

# TP Sécurité des réseaux

BRIZAI Olivier  
THORAVAL Maxime

3 février 2011

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Installation et Configurations préliminaires</b>	<b>4</b>
2.1	Client . . . . .	4
2.2	Serveur . . . . .	5
<b>3</b>	<b>Configuration Inside</b>	<b>6</b>
3.1	Configuration du NAT . . . . .	6
3.2	Règles de filtrage . . . . .	7
<b>4</b>	<b>Configuration DMZ</b>	<b>14</b>
4.1	Configuration du NAT . . . . .	14
4.2	Règles de filtrage . . . . .	14
4.3	HTTP et SSH à partir du réseau ENSICAEN . . . . .	18
<b>5</b>	<b>Annexe</b>	<b>21</b>
5.1	Configuration firewall . . . . .	21

# 1 Introduction

Le but de ce TP est de mettre en place un réseau sécurisé à l'aide d'un firewall CISCO ASA.

Ci-dessous, le réseau que nous souhaitons obtenir (les règles de filtrage ne sont pas représentées).

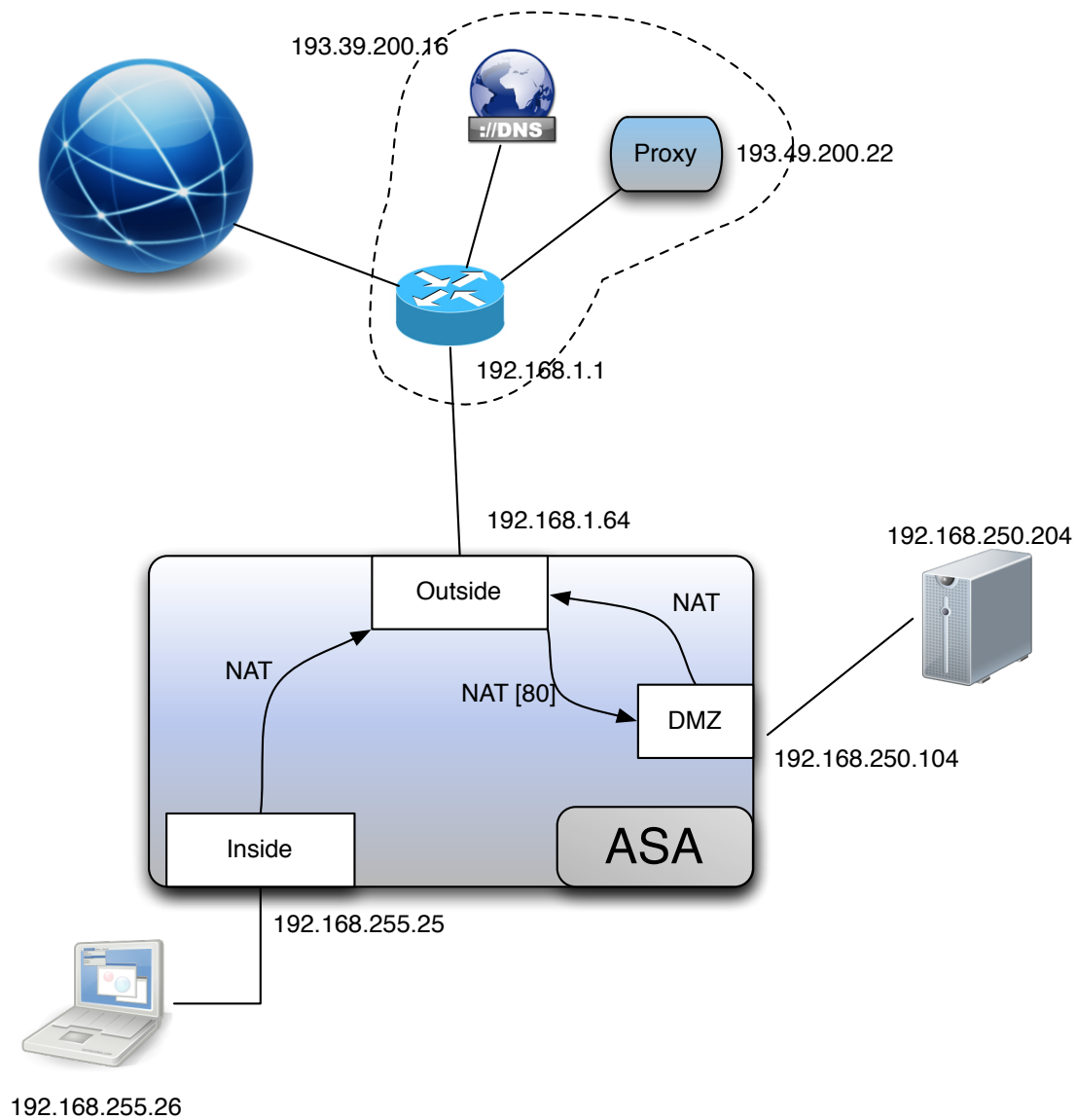


FIGURE 1 – Réseau à obtenir

## 2 Installation et Configurations préliminaires

### 2.1 Client

Dans un premier temps, nous avons installé Ubuntu 9.04 (version client) sur le PC relié à l'interface *inside*. Celle-ci effectuée, nous réalisons les démarches suivantes, c'est à dire mise en place de Java ainsi que l'installation du paquet « Minicom ». Nous lançons ensuite la commande **minicom -s** et définissons les divers paramètres afin de configurer le port console. Puis, nous définissons l'adresse *inside* de l'ASA. Nous pouvons maintenant, à partir de celle-ci, accéder à l'interface d'administration de l'ASA au sein de notre navigateur. La figure ci-dessous présente l'accueil de celle-ci.

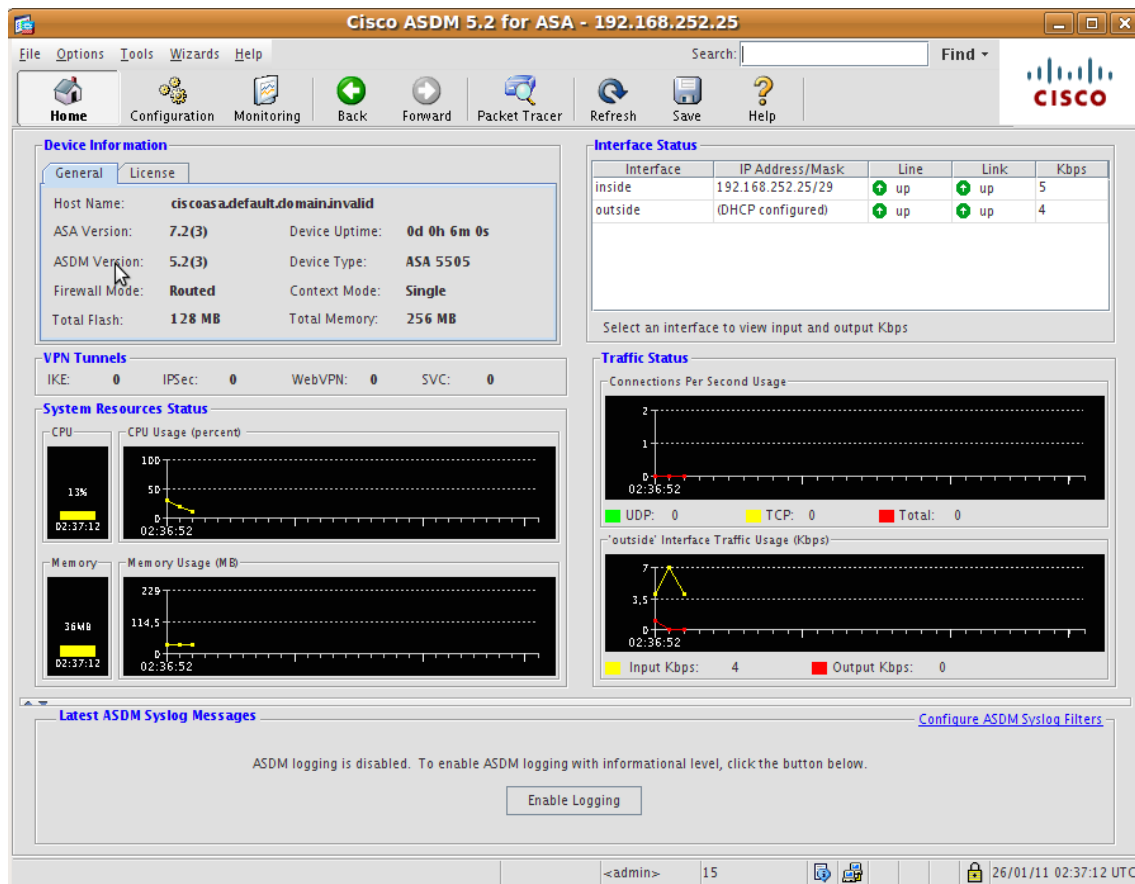


FIGURE 2 – Interface de configuration

Nous avons ensuite utilisé le « Wizard » de l'application pour mettre en place un certain nombre de paramètres tel que adresses IP (inside, outside, dmz) ou encore la répartition des interfaces du firewall (cf. figure ci-dessous).

Name	Switch Ports	Security Level	IP Address	Subnet Mask	VLAN
inside	Ethernet0/1, Ethernet0/2, Ethernet0/3, Ethernet0/4, Ethernet0/5, Ethernet0/6	100	192.168.252.25	255.255.255.248	vlan1
outside	Ethernet0/0	0	192.168.1.64	255.255.255.0	vlan2
dmz	Ethernet0/7	50	192.168.250.104	255.255.255.0	vlan3

FIGURE 3 – Configuration des interfaces

Enfin, nous mettons en place une route statique sur l'interface *outside* afin que

tous les paquets provenant des autres interfaces soit envoyés par défaut à la passerelle de l'ENSICAEN.

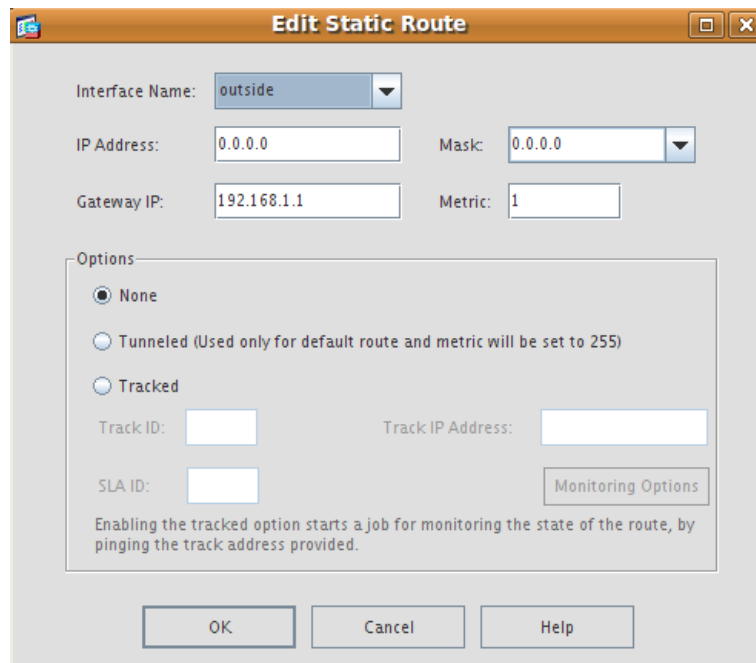


FIGURE 4 – Route statique

## 2.2 Serveur

En parallèle, nous installons « Ubuntu 9.0.4 server » sur le PC relié à l'interface *dmz* du firewall. Lors de l'installation, nous indiquons que nous souhaitons avoir par défaut les services suivants : un serveur SSH et un serveur web (LAMP).

L'installation terminée, nous allons maintenant configurer les informations réseau de notre serveur. Nous renseignons son adresse IP (192.168.250.204), le masque associé et enfin le routeur (ici il s'agit de l'adresse de l'interface *dmz* de notre firewall).

Afin de mettre en place ces informations, nous allons modifier le fichier */etc/network/interfaces* de la sorte :

```
1 auto eth0
2 iface eth0 inet static
3     address 192.168.250.204
4     netmask 255.255.255.0
5     gateway 192.168.250.104
```

### 3 Configuration Inside

Dans cette partie, nous avons configuré notre firewall afin de permettre certaines actions au sous réseau relié à l'interface *inside*.

#### 3.1 Configuration du NAT

Dans un premier temps, il nous a fallu configurer une règle de NAT afin de traduire l'adresse privée de l'interface *inside* en l'adresse publique de l'interface *outside*. Nous devons effectuer cette étape afin de réduire les adresses IP utilisées, d'une part dans le but de ralentir la pénurie d'adresse IPv4, mais aussi pour que la passerelle de l'ENSICAEN c'est qu'une adresse à gérer (celle définie à l'interface *outside*).

Un NAT a pour effet de remplacer les adresses sources des paquets provenant d'un réseau (ou PC) par celle souhaitée (ici remplacement de celles du sous-réseau *inside* par *outside*). Pour les paquets retours (exemple paquet acquitant la réception), le firewall va pouvoir le transmettre au bon destinataire grâce à une sauvegarde de la transaction.

Ci-dessous, la configuration de notre NAT, pour le sous-réseau de lié à notre interface *inside* (192.168.252.24), nous lions l'adresse de l'interface *outside*.

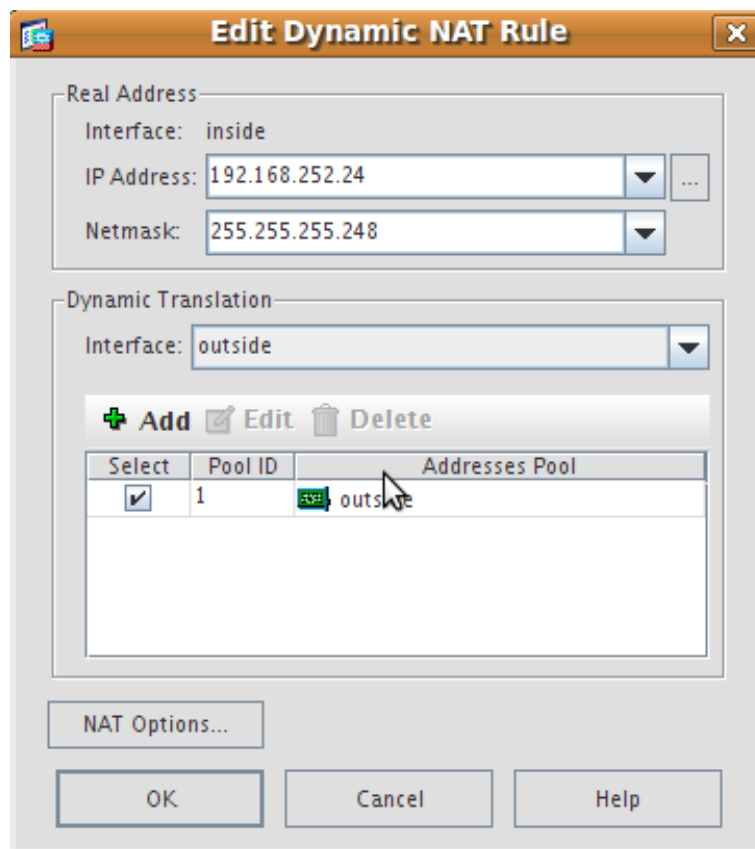


FIGURE 5 – Configuration du NAT

## 3.2 Règles de filtrage

Notre NAT crée, nous allons maintenant mettre en place des règles de filtrage afin de ne laisser passer que les paquets liés à des services définis. Il faut savoir que lorsque des paquets TCP et UDP sont envoyés, une connexion est établie. Cela permet de n'avoir à définir que les règles de sortie, celles d'entrées étant liées. Nous pourrions remarquer que le port source des règles est toujours définis sur « Any », en effet, l'application effectuant la demande n'utilise pas forcément le port dédié.

Chaque règle appliquées ici autorise les services à tout le sous réseau connecté à l'interface *inside*. Il aurait, par exemple, pu être possible de réduire l'accès au service SSH qu'à certaines machines mais nous pensons que ce n'est pas nécessaire dans le cadre de notre TP.

Dans un premier temps, nous autorisons les flux TCP et UDP sur le port 53 (DOMAIN) qui sont à destination de 193.49.200.16 (adresse du serveur DNS de l'ENSICAEN).

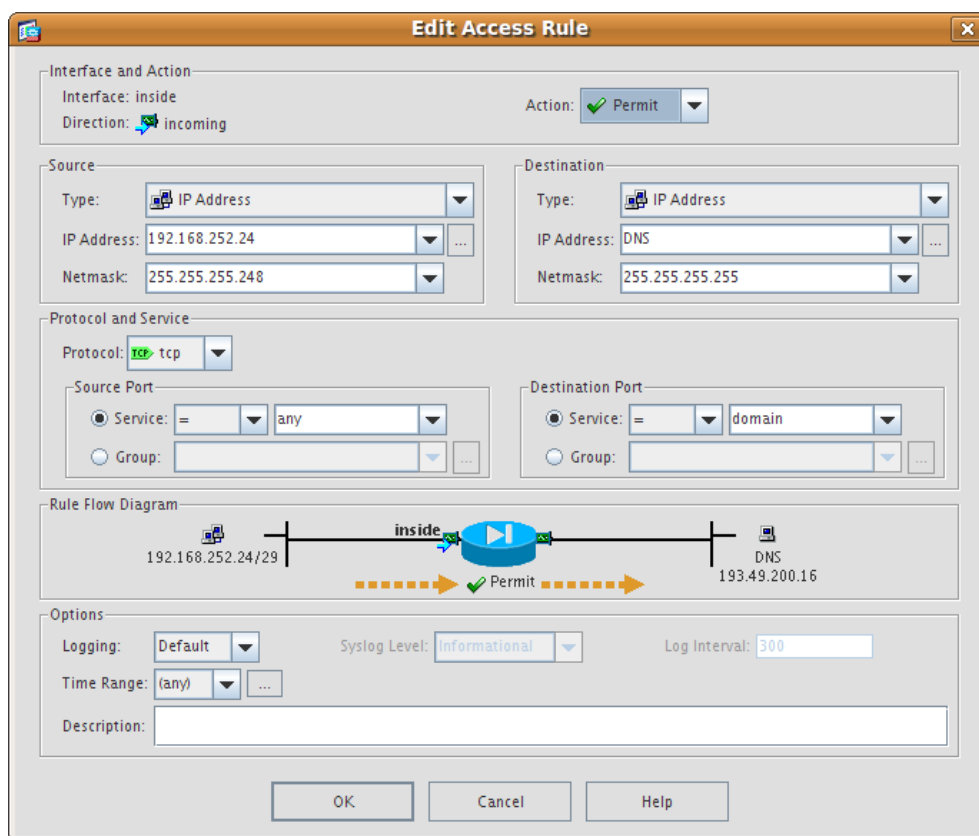


FIGURE 6 – Règle TCP d'accès au DNS


**Edit Access Rule**

Interface and Action  
 Interface: inside  
 Direction: incoming  
 Action: Permit

Source  
 Type: IP Address  
 IP Address: 192.168.252.24  
 Netmask: 255.255.255.248

Destination  
 Type: IP Address  
 IP Address: DNS  
 Netmask: 255.255.255.255

Protocol and Service  
 Protocol: udp  
 Source Port:  
☒ Service: = any  
☐ Group:  
 Destination Port:  
☒ Service: = domain  
☐ Group:

Rule Flow Diagram  


Options  
 Logging: Default  
 Syslog Level: Informational  
 Log Interval: 300  
 Time Range: (any)  
 Description:

OK Cancel Help

FIGURE 7 – Règle UDP d'accès au DNS



Maintenant, nous créons la règle autorisant le flux SSH (TCP sur le port 22). Nous ne nous soucions pas de la cible de la demande.

**Edit Access Rule**

Interface and Action  
Interface: inside  
Direction: incoming  
Action: ☒ Permit

Source  
Type: IP Address  
IP Address: 192.168.252.24  
Netmask: 255.255.255.248

Destination  
Type: any

Protocol and Service  
Protocol: tcp  
Source Port: ☒ Service: = any  
Destination Port: ☒ Service: = ssh

Rule Flow Diagram  
192.168.252.24/29 → inside → any  
Permit

Options  
Logging: Default  
Syslog Level: Informational  
Log Interval: 300  
Time Range: (any)  
Description:

OK Cancel Help

FIGURE 8 – Règle SSH

Puis la règle autorisant le flux HTTP (TCP sur le port 80) à destination de n'importe quelle machine.

**Edit Access Rule**

Interface and Action  
Interface: inside  
Direction: incoming  
Action: ☒ Permit

Source  
Type: IP Address  
IP Address: 192.168.252.24  
Netmask: 255.255.255.248

Destination  
Type: any

Protocol and Service  
Protocol: tcp  
Source Port: ☒ Service: = any  
Destination Port: ☒ Service: = http

Rule Flow Diagram  
192.168.252.24/29 → inside → any  
Dashed arrow: Permit

Options  
Logging: Default  
Syslog Level: Informational  
Log Interval: 300  
Time Range: (any)  
Description:

OK Cancel Help

FIGURE 9 – Règle HTTP

Enfin, nous autorisons le flux à destination d'un proxy (TCP sur le port 3128 = port du proxy de l'école). Bien entendu, nous nous restreignons à l'adresse du proxy de l'ENSICAEN.

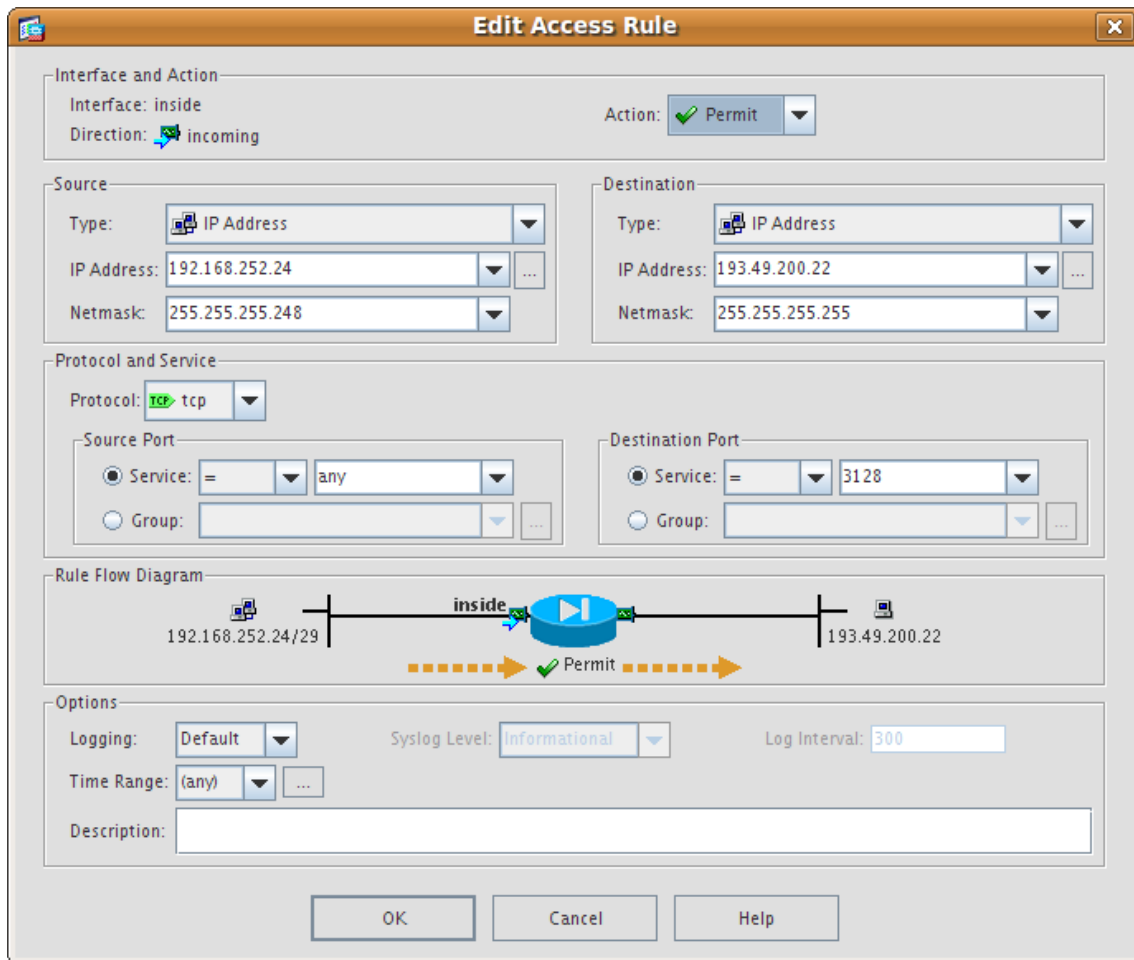


FIGURE 10 – Règle Proxy

## Problèmes rencontrés

### Ping

Nous sommes maintenant censé pouvoir accéder au routeur de l'école (adresse 192.168.1.1). Pour le vérifier, nous lançons la commande **ping** sur son adresse. On remarque que nous n'avons pas de retour de cette commande. Afin de vérifier l'erreur, nous allons regarder le *monitoring* de notre firewall. Ceci va nous permettre de suivre son activité. Après analyse des traces, nous avons pu comprendre l'échec de la commande **ping**. En effet, elles nous informent que les paquets de type ICMP ne sont pas autorisés à destination de l'interface *inside*. Afin de résoudre ce problème, nous devons rajouter une nouvelle règle de filtrage que nous avons défini de la manière ci-dessous.

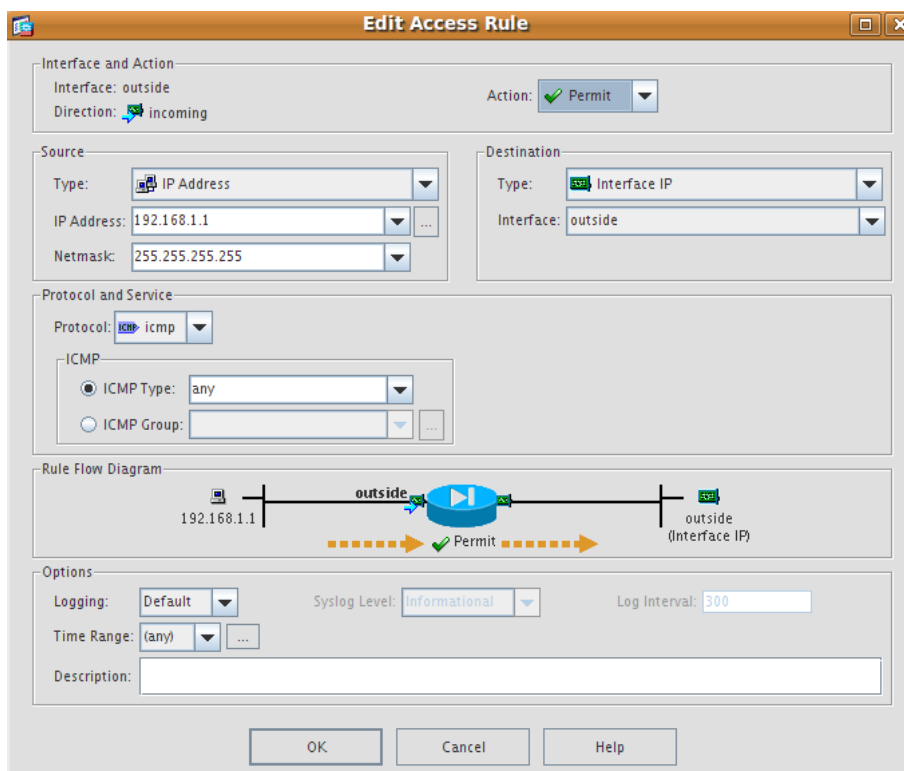


FIGURE 11 – Filtrage ICMP pour autoriser le retour de ping

Cette règle mise en place, nous lançons une nouvelles fois la commande **ping**. Comme visible sur la figure ci-dessous, il n'y a plus d'échec.

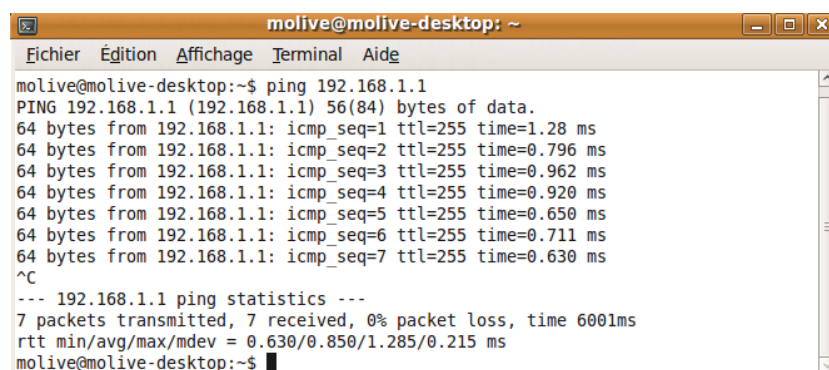


FIGURE 12 – Résultat ping

### Utilisation du DNS

La règle du DNS étant active, nous considérons que l'ordinateur branché sur *inside* pouvait accéder à son service. Pour le vérifier, nous avons essayé plusieurs commandes listées ci-dessous

```
1 ping google.fr
2 host google.fr
3 nslookup google.fr
```

Chacune de ses commandes ne fonctionnaient pas, en effet, elles indiquaient qu'elle n'arrivait pas à récupérer l'adresse IP lié au nom de domaine. Etant donnée que c'est au DNS de nous fournir ces informations, nous avons conclu qu'il y avait un problème de configuration. Pour tester si notre règle de filtrage fonctionnait, nous avons effectué ceci

```
1 telnet 193.49.200.16 53
2 Trying 193.49.200.16...
3 Connected to ns.ecole.ensicaen.fr.
4 Escape character is '^]'.
```

Nous testons la connexion au serveur DNS sur le port 53 (comme définis dans nos règles). Nous remarquons que nous avons pu nous y connecter. (les messages suivants sont dû au fait que nous utilisons **telnet** pour nous connecter). Nos règles sont donc fonctionnelles. Afin de vérifier l'utilisation du DNS, nous avons fait ceci

```
1 nslookup
2 > server 193.49.200.16
3 Default server: 193.49.200.16
4 Address: 193.49.200.16#53
5 > google.fr
6 Server:          193.49.200.16
7 Address:         193.49.200.16#53
8
9 Non-authoritative answer:
10 Name:   google.fr
11 Address: 209.85.229.99
12 Name:   google.fr
13 Address: 209.85.229.147
14 Name:   google.fr
15 Address: 209.85.229.104
```

Dans un premier temps, nous indiquons à **nslookup** l'adresse IP du serveur DNS. Puis, re-testons avec le nom de domaine *google.fr*. Nous pouvons voir qu'il y a un retour, le DNS est donc utilisable. Après recherche, il se trouve que c'est dans le système Linux en lui-même que nous avons oublié d'indiquer l'adresse IP du serveur DNS au moment de rentrer celle de la machine.

## 4 Configuration DMZ

Maintenant que la configuration du PC client est mis en place et fonctionnelle, nous allons configurer notre serveur. Celui-ci doit être accessible de l'extérieur en HTTP et SSH, mais aussi par notre PC client.

### 4.1 Configuration du NAT

Comme pour l'interface *inside*, nous allons définir une NAT afin de traduire les adresses du sous réseau. Etant donné que nous n'avons qu'un serveur de connecté sur l'interface *dmz*, nous donnons son adresse et non celle du sous réseau.

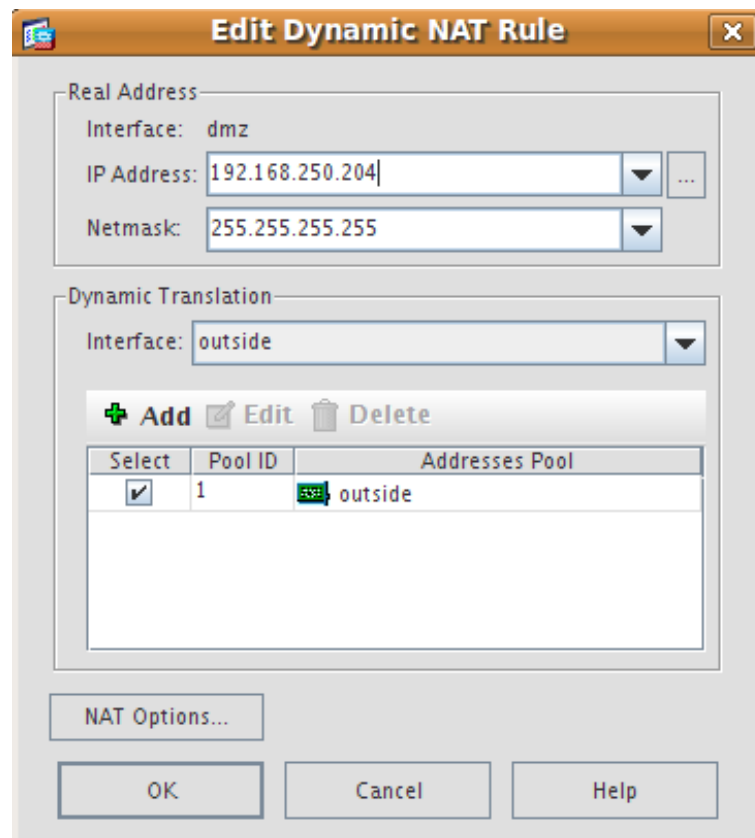


FIGURE 13 – NAT pour l'interface *dmz*

### 4.2 Règles de filtrage

Nous allons aussi autorisé quelques services à notre serveur. Pour toutes nos règles, nous allons limiter l'adresse IP source à celle de notre serveur, en effet, c'est la seule machine du sous-réseau.

Dans un premier temps, l'accès au serveur DNS. Nous autorisons le flux TCP et UDP sur le port 53 spécifiquement pour notre serveur (192.168.250.204).

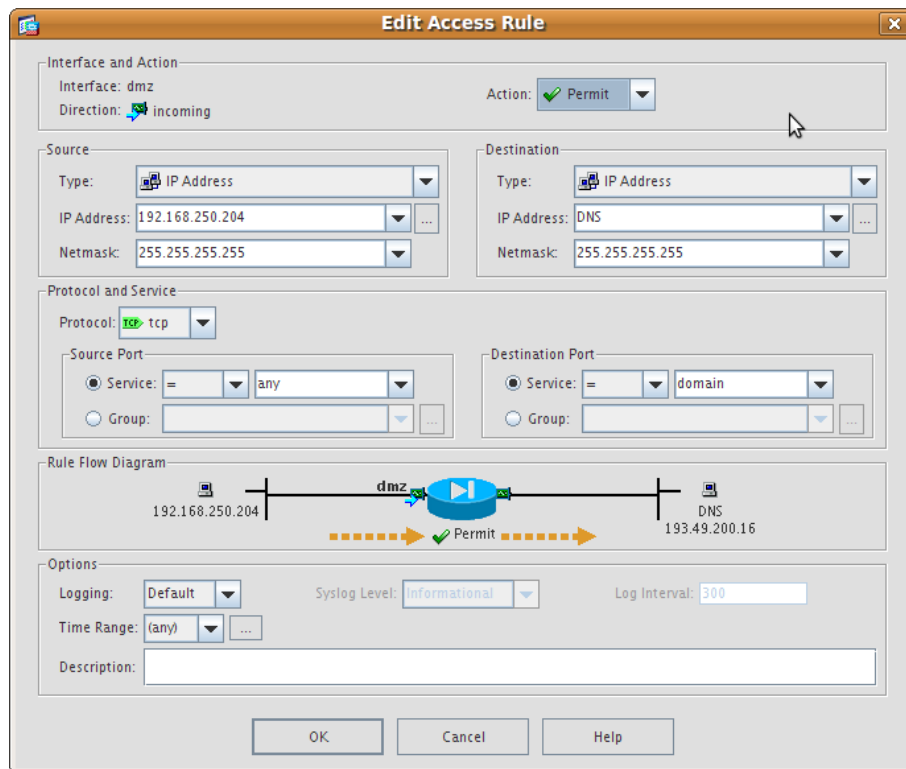


FIGURE 14 – Règle TCP pour l'accès au DNS

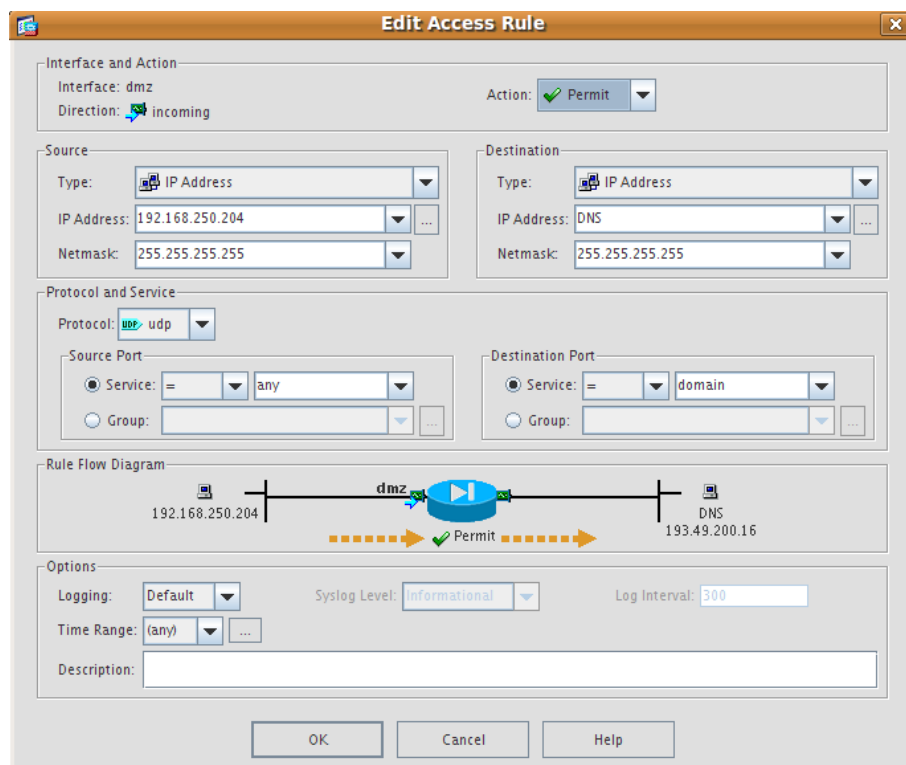


FIGURE 15 – Règle UDP pour l'accès au DNS

Nous devons aussi indiquer au système l'adresse du serveur DNS. Pour ce faire, nous éditons le fichier */etc/resolv.conf* de cette manière.

```
1 nameserver 193.49.200.16
```

Nous autorisons aussi le flux HTTP et SSH en sortie, tout comme pour l'interface *inside*.

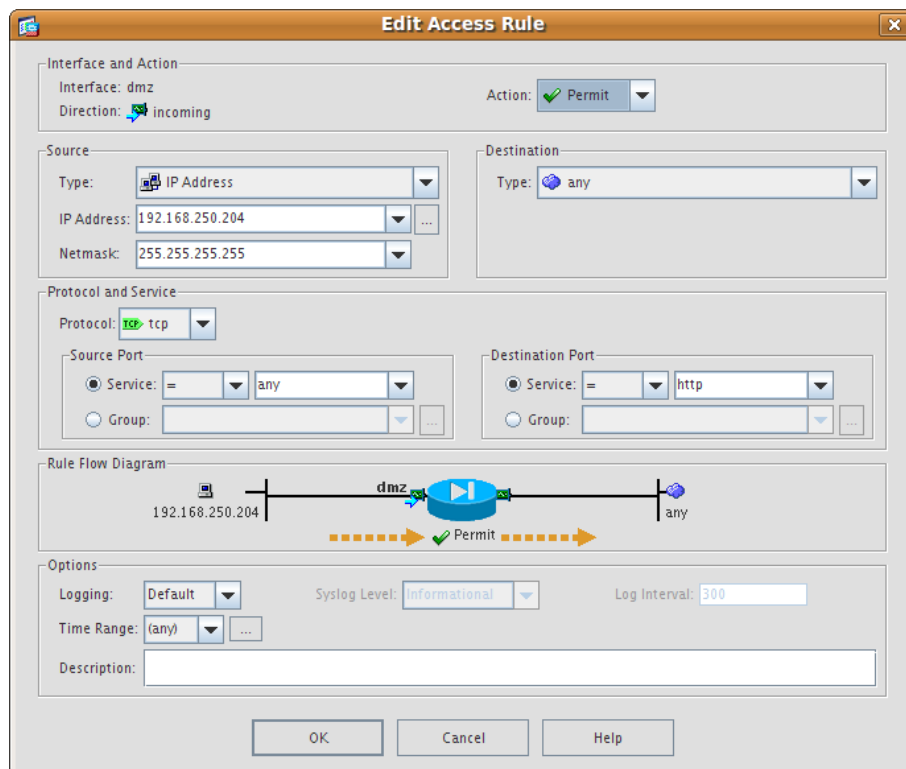


FIGURE 16 – Règle HTTP

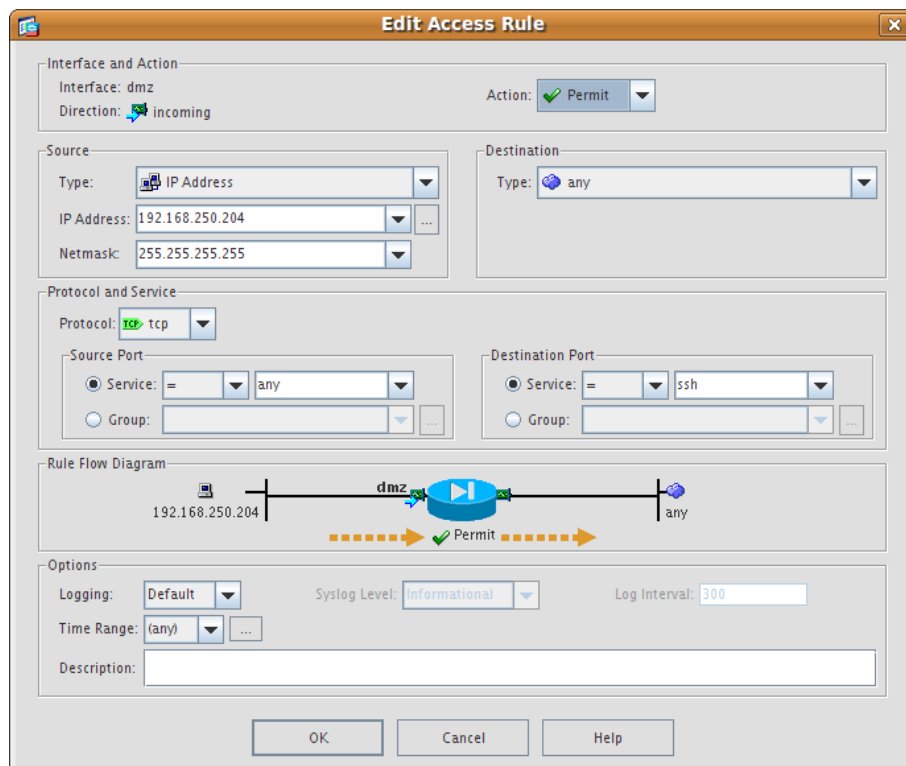


FIGURE 17 – Règle SSH

Nous donnons aussi l'autorisation d'envoyer des paquets en direction du proxy



de l'ENSICAEN.

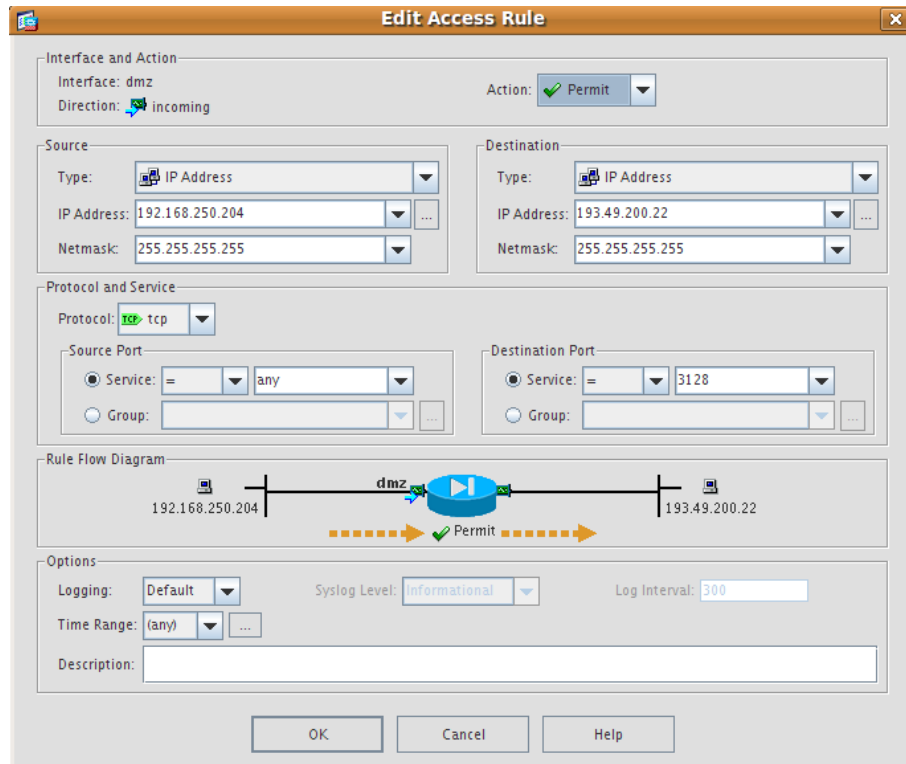


FIGURE 18 – Règle d'accès au proxy

Notre serveur est maintenant capable de discuter avec l'extérieur. Notre but est de pouvoir y accéder à partir de notre client. Pour cela, nous utiliserons la technologie SSH. Avant de tester celle-ci, nous allons voir si notre serveur est accessible. Pour cela, nous allons utiliser la commande **ping** une nouvelle fois à partir de notre pc client. Nous remarquons que cela ne fonctionne pas, nous ne pouvons y accéder. Les log nous indique qu'il y a un problème de translation. Ceci est dû au NAT crée entre l'interface *inside* et *outside*. Afin de résoudre ceci, nous allons y créer une exception indiquant qu'il ne faut pas traduire les messages en provenance du sous-réseau *inside* à destination du serveur (192.168.250.204).

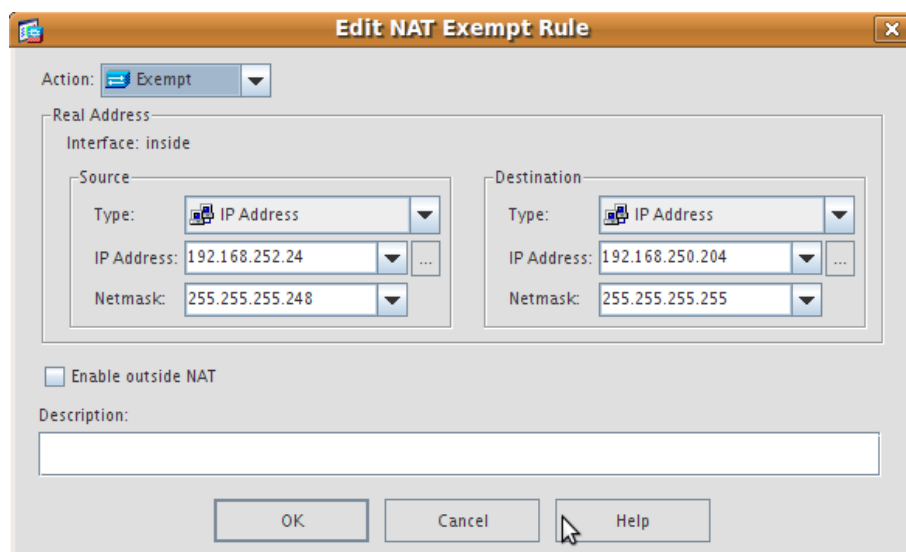


FIGURE 19 – Exception du NAT

### 4.3 HTTP et SSH à partir du réseau ENSICAEN

Nous pouvons maintenant accéder à notre serveur à partir de notre PC que ce soit en SSH ou en HTML. Cependant, cet accès est restreint à l'interface *inside*, en effet, nous souhaiterions que des personnes reliées au serveur de l'ENSICAEN puisse accéder à nos pages web ou encore se connecter en SSH.

Dans un premier temps, nous acceptons les flux HTTP. Nous aurions pu mettre l'adresse IP de notre serveur en tant que destination, mais sachant qu'il n'y a pas de serveur HTTP sur notre PC client, les utilisateurs n'ont aucun intérêt à aller l'interroger.

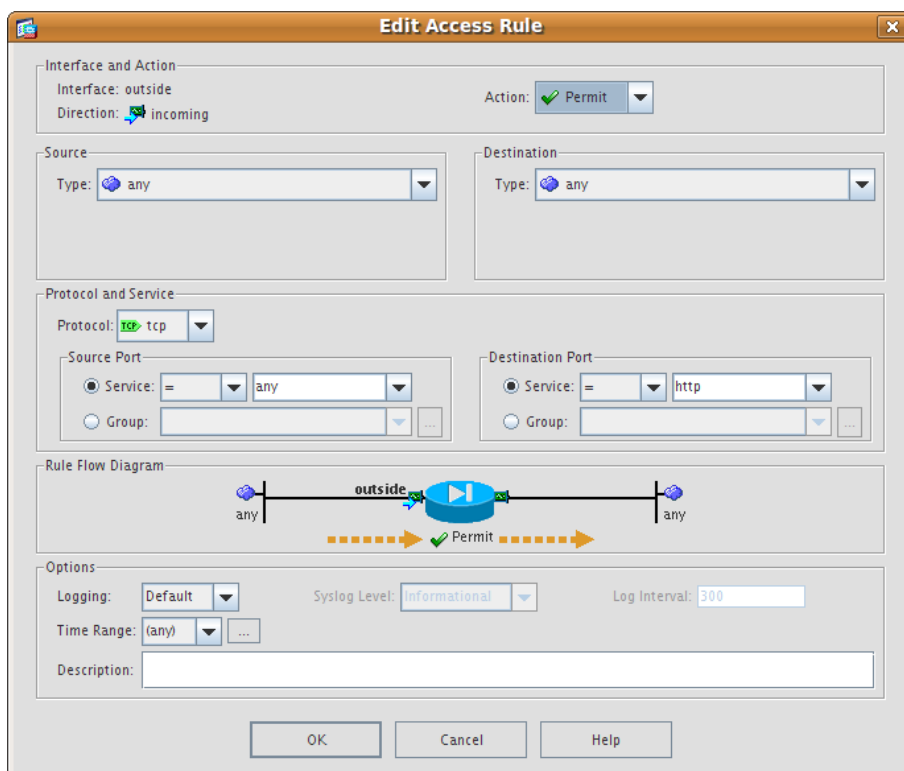


FIGURE 20 – Filtrage HTTP sur outside

Nous configurons maintenant afin qu'il puisse y avoir des demandes de connexion SSH à partir de l'extérieur. Pour plus de sécurité, nous n'avons indiqué que l'IP du serveur. Nous aurions aussi pu ne pas le renseigner, limitant l'accès SSH à notre client *inside*.

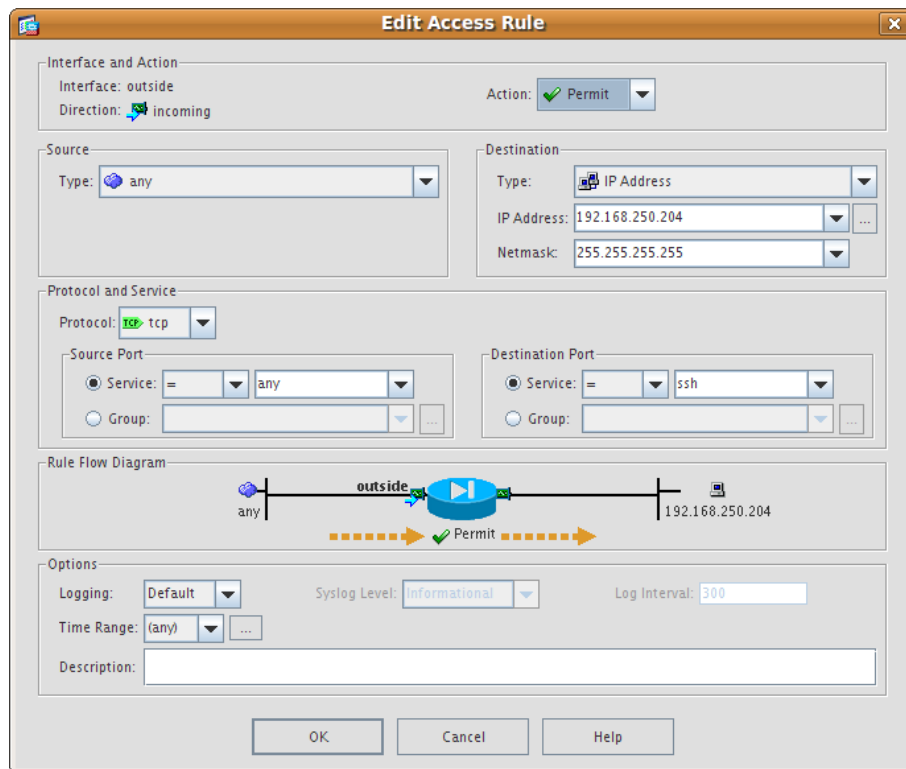


FIGURE 21 – Filtrage SSH

Il nous fallait indiquer au firewall que les flux HTTP et SSH rentrant dans l'interface *dmz* doivent obligatoirement être redirigé à *outside* en utilisant son adresse. Ci-dessous, les deux NAT statiques définis pour effectuer ceci.

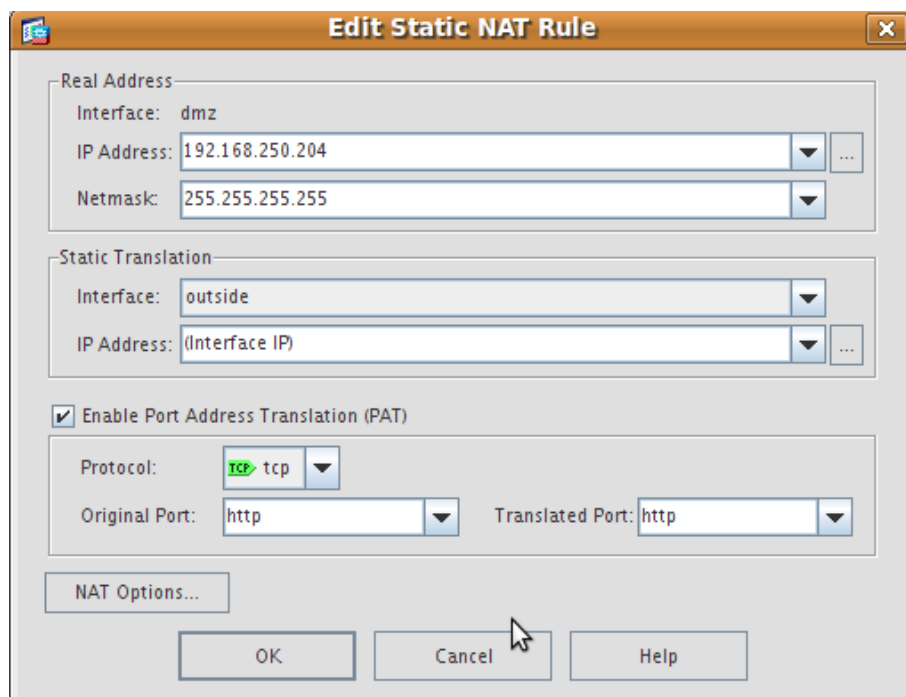


FIGURE 22 – Route statique pour HTTP

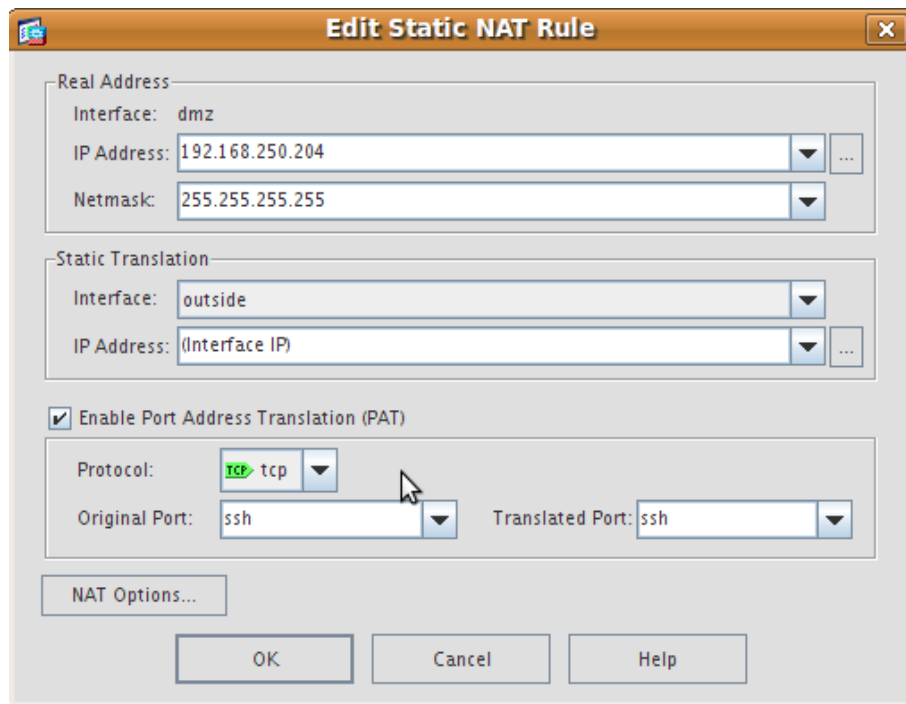


FIGURE 23 – Route statique pour SSH

A partir de ce moment, il était possible aux personnes présentes dans le réseau ENSICAEN d'accéder à notre serveur HTTP mais aussi à se connecter en SSH. Vous pourrez trouver en Annexe notre fichier de configuration.

## 5 Annexe

### 5.1 Configuration firewall

```
1 : Saved
2 :
3 ASA Version 7.2(3)
4 !
5 hostname ciscoasa
6 domain-name ensicaen.fr
7 enable password 8Ry2YjIyt7RRXU24 encrypted
8 names
9 name 193.49.200.16 DNS
10 name 192.168.1.1 Passerelle
11 !
12 interface Vlan1
13     nameif inside
14     security-level 100
15     ip address 192.168.252.25 255.255.255.248
16 !
17 interface Vlan2
18     nameif outside
19     security-level 0
20     ip address 192.168.1.64 255.255.255.0
21 !
22 interface Vlan3
23     no forward interface Vlan1
24     nameif dmz
25     security-level 50
26     ip address 192.168.250.104 255.255.255.0
27 !
28 interface Ethernet0/0
29     switchport access vlan 2
30 !
31 interface Ethernet0/1
32 !
33 interface Ethernet0/2
34 !
35 interface Ethernet0/3
36 !
37 interface Ethernet0/4
38 !
39 interface Ethernet0/5
40 !
41 interface Ethernet0/6
42 !
43 interface Ethernet0/7
44     switchport access vlan 3
45 !
46 passwd 2KFQnbNIdI.2KY0U encrypted
```

```

47 ftp mode passive
48 dns server-group DefaultDNS
49 domain-name ensicaen.fr
50 same-security-traffic permit inter-interface
51 same-security-traffic permit intra-interface
52 access-list outside_access_in extended permit icmp any
   interface outside
53 access-list outside_access_in extended permit tcp any
   host 192.168.250.204 eq ssh
54 access-list outside_access_in extended permit tcp any any
   eq www
55 access-list inside_access_in extended permit udp
   192.168.252.24 255.255.255.248 host DNS eq domain
56 access-list inside_access_in extended permit tcp
   192.168.252.24 255.255.255.248 host DNS eq domain
57 access-list inside_access_in extended permit tcp
   192.168.252.24 255.255.255.248 any eq ssh
58 access-list inside_access_in extended permit tcp
   192.168.252.24 255.255.255.248 any eq www
59 access-list inside_access_in extended permit tcp
   192.168.252.24 255.255.255.248 host 193.49.200.22 eq
   3128
60 access-list inside_access_in extended permit icmp any any
61 access-list dmz_access_in extended permit tcp host
   192.168.250.204 host 193.49.200.22 eq 3128
62 access-list dmz_access_in extended permit udp host
   192.168.250.204 host DNS eq domain
63 access-list dmz_access_in extended permit tcp host
   192.168.250.204 host DNS eq domain
64 access-list dmz_access_in extended permit icmp any any
65 access-list dmz_access_in extended permit tcp host
   192.168.250.204 any eq www
66 access-list dmz_access_in extended permit tcp host
   192.168.250.204 any eq ssh
67 access-list inside_nat0_outbound extended permit ip
   192.168.252.24 255.255.255.248 host 192.168.250.204
68 pager lines 24
69 logging enable
70 logging asdm informational
71 mtu inside 1500
72 mtu outside 1500
73 mtu dmz 1500
74 icmp unreachable rate-limit 1 burst-size 1
75 asdm image disk0:/asdm-523.bin
76 no asdm history enable
77 arp timeout 14400
78 global (outside) 1 interface
79 nat (inside) 0 access-list inside_nat0_outbound
80 nat (inside) 1 192.168.252.24 255.255.255.248
81 nat (dmz) 1 192.168.250.204 255.255.255.255

```

```

82  static (dmz,outside) tcp interface www 192.168.250.204
    www netmask 255.255.255.255
83  static (dmz,dmz) tcp interface ssh 192.168.250.204 ssh
    netmask 255.255.255.255
84  access-group inside_access_in in interface inside
85  access-group outside_access_in in interface outside
86  access-group dmz_access_in in interface dmz
87  route outside 0.0.0.0 0.0.0.0 Passerelle 1
88  timeout xlate 3:00:00
89  timeout conn 1:00:00 half-closed 0:10:00 udp 0:02:00 icmp
    0:00:02
90  timeout sunrpc 0:10:00 h323 0:05:00 h225 1:00:00 mgcp
    0:05:00 mgcp-pat 0:05:00
91  timeout sip 0:30:00 sip_media 0:02:00 sip-invite 0:03:00
    sip-disconnect 0:02:00
92  timeout uauth 0:05:00 absolute
93  http server enable
94  http 192.168.252.24 255.255.255.248 inside
95  no snmp-server location
96  no snmp-server contact
97  snmp-server enable traps snmp authentication linkup
    linkdown coldstart
98  telnet timeout 5
99  ssh timeout 5
100 console timeout 0
101 dhcpd auto_config outside
102 !
103 dhcpd address 192.168.252.26-192.168.252.30 inside
104 !
105
106 !
107 class-map inspection_default
108 match default-inspection-traffic
109 !
110 !
111 policy-map type inspect dns preset_dns_map
112 parameters
113 message-length maximum 512
114 policy-map global_policy
115 class inspection_default
116 inspect dns preset_dns_map
117 inspect ftp
118 inspect h323 h225
119 inspect h323 ras
120 inspect rsh
121 inspect rtsp
122 inspect esmtp
123 inspect sqlnet
124 inspect skinny
125 inspect sunrpc

```

```
126    inspect xdmcp
127    inspect sip
128    inspect netbios
129    inspect tftp
130    !
131    service-policy global_policy global
132    prompt hostname context
133    Cryptochecksum:45edd108b71918d0060196155b01d872
134    : end
135    asdm image disk0:/asdm-523.bin
136    no asdm history enable
```