****

UNIVERSITE CHEIKH ANTA DIOP DE DAKAR

FACULTE DES SCIENCES ET TECHNIQUE

DEPARTEMENT MATHS-INFO

[Date]

**Sécurité des systèmes**

**Rapport GLOBAