2017-2018

Mise en place d'un serveur DNS sous Linux

Epreuve E6

Raphaël Andrieu
ESICAD

Table des matières

Cahier des charges

L'établissement Esicad souhaite mettre en place une solution pour faciliter les accès à son intranet, notamment avec la mise en place d'un système de résolution de noms DNS.

Actuellement pour accéder à l'intranet se fait par http://192.168.1.210/intranet/index.html. Après la mise en place du DNS, on souhaiterait y accéder par l'adresse http://www.intranet-esicad.com

Le serveur web Apache sous Debian, hébergeant l'intranet d'adresse 192.168.1.210/24 est dans le réseau serveurs qui a pour adresse réseau 192.168.1.0/24. Pour y accéder il faut être dans le réseau_eleves (192.168.2.0/24) ou dans le réseau_administratif (192.168.3.0/24).

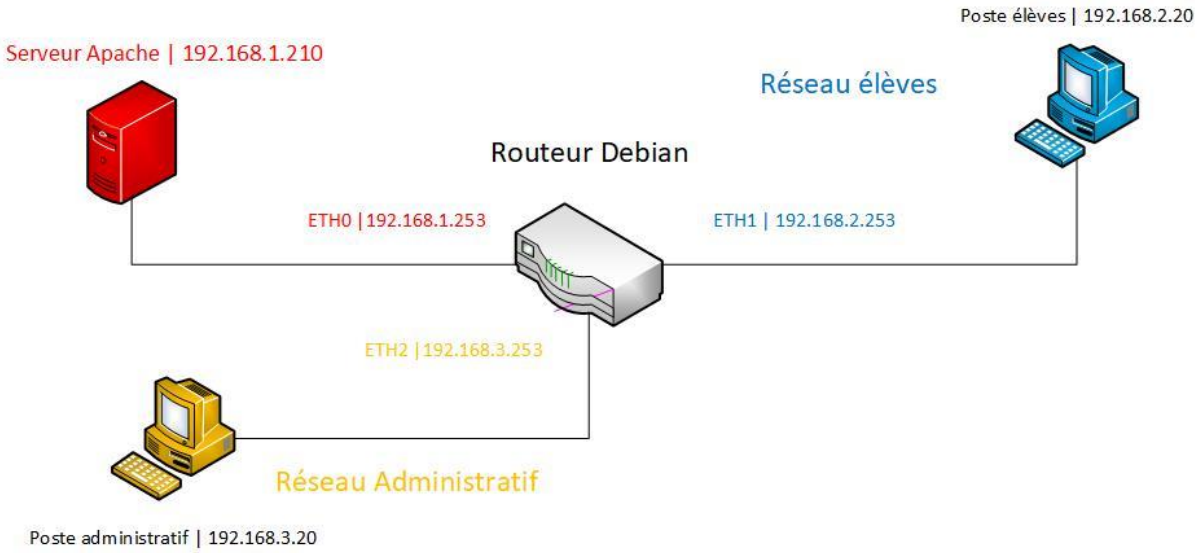
Vous disposez d'ordinateurs sous Linux pour proposer une architecture répondant au cahier des charges.

- ✓ Proposer une maquette de la solution à mettre en œuvre avec les paramètres TCP/IP de tous les composants

- ✓ Mettre en place un système de résolution de noms avec Bind9 et trouver une solution pour remplacer la méthode http://192.168.1.210/intranet/index.html par www.intranet-esicad.com

- ✓ Seul l'accès au service Web est accessible sur le serveur web par tout le monde. Le service FTP installé sur le serveur web n'est qu'accessible que par le réseau administratif.

Maquette



Configurer les interfaces

Nous allons avoir besoin pour répondre à ce cahier des charges de 3 interfaces sur le serveur Linux. Pour cela nous allons configurer ses interfaces.

Ouvrez le terminal, puis passez en mode « root » en écrivant :

- su root

Un mot de passe vous sera demandé.

Après être passé en mode « root » nous allons ajouter les interfaces dans le fichier « interfaces » en écrivant la commande :

- nano /etc/network/interfaces

```
source /etc/network/interfaces.d/*

auto eth0
iface eth0 inet static
        address 192.168.1.253
        netmask 255.255.255.0
        gateway 192.168.1.1
auto eth1
iface eth1 inet static
        address 192.168.2.253
        netmask 255.255.255.0
auto eth2
iface eth2 inet static
        address 192.168.3.253
        netmask 255.255.255.0
```

Nous configurons la première interface en eth0, avec comme adresse : 192.168.1.253, un masque de sous réseau : 255.255.255.0 et une passerelle : 192.168.1.1 (l'adresse du routeur internet)

Puis la deuxième interface en eth1, avec comme adresse : 192.168.2.253 et un masque de sous réseau : 255.255.255.0

Puis la troisième interface en eth2, avec comme adresse : 192.168.3.253 et un masque de sous réseau 255.255.255.0

Nous ne configurons pas de passerelle sur eth1 et eth2 car ces 2 interfaces n'ont pas à envoyer de trafic sur d'autres routeurs

Après avoir fini de configurer le fichier « interfaces », il faut redémarrer le service réseau avec la commande :

- /etc/init.d/networking restart

Pour vérifier que nos paramètres ont bien été pris en compte, il faut écrire la commande :

- Ifconfig

```
root@srv-ra:/home/andri# ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:83:82:2b
          inet adr:192.168.1.253  Bcast:192.168.1.255  Masque:255.255.255.0
          adr inet6: fe80::a00:27ff:fe83:822b/64 Scope:Lien
          adr inet6: 2a02:8435:6b0:7501:a00:27ff:fe83:822b/64 Scope:Global
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:365 errors:0 dropped:0 overruns:0 frame:0
          TX packets:183 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 lg file transmission:1000
          RX bytes:43833 (42.8 KiB)  TX bytes:24188 (23.6 KiB)

eth1      Link encap:Ethernet  HWaddr 08:00:27:62:e6:0c
          inet adr:192.168.2.253  Bcast:192.168.2.255  Masque:255.255.255.0
          adr inet6: fe80::a00:27ff:fe62:e60c/64 Scope:Lien
          adr inet6: 2a02:8435:6b0:7501:a00:27ff:fe62:e60c/64 Scope:Global
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:245 errors:0 dropped:6 overruns:0 frame:0
          TX packets:112 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 lg file transmission:1000
          RX bytes:27194 (26.5 KiB)  TX bytes:14823 (14.4 KiB)

eth2      Link encap:Ethernet  HWaddr 08:00:27:7a:ed:00
          inet adr:192.168.3.253  Bcast:192.168.3.255  Masque:255.255.255.0
          adr inet6: 2a02:8435:6b0:7501:a00:27ff:fe7a:ed00/64 Scope:Global
          adr inet6: fe80::a00:27ff:fe7a:ed00/64 Scope:Lien
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:230 errors:0 dropped:6 overruns:0 frame:0
          TX packets:115 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 lg file transmission:1000
          RX bytes:25084 (24.4 KiB)  TX bytes:14691 (14.3 KiB)
```

Vérifiez que chaque interface a bien la bonne configuration.

Activation du routeur

Pour relier les différents sous-réseaux nous avons besoin d'un routeur. Ici nous utiliserons ce serveur Linux comme routeur.

Après avoir configuré les interfaces, nous pouvons mettre en place le mode routeur sur notre Linux, pour cela il faut modifier le fichier sysctl.conf en écrivant la commande :

- nano /etc/sysctl.conf

Dans ce fichier, il faut décommenter la ligne :

- net.ipv4.ip_forward=1

```
Fichier  Edition  Affichage  Rechercher  Terminal  Aide
 GNU nano 2.2.6          Fichier : /etc/sysctl.conf

#net.ipv4.tcp_syncookies=1

# Uncomment the next line to enable packet forwarding for IPv4
net.ipv4.ip_forward=1
```

Après avoir modifié ce fichier, nous pouvons recharger le fichier en écrivant la commande :

- sysctl -p /etc/sysctl.conf

Activation du NAT

Pour utiliser notre routeur Linux comme passerelle internet pour tous les sous-réseaux, nous devons y implémenter la translation d'adresses, le NAT. Cela permettra aux adresses privées des réseaux locaux d'accéder à Internet.

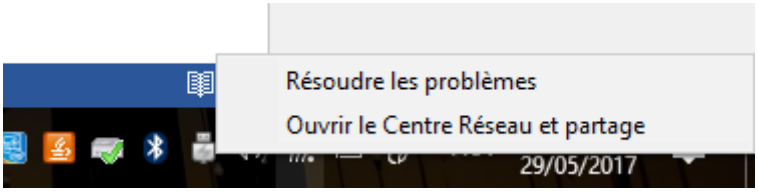
Nous allons donc préciser que eth0 est notre interface qui communiquera avec l'extérieur. Pour activer le nat sur l'interface eth0, il faut écrire la commande :

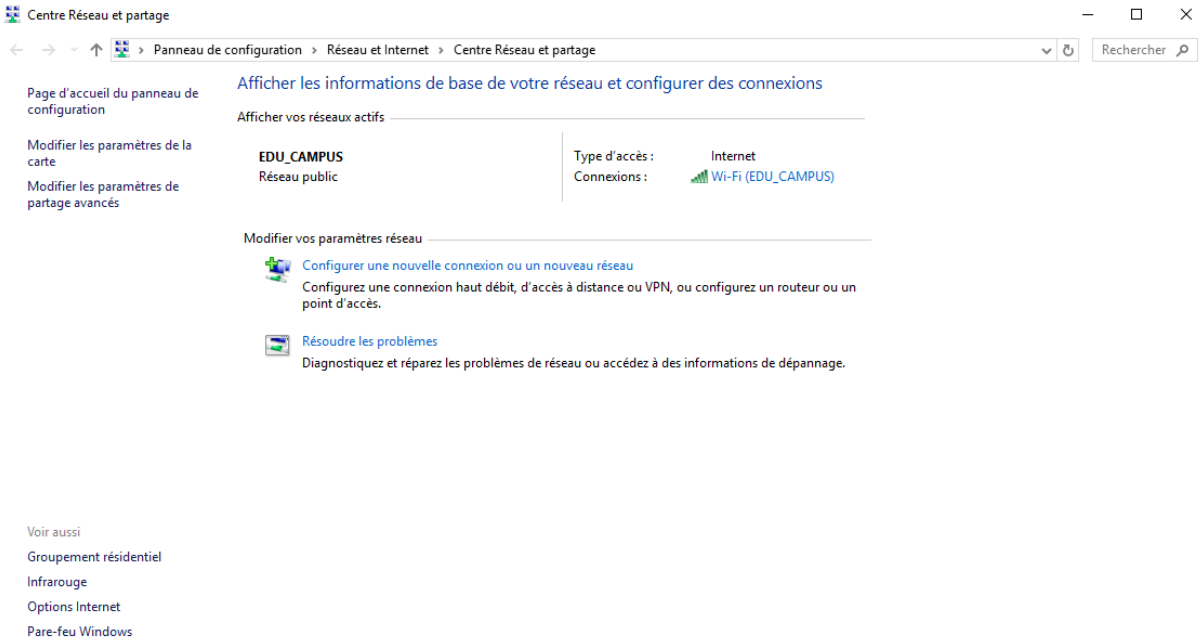
- iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE

Tester la communication entre les réseaux

Après avoir configuré les interfaces, activé le routeur et activé le NAT, nous allons « essayer de pinger » les réseaux entre eux. Pour cela nous allons mettre en place un ordinateur dans le réseau 192.168.3.0 qui sera notre réseau pour le service administratif, un ordinateur dans le réseau : 192.168.2.0 pour le réseau élève et un serveur sous Linux dans le réseau 192.168.1.0 qui nous servira par la suite de serveur WEB..

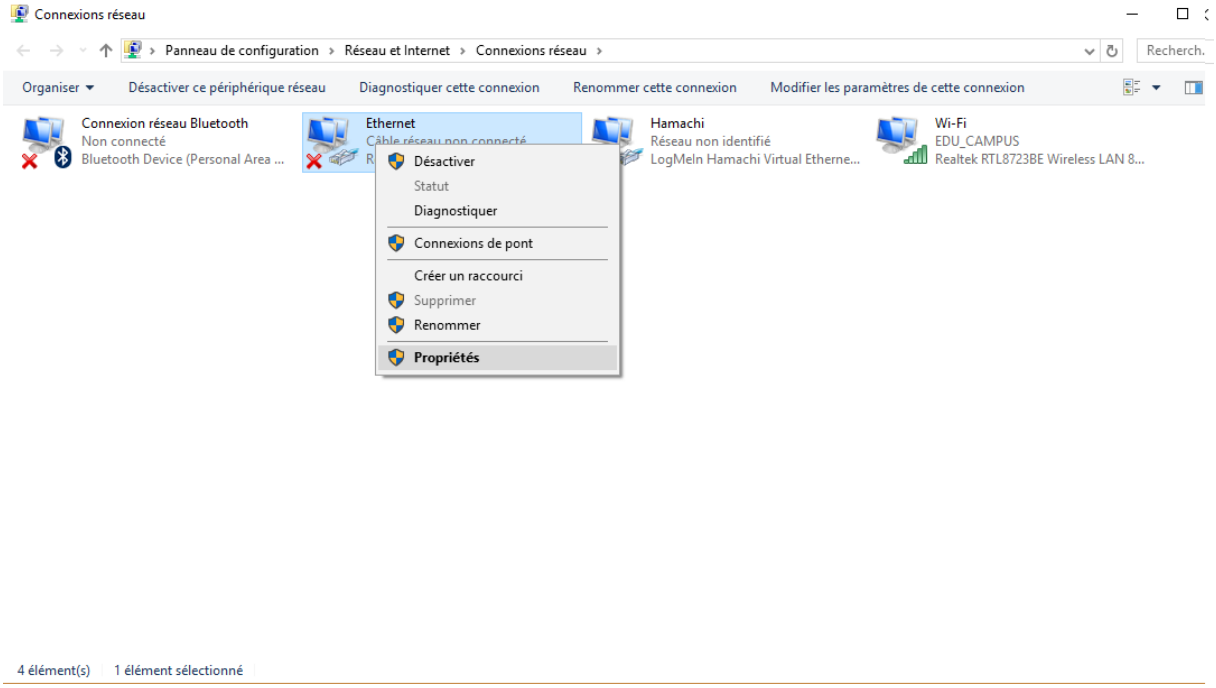
Pour configurer le poste sous Windows du réseau administratif. Il vous faut ouvrir le centre de réseau et partage en faisant un clic droit sur internet dans la barre des tâches.



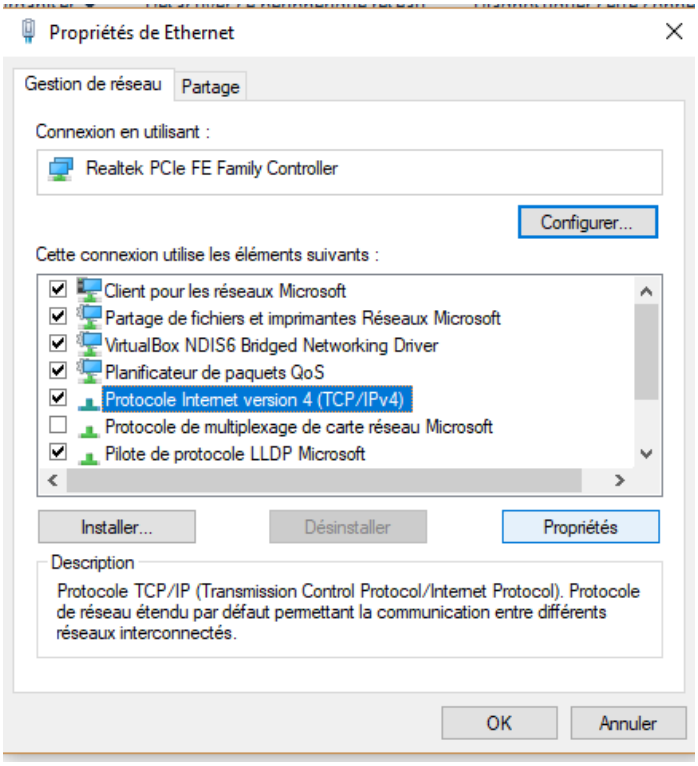


Puis sur « modifier les paramètres de la carte » sur la gauche.

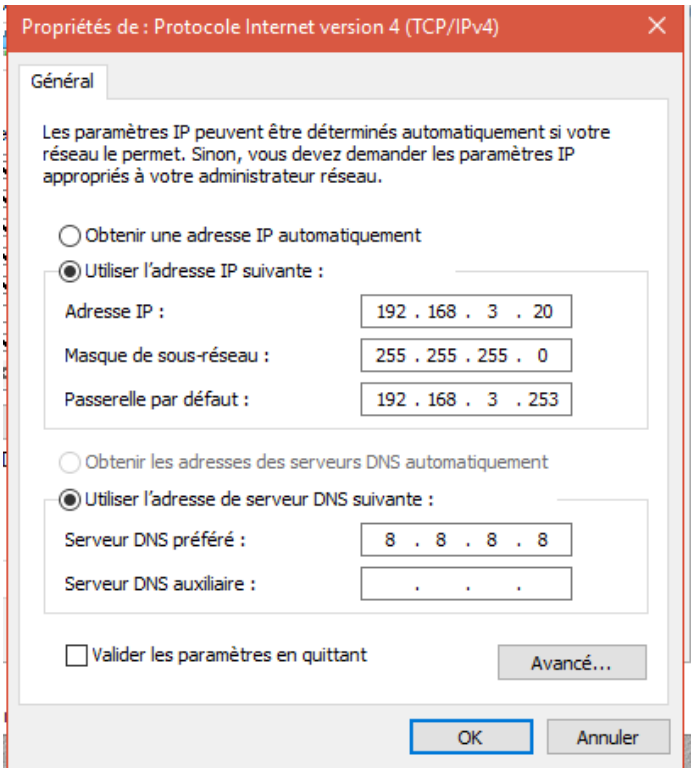
Puis faire un clic-droit > propriété sur la carte réseau.



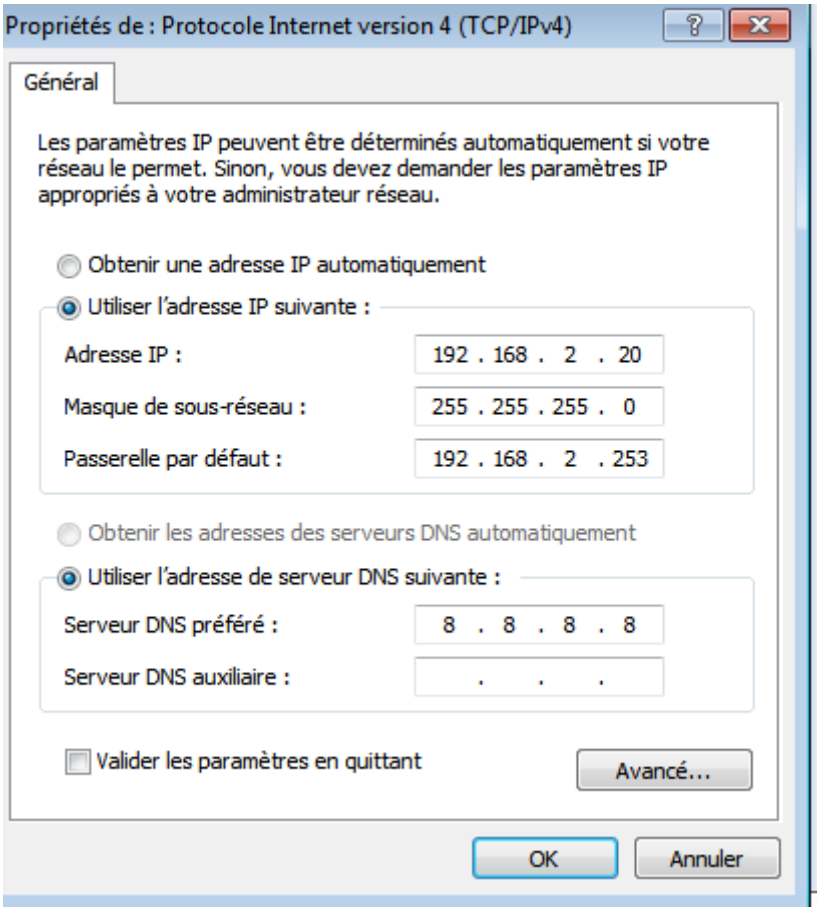
Puis sur « protocole internet version 4 (TCP/IVP4) » et sur propriétés



Puis remplissez les champs Adresse ip avec une adresse sur le réseau 192.168.3.0, un masque de sous réseau en 255.255.255.0 et mettre en passerelle, l'adresse ip de l'interface eth2.



Faire la même chose pour l'ordinateur sur le réseau de l'interface eth1, et mettre en passerelle l'adresse de l'interface eth1



Faire la même chose sur le serveur Web linux en configurant le fichier : /etc/network/interfaces

```
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto eth0
iface eth0 inet static
        address 192.168.1.210
        netmask 255.255.255.0
        gateway 192.168.1.253
```

Avec une adresse sur le réseau : 192.168.1.210 et avec comme passerelle l'adresse ip de eth0, ici : 192.168.1.253

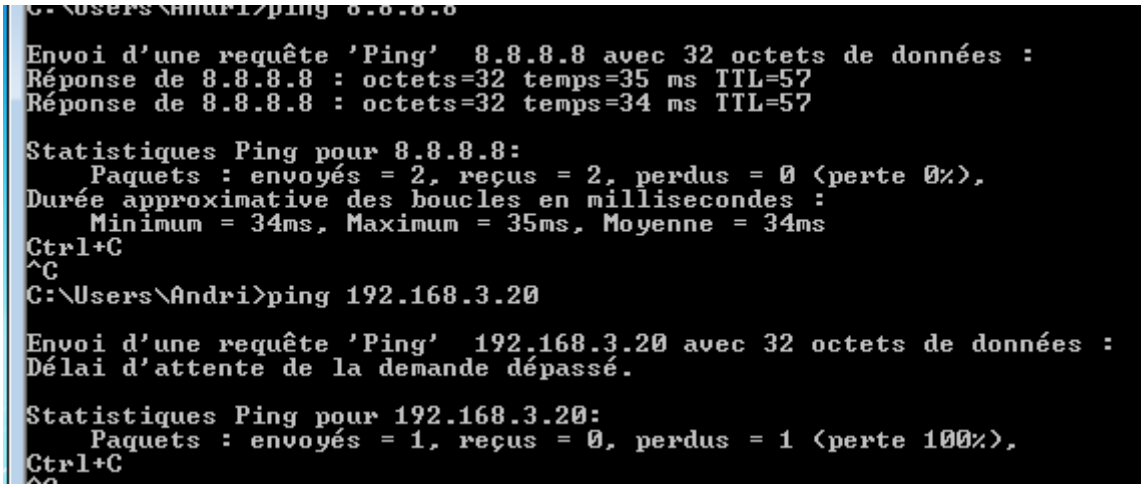
Maintenant que les cartes réseaux sont bien configuré sur chacun des postes, nous pouvons effectuer des tests pour voir si les postes sont bien connectés à internet.

Sur le poste du réseau Administratif :

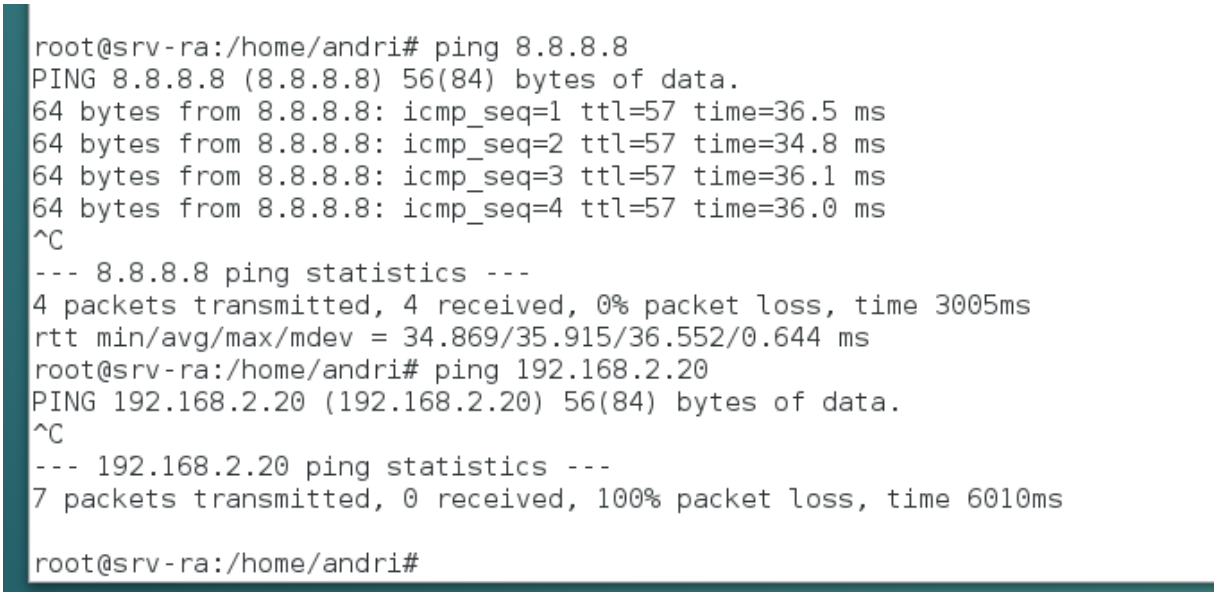
```
Statistiques Ping pour 8.8.8.8:
    Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
Durée approximative des boucles en millisecondes :
    Minimum = 34ms, Maximum = 35ms, Moyenne = 34ms

C:\Users\Andri>ping 192.168.2.20

Envoi d'une requête 'Ping'  192.168.2.20 avec 32 octets de données :
Délai d'attente de la demande dépassé.
Délai d'attente de la demande dépassé.
Délai d'attente de la demande dépassé.
Délai d'attente de la demande dépassé.

Statistiques Ping pour 192.168.2.20:
    Paquets : envoyés = 4, reçus = 0, perdus = 4 (perte 100%),

C:\Users\Andri>
```

Nous pouvons voir qu'il accède bien à internet car il peut ping l'adresse de Google 8.8.8.8, et il ne peut par contre pas ping le poste du réseau élève.

Pareil pour le poste élève :



Et sur le serveur Linux :



Installation du service APACHE et FTP sur le serveur LINUX

Installation de Apache

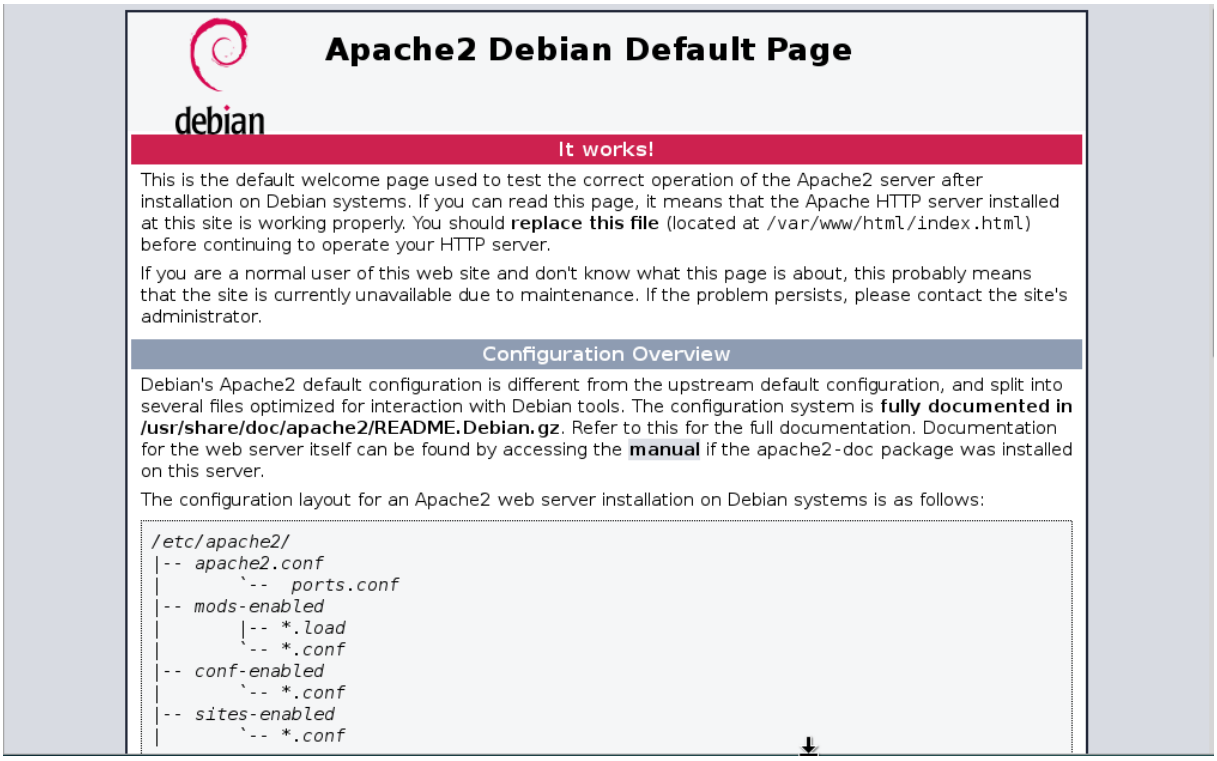
Nous allons mettre en place un serveur WEB sur le serveur linux, pour cela nous allons installer le plugin apache en écrivant la commande :

- apt-get install apache2

N'hésitez pas avant d'écrire cette commande, de mettre à jour les paquets avec la commande :

- apt-get update

Après avoir installé le module apache, vous pouvez essayer d'accéder au site intranet en rentrant l'adresse ip du serveur linux sur l'ordinateur d'un autre poste.



Installation de vsftpd

Nous allons installer le module SSH en écrivant la commande :

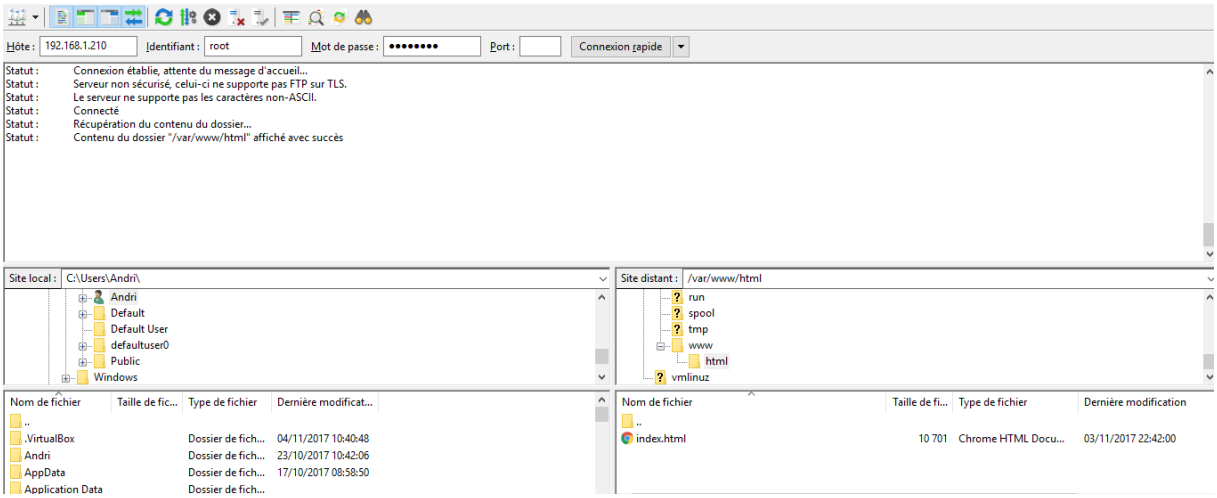
- apt-get install vsftpd

Nous allons maintenant rediriger la connexion vers la racine du site web en modifiant la config de vstpd qui se trouve en /etc/vsftpd.conf.

```
#
# The default compiled in settings are fairly paranoid. Th
# loosens things up a bit, to make the ftp daemon more usa
# Please see vsftpd.conf.5 for all compiled in defaults.
#
# READ THIS: This example file is NOT an exhaustive list o
# Please read the vsftpd.conf.5 manual page to get a full
# capabilities.

local_root=/var/www/html
#
#
# Run standalone?  vsftpd can run either from an inetd or
# daemon started from an initscript.
listen=NO
#
```

Pour redirigé le FTP vers la racine www, il faut rajouter la ligner « local_root=/var/www/html » dans le fichier de configuration. Décommenter aussi la ligne « write_enable=YES » pour autoriser l'accès en écriture.

Maintenant que Vsftpd est installé et configuré, vous pouvez essayer de vous connecter avec un des postes avec un programme qui permet l'accès en FTP, pour ce cas, j'utilise le logiciel « FileZilla » qui est disponible gratuitement sur internet.



Vous pouvez maintenant ajouter les pages que vous voulez sur le site internet grace a cette utilitaire.

Configuration avant l'installation du DNS

Avant l'installation de Bind9, nous allons configurer le nom de notre serveur, vous pouvez vérifier le nom de votre serveur en écrivant « hostname »

Pour changer le nom de notre serveur, il faut modifier le fichier « /etc/hostname » et puis recharger la config avec « /etc/init.d/hostname.sh start »

Ici j'ai choisi d'appeler mon serveur « srv-ra-dns »

```
root@srv-ra-dns:/home/andri# hostname
srv-ra-dns
root@srv-ra-dns:/home/andri#
```

Modifiez aussi le nom du serveur Apache, pour ma part je l'ai modifié en « srv-ra-apache »

```
root@srv-ra-apache:/var/www# hostname
srv-ra-apache
root@srv-ra-apache:/var/www#
```

Puis nous allons modifier le fichier « /etc/host.conf » pour que le serveur DNS soit utilisable à la fois pour le client et pour le serveur. Ajoutez les lignes :

- Order hosts, bind
- Multi on

```
  GNU nano 2.2.6          Fichier : /etc/host.conf

order hosts, bind
multi on
```

Maintenant nous allons compléter le fichier « /etc/hosts » du Serveur DNS, pour renseigner le nom des clients du réseau local et le nom de domaine.

```
                                    andri@srv-ra-dns: ~
 Fichier  Édition  Affichage  Rechercher  Terminal  Aide
  GNU nano 2.2.6          Fichier : /etc/hosts

127.0.0.1       localhost.intranet-esciad.com   localhost
192.168.1.210   srv-ra-apache
192.168.1.253   srv-ra-dns.intranet-esicad.com  src-ra-dns

# The following lines are desirable for IPv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
```

J'ai ajouté une ligne pour l'adresse 192.168.1.210 qui est l'adresse de notre serveur web, et je lui ai assigné son nom que nous avons configuré précédemment. Puis une ligne pour 192.168.1.253 qui est l'une des adresses de nos interfaces, qui sera l'adresse de notre DNS.

Puis nous allons modifier le fichier « /etc/resolv.conf » sur les 2 serveurs (srv-ra-apache et srv-ra-dns)

```
  GNU nano 2.2.6          Fichier :

# Generated by NetworkManager
domain intranet-esicad.com
search intranet-esicad.com
nameserver 192.168.1.253
```

La ligne domain contient le nom du domaine, ici « intranet-esicad.com » puis la ligne search et nameserver contient l'adresse du DNS, ici l'interface eth0.

Installation et configuration d'un serveur DNS sous Bind9

- Apt-get update
- Apt-get install bind9

Puis nous allons nous positionner sur le dossier Bind en faisant « cd /etc/bind/ » puis vous pouvez voir ce que contient ce dossier en écrivant « ls -l »

Nous allons maintenant configuré notre serveur, pour cela nous allons copier un fichier modèle « cp db.local db.intranet-esicad.com » puis nous allons l'éditer « nano db.intranet-esicad.com »

```
  GNU nano 2.2.6               Fichier : /etc/bind/db.intranet-esicad.com

;
; BIND data file for local loopback interface
;
$TTL    604800
@       IN      SOA     srv-ra-dns.intranet-esicad.com. root.intranet-esicad.com. (
                              2         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
;
@       IN      NS      srv-ra-dns.intranet-esicad.com.
srv-ra-dns      A       192.168.1.253
srv-ra-apache   A       192.168.1.210

www     CNAME   srv-ra-apache
```

Explication de la configuration :

-$TTL 604800 : défini en secondes le délai maximum pendant lequel un enregistrement pourra être gardé en cache.

-Le nom du serveur est défini avec la ligne « @ IN SOA srv-ra-dns.intranet-esicad.com. root.intranet-esicad.com. » Cette ligne contient le nom du serveur, puis l'adresse électronique de l'administrateur. Puis les parenthèses contiennent des informations, comme un numéro de série et des délais exprimés en seconde qui vont piloter le comportement des serveurs esclaves.

-Une ligne permet de faire correspondre les adresses ip des serveurs avec leur nom « srv-ra-dns A 192.168.1.253 » et « srv-ra-apache A 192.168.1.210 »

-Une ligne permet de mettre en place un alias depuis l'enregistrement de type A de « srv-ra-apache » en utilisant comme nom « www » avec la ligne « www CNAME srv-ra-apache »

-Le serveur qui doit répondre a zone est défini avec la ligne « @ IN NS srv-ra-dns.intranet-esicad.com. »

Maintenant que nous avons configuré ce fichier, nous allons configurer la recherche inverse du domaine, pour permettre au client d'utiliser l'adresse du serveur pour obtenir son nom de domaine. Pour cela nous allons copier la configuration de base « cp db.127 db.intranet-esicad.com.inv » et modifier cette configuration « nano db.intranet-esicad.com.inv »

```
;
; BIND reverse data file for local loopback interface
;
$TTL    604800
@       IN      SOA     srv-ra-dns.intranet-esicad.com. root.intranet-esicad.com. (
                              1         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
;
@       IN      NS      srv-ra-dns.
@       IN      A       192.168.1.253
253     IN      PTR     srv-ra-dns.intranet-esicad.com.
210     IN      PTR     srv-ra-apache.intranet-esicad.com.
```

Explication de la configuration :

Cette configuration est très similaire au fichier précèdent, les changements viennent au niveau des enregistrement DNS :

- L'adresse et le nom du serveur est défini avec « @ IN NS srv-ra-dns. » et « @ IN A 192.168.1.253 »
- Et les adresses du serveur apache et DNS, en utilisant que le dernier octet de l'adresse, soit 253 pour le serveur DNS et 210 pour le serveur apache.

Après avoir créé et modifié ces 2 deux fichiers, nous allons modifié le fichier « named.conf.local »

```
 GNU nano 2.2.6                 Fichier : /etc/bind/named.conf.local

//
// Do any local configuration here
//

// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";

zone "intranet-esicad.com" {
        type master;
        file "/etc/bind/db.intranet-esicad.com";

};
zone "1.168.192.in-addr.arpa" {
        type master;
        file "/etc/bind/db.intranet-esicad.com.inv";
};
```

Explication de la configuration :

- Une zone DNS est définie avec comme nom « intranet-esicad.com » nous avons défini le chemin du fichier configuré précédemment « db.intranet-esicad.com »
- Puis une zone de recherche inversée « 1.168.192.in-addr.arpa » avec le chemin du fichier crée précédemment « db.intranet-esicad.com.inv »

Modifiez le fichier « named.conf.options » et ajoutez dans la partie « forwarders » les adresses « 192.168.1.253 » « 8.8.8.8 » et « 8.8.4.4 »

```
// nameservers, you probably want to use them as forwarder
// Uncomment the following block, and insert the addresses
// the all-0's placeholder.

forwarders {
        192.168.1.253;
        8.8.8.8;
        8.8.4.4;
 };
```

Nous avons donc ajouté les forwarders de google et l'adresse du serveur DNS.

