] \
[flags tcp_flags] [state]
```

- **action** : prend les mots clefs `pass` ou `block`
- **quick** : si le mot clef est présent et que le paquet correspond à la règle, l'action est immédiatement appliquée
- **on interface** : l'interface où a lieu le filtrage
- **state** : si la règle contient les mots clefs `keep state`, les paquets futurs correspondant à cet échange seront acceptés

Example 3.7 Exemples de règles

```
pass in on dc0 from 192.168.0.0/24 to 192.168.0.1
pass out on dc0 from 192.168.0.1 to 192.168.0.0/24
pass in on fxp0 proto tcp from any to fxp0 port www
block in on fxp0 proto tcp from any to any port ssh
pass out on fxp0 proto tcp from any to any keep state
```

4 TP

4.1 Environnement

Le TP peut-être réalisé pour des raisons de simplification de mise en oeuvre sur des machines virtuelles. Il s'agit de simuler un réseau LAN classique contenant :

- LAN : une ou plusieurs machines simulant des postes de travail d'un LAN
- DMZ : un ou plusieurs serveurs abritant par exemple un serveur WEB, accessible depuis le WAN, et depuis le LAN en HTTP et SSH
- WAN : une machine simulant un poste sur Internet, qui pourra jouer le rôle d'un pirate ou d'un client WEB, et un serveur jouant le rôle de "site internet"

4.2 Schéma général du réseau

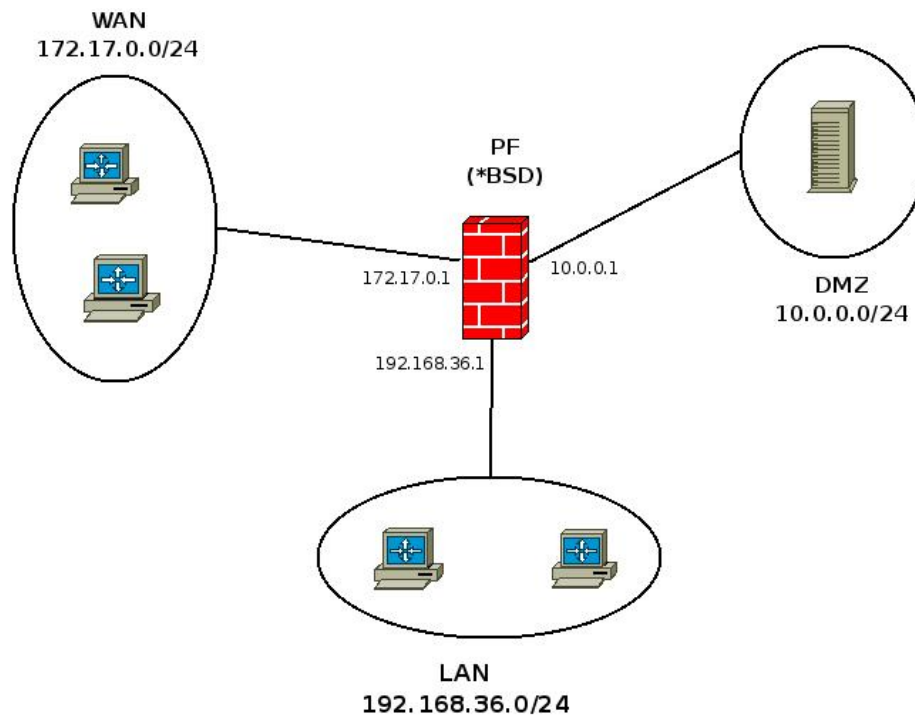


FIG. 1 – Schéma du réseau

4.3 Travail à faire

Les différents objectifs à atteindre :

- Mise en place des réseaux : installation des machines (éventuellement virtuelles), des services (serveurs WEB, SSH, ...)
- Configuration de la passerelle : interconnexion des réseaux, tests (ICMP)
- Configuration du Firewall : filtrage des paquets en fonctions des réseaux, des services, etc ...

Charge aux stagiaires d'imaginer LEUR réseau ...

5 Un exemple de fichier pf.conf complet pour un serveur DNS

Le fichier de configuration suivant pourrait être appliqué à un serveur DNS. Certaines parties ont volontairement été supprimées.

Example 5.1 Exemples d'un fichier pf.conf

```
### ←
===== ↔

### Macros: define common values, so they can be referenced ←
and changed easily.
### ←
===== ↔

dmz_if="xl0"
vlan_if="vlan30"

ssh_port="3022"
unPrivPorts="{ 1024 >< 65535 }"

proxy_p="3128"
ftp_RP="{ ftp, 30000:40000 }"

### ←
===== ↔

### Tables: similar to macros, but more flexible for many ←
addresses.
### ←
===== ↔

table <ssh_host> const { \
    172.17.0.1/32, \
    172.17.0.239/32, \
    172.17.0.240/32 \
}

table <dmz_net> const { \
    172.17.0.1/32, \
    172.17.0.126/32, \
    172.17.0.200/32, \
    172.17.0.201/32, \
    172.17.0.223/32, \
    172.17.0.225/32, \
    172.17.0.228/31, \
    172.17.0.230/32, \
    172.17.0.231/32, \
    172.17.0.232/32, \
    172.17.0.233/32, \
    172.17.0.234/32 \
    172.17.0.237/32, \
    172.17.0.239/32, \
    172.17.0.240/28 \
}

table <auth_net> const { \
    172.17.1.0/25, \
    172.17.2.0/25, \
    172.17.3.0/25, \
    172.17.4.0/25, \
    172.17.5.0/25, \
    172.17.6.0/25, \
    172.17.7.0/25, \
    172.17.8.0/25, \
    172.17.9.0/25, \
    172.17.10.0/25, \
    172.17.11.0/25, \
    172.17.12.0/25, \
    172.17.13.0/25, \
    172.17.14.0/25, \
    172.17.15.0/25, \
    172.17.16.0/25, \
    172.17.17.0/25, \
    172.17.18.0/25, \
    172.17.19.0/25, \
    172.17.20.0/25, \
    172.17.21.0/25, \
    172.17.22.0/25, \
    172.17.23.0/25, \
    172.17.24.0/25, \
    172.17.25.0/25, \
    172.17.26.0/25, \
    172.17.27.0/25, \
    172.17.28.0/25, \
    172.17.29.0/25, \
    172.17.30.0/25, \
    172.17.31.0/25, \
    172.17.32.0/25, \
    172.17.33.0/25, \
    172.17.34.0/25, \
    172.17.35.0/25, \
    172.17.36.0/25, \
    172.17.37.0/25, \
    172.17.38.0/25, \
    172.17.39.0/25, \
    172.17.40.0/25, \
    172.17.41.0/25, \
    172.17.42.0/25, \
    172.17.43.0/25, \
    172.17.44.0/25, \
    172.17.45.0/25, \
    172.17.46.0/25, \
    172.17.47.0/25, \
    172.17.48.0/25, \
    172.17.49.0/25, \
    172.17.50.0/25, \
    172.17.51.0/25, \
    172.17.52.0/25, \
    172.17.53.0/25, \
    172.17.54.0/25, \
    172.17.55.0/25, \
    172.17.56.0/25, \
    172.17.57.0/25, \
    172.17.58.0/25, \
    172.17.59.0/25, \
    172.17.60.0/25, \
    172.17.61.0/25, \
    172.17.62.0/25, \
    172.17.63.0/25, \
    172.17.64.0/25, \
    172.17.65.0/25, \
    172.17.66.0/25, \
    172.17.67.0/25, \
    172.17.68.0/25, \
    172.17.69.0/25, \
    172.17.70.0/25, \
    172.17.71.0/25, \
    172.17.72.0/25, \
    172.17.73.0/25, \
    172.17.74.0/25, \
    172.17.75.0/25, \
    172.17.76.0/25, \
    172.17.77.0/25, \
    172.17.78.0/25, \
    172.17.79.0/25, \
    172.17.80.0/25, \
    172.17.81.0/25, \
    172.17.82.0/25, \
    172.17.83.0/25, \
    172.17.84.0/25, \
    172.17.85.0/25, \
    172.17.86.0/25, \
    172.17.87.0/25, \
    172.17.88.0/25, \
    172.17.89.0/25, \
    172.17.90.0/25, \
    172.17.91.0/25, \
    172.17.92.0/25, \
    172.17.93.0/25, \
    172.17.94.0/25, \
    172.17.95.0/25, \
    172.17.96.0/25, \
    172.17.97.0/25, \
    172.17.98.0/25, \
    172.17.99.0/25, \
    172.17.100.0/25, \
    172.17.101.0/25, \
    172.17.102.0/25, \
    172.17.103.0/25, \
    172.17.104.0/25, \
    172.17.105.0/25, \
    172.17.106.0/25, \
    172.17.107.0/25, \
    172.17.108.0/25, \
    172.17.109.0/25, \
    172.17.110.0/25, \
    172.17.111.0/25, \
    172.17.112.0/25, \
    172.17.113.0/25, \
    172.17.114.0/25, \
    172.17.115.0/25, \
    172.17.116.0/25, \
    172.17.117.0/25, \
    172.17.118.0/25, \
    172.17.119.0/25, \
    172.17.120.0/25, \
    172.17.121.0/25, \
    172.17.122.0/25, \
    172.17.123.0/25, \
    172.17.124.0/25, \
    172.17.125.0/25, \
    172.17.126.0/25, \
    172.17.127.0/25, \
    172.17.128.0/25, \
    172.17.129.0/25, \
    172.17.130.0/25, \
    172.17.131.0/25, \
    172.17.132.0/25, \
    172.17.133.0/25, \
    172.17.134.0/25, \
    172.17.135.0/25, \
    172.17.136.0/25, \
    172.17.137.0/25, \
    172.17.138.0/25, \
    172.17.139.0/25, \
    172.17.140.0/25, \
    172.17.141.0/25, \
    172.17.142.0/25, \
    172.17.143.0/25, \
    172.17.144.0/25, \
    172.17.145.0/25, \
    172.17.146.0/25, \
    172.17.147.0/25, \
    172.17.148.0/25, \
    172.17.149.0/25, \
    172.17.150.0/25, \
    172.17.151.0/25, \
    172.17.152.0/25, \
    172.17.153.0/25, \
    172.17.154.0/25, \
    172.17.155.0/25, \
    172.17.156.0/25, \
    172.17.157.0/25, \
    172.17.158.0/25, \
    172.17.159.0/25, \
    172.17.160.0/25, \
    172.17.161.0/25, \
    172.17.162.0/25, \
    172.17.163.0/25, \
    172.17.164.0/25, \
    172.17.165.0/25, \
    172.17.166.0/25, \
    172.17.167.0/25, \
    172.17.168.0/25, \
    172.17.169.0/25, \
    172.17.170.0/25, \
    172.17.171.0/25, \
    172.17.172.0/25, \
    172.17.173.0/25, \
    172.17.174.0/25, \
    172.17.175.0/25, \
    172.17.176.0/25, \
    172.17.177.0/25, \
    172.17.178.0/25, \
    172.17.179.0/25, \
    172.17.180.0/25, \
    172.17.181.0/25, \
    172.17.182.0/25, \
    172.17.183.0/25, \
    172.17.184.0/25, \
    172.17.185.0/25, \
    172.17.186.0/25, \
    172.17.187.0/25, \
    172.17.188.0/25, \
    172.17.189.0/25, \
    172.17.190.0/25, \
    172.17.191.0/25, \
    172.17.192.0/25, \
    172.17.193.0/25, \
    172.17.194.0/25, \
    172.17.195.0/25, \
    172.17.196.0/25, \
    172.17.197.0/25, \
    172.17.198.0/25, \
    172.17.199.0/25, \
    172.17.200.0/25, \
    172.17.201.0/25, \
    172.17.202.0/25, \
    172.17.203.0/25, \
    172.17.204.0/25, \
    172.17.205.0/25, \
    172.17.206.0/25, \
    172.17.207.0/25, \
    172.17.208.0/25, \
    172.17.209.0/25, \
    172.17.210.0/25, \
    172.17.211.0/25, \
    172.17.212.0/25, \
    172.17.213.0/25, \
    172.17.214.0/25, \
    172.17.215.0/25, \
    172.17.216.0/25, \
    172.17.217.0/25, \
    172.17.218.0/25, \
    172.17.219.0/25, \
    172.17.220.0/25, \
    172.17.221.0/25, \
    172.17.222.0/25, \
    172.17.223.0/25, \
    172.17.224.0/25, \
    172.17.225.0/25, \
    172.17.226.0/25, \
    172.17.227.0/25, \
    172.17.228.0/25, \
    172.17.229.0/25, \
    172.17.230.0/25, \
    172.17.231.0/25, \
    172.17.232.0/25, \
    172.17.233.0/25, \
    172.17.234.0/25, \
    172.17.235.0/25, \
    172.17.236.0/25, \
    172.17.237.0/25, \
    172.17.238.0/25, \
    172.17.239.0/25, \
    172.17.240.0/25, \
    172.17.241.0/25, \
    172.17.242.0/25, \
    172.17.243.0/25, \
    172.17.244.0/25, \
    172.17.245.0/25, \
    172.17.246.0/25, \
    172.17.247.0/25, \
    172.17.248.0/25, \
    172.17.249.0/25, \
    172.17.250.0/25, \
    172.17.251.0/25, \
    172.17.252.0/25, \
    172.17.253.0/25, \
    172.17.254.0/25, \
    172.17.255.0/25, \
}
```

6 Références

6.1 Références

- [1] *Firewalls* - Pascal Picard, <http://zero202.free.fr/cs9-fire/html/> .
- [2] *FAQ OpenBSD*, <http://www.openbsd.org/faq/pf/index.html> .
- [3] *PF* - Peter N. M. Hansteen, <http://home.nuug.no/~peter/pf/en/index.html> .
- [4] *The Book of PF* - NO STARCH PRESS , Peter N.M. Hansteen .