Vous pouvez maintenant redémarrer le service Bind9 avec :

- Service bind9 restart ou /etc/init.d/bind9 restart

Nous allons vérifier la validité des fichiers de zones avec la commande

- named-checkzone intranet-esicad.com /etc/bind/db.intranet-esicad.com
- named-checkzone intranet-esicad.com /etc/bind/db.intranet-esicad.com.inv

```
root@srv-ra-dns:/home/andri# named-checkzone intranet-esicad.com /etc/bind/db.i
ntranet-esicad.com
zone intranet-esicad.com/IN: loaded serial 2
OK
root@srv-ra-dns:/home/andri#
```

Nous allons aussi vérifier que le serveur DNS se connaisse lui-même avec la commande « nslookup 192.168.1.253 »

```
Server:         192.168.1.253
Address:        192.168.1.253#53

253.1.168.192.in-addr.arpa      name = srv-ra-dns.intranet-esicad.com.

root@srv-ra-dns:/home/andri#
```

Vous pouvez voir que la commande répond «253.1.168.192.in-addr.arpa name = srv-ra-dns.intranet-esicad.com. » ce qui veut dire que la résolution ce fait bien entre l'adresse IP et le nom de domaine.

Et tester nslookup avec l'adresse du site web www.intranet-esicad.com

```
OK
root@srv-ra-dns:/home/andri# nslookup www.intranet-esicad.com
Server:         192.168.1.253
Address:        192.168.1.253#53

www.intranet-esicad.com canonical name = srv-ra-apache.intranet-esicad.com.
Name:   srv-ra-apache.intranet-esicad.com
Address: 192.168.1.210

root@srv-ra-dns:/home/andri#
```

Vous pouvez voir que la commande vous retourne bien l'adresse de votre serveur apache.

VirtualHost

Pour remplacer la méthode http://www.intranet-esicad.com/intranet/index.html par www.intranet-esicad.com, il faut créer un hote virtuel, le serveur Apache2 est capable de gérer simultanément plusieurs arborescences Web grace a ces Hotes. Nous allons voir comment créer un Hote pour permettre d'avoir accès a l'index.html du dossier intranet, directement a la racine en utilisant l'adresse www-intranet-esicad.com.

Pour commencer, copié la configuration de base d'apache2 content les virtualhosts en faisant : « cd /etc/apache2/sites-available » puis « cp 000-default.conf intranet-esicad.conf »

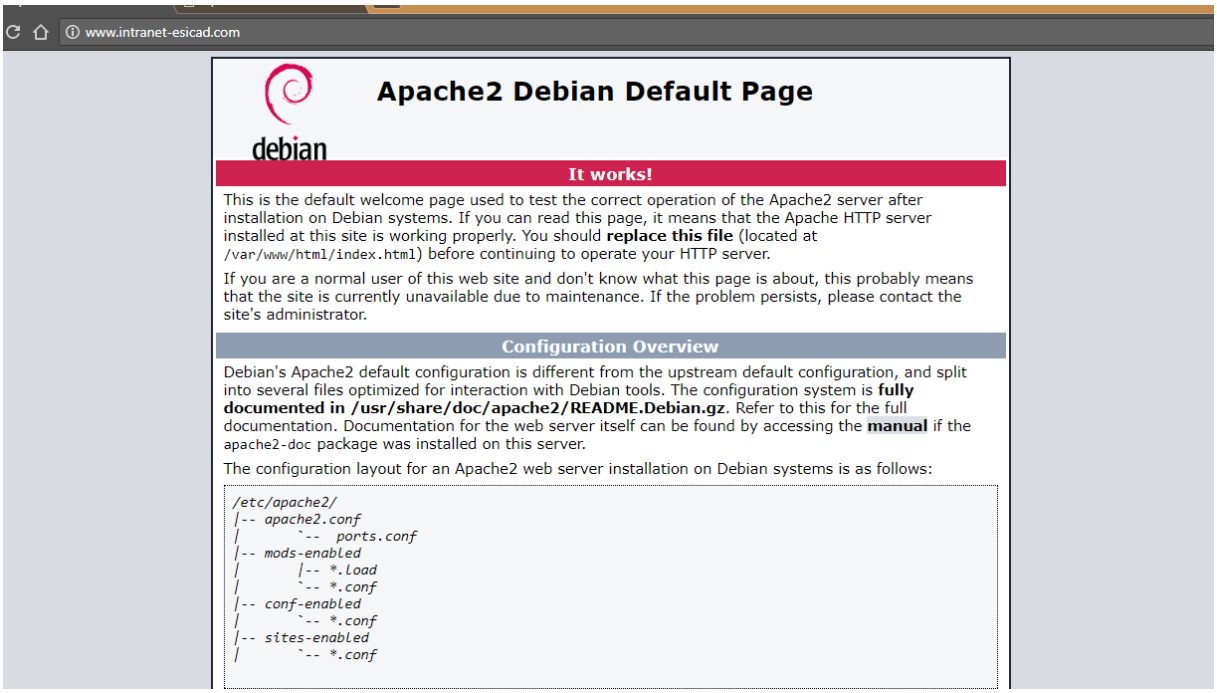
Ouvrez ce fichier de configuration, et modifiez :

- ServerName www.intranet-esicad.com
- DocumentRoot /var/www/html/intranet

```
<VirtualHost *:80>
        # The ServerName directive sets the request scheme, hostname and port t$
        # the server uses to identify itself. This is used when creating
        # redirection URLs. In the context of virtual hosts, the ServerName
        # specifies what hostname must appear in the request's Host: header to
        # match this virtual host. For the default virtual host (this file) this
        # value is not decisive as it is used as a last resort host regardless.
        # However, you must set it for any further virtual host explicitly.
        ServerName www.intranet-esicad.com
        DocumentRoot /var/www/html/intranet

        # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
        # error, crit, alert, emerg.
        # It is also possible to configure the loglevel for particular
        # modules, e.g.
        #LogLevel info ssl:warn
```

Pour terminer, il vous suffit de créer le lien du fichier dans le dossier /etc/apache2/sites-enabled, une commande a été faite spécialement : « a2ensite intranet-esicad.conf »

Puis redemarrez le service Apache2, « /etc/init.d/apache2 restart »



Sécurisation des accès avec filtrage du trafic par iptables

Restriction demandée :

- Les élèves n'ont accès qu'au serveur web (HTTP : 80)
- Le service administratif aura aussi accès à l'intranet.
- L'accès en FTP est réservé au réseau administratif.

Accepter le FTP pour le réseau administratif

Nous allons tout d'abord mettre en place la régle permettant au réseau administratif d'accéder au FTP en écrivant la commande :

1. Iptables –A FORWARD –s 192.168.3.0/24 –p tcp --dport 21 –j ACCEPT

La commande permet d'autoriser le réseau 192.168.3.0 d'accéder au port 21 qui est le port du FTP.

Vous pouvez vérifier la régle rentrée en écrivant la commande :

- Iptables –L

```
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
ACCEPT     tcp  --  192.168.3.0/24       anywhere             tcp dpt:ftp

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

Accepter l'accès à l'intranet pour les élèves et le réseau administratif

Pour permettre aux élèves et au service administratif d'accéder au serveur web, il faut accepter le port 80 sur leurs réseaux, pour cela il faut écrire la commande :

1. Iptables –A FORWARD –s 192.168.3.0/24 –p tcp –dport 80 –j ACCEPT

2. Iptables –A FORWARD –s 192.168.2.0/24 –p tcp –dport 80 –j ACCEPT

La première commande permet au réseau 192.168.3.0 d'accéder au port 80 qui est le port http

La deuxième commande permet au réseau 192.168.2.0 d'accéder au port 80 qui est le port http

```
Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
ACCEPT     tcp  --  192.168.3.0/24       anywhere            tcp dpt:ftp
ACCEPT     tcp  --  192.168.3.0/24       anywhere            tcp dpt:http
ACCEPT     tcp  --  192.168.2.0/24       anywhere            tcp dpt:http

Chain OUTPUT (policy ACCEPT)
```

Refuser aux élèves la connexion FTP

Les stagiaires ne peuvent accéder qu'à l'intranet, donc il faut refuser la connexion FTP. Pour cela nous allons écrire la commande :

- Iptables –A FORWARD –s 192.168.2.0/24 –p tcp –dport 21 –j DROP

Automatiser le montage des règles IPTABLES

Pour aller plus loin dans les règles IPTABLES, nous pouvons automatiser ses règles pour éviter de devoir les écrire à chaque fois que le routeur est redémarré. Pour cela nous allons écrire la commande :

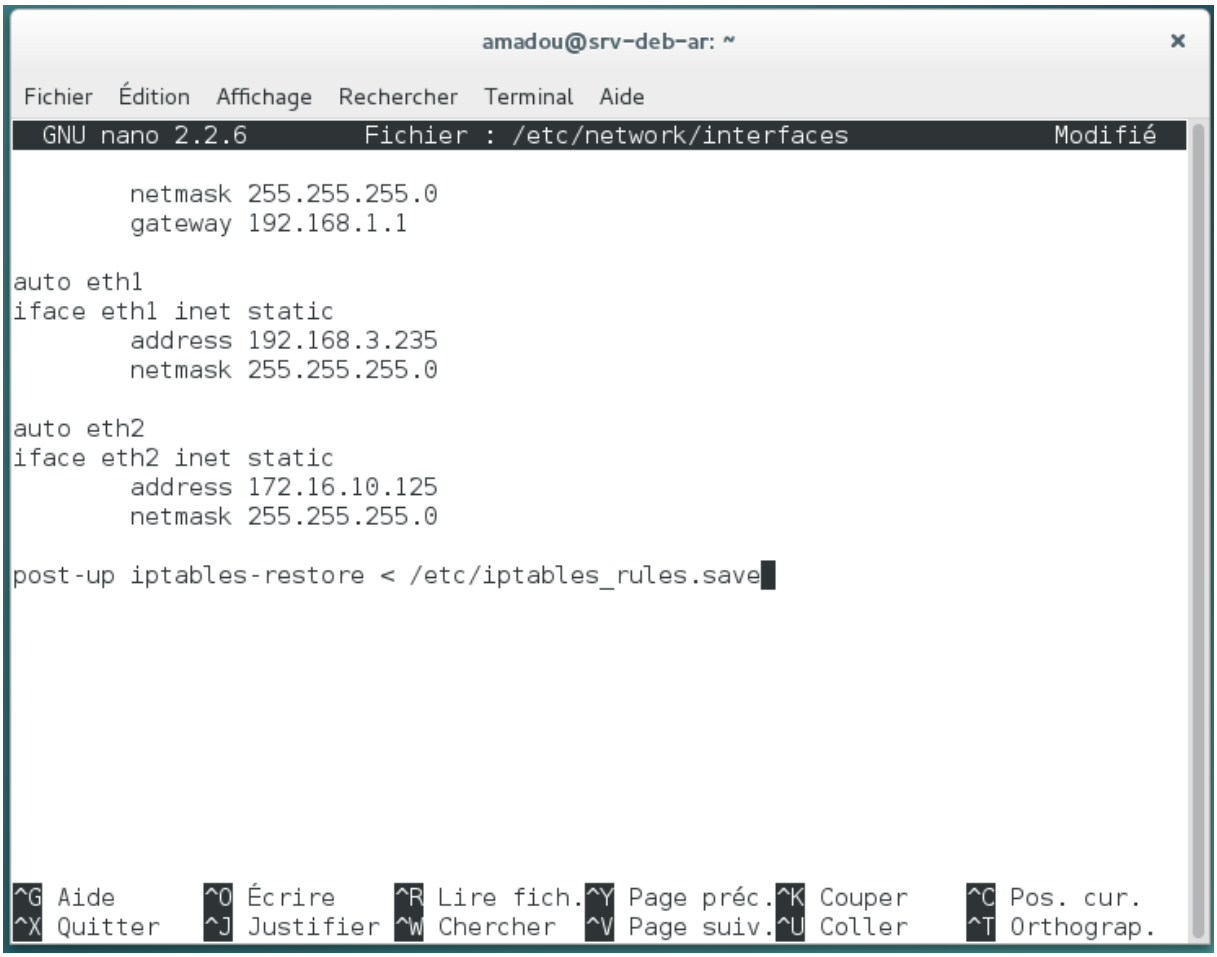
- iptables-save > /etc/iptables_rules.save

Iptables_rules.save = correspond à un nom que nous avons donné , ce nom n'est pas obligatoire, nous pouvons aussi appeler ce fichier : rulesiptables.save

Après avoir entré cette commande, nous allons ajouter dans le fichier /etc/network/interfaces la ligne de commande :

- post-up iptables-restore < /etc/iptables_rules.save

Cette commande permet à chaque fois de charger les règles du fichier « iptables_rules.save »


amadou@srv-deb-ar: ~
Fichier Édition Affichage Rechercher Terminal Aide
GNU nano 2.2.6 Fichier : /etc/network/interfaces Modifié
netmask 255.255.255.0
gateway 192.168.1.1
auto eth1
iface eth1 inet static
address 192.168.3.235
netmask 255.255.255.0
auto eth2
iface eth2 inet static
address 172.16.10.125
netmask 255.255.255.0
post-up iptables-restore < /etc/iptables_rules.save
^G Aide ^O Écrire ^R Lire fich. ^Y Page préc. ^K Couper ^C Pos. cur.
^X Quitter ^J Justifier ^W Chercher ^V Page suiv. ^U Coller ^T Orthograp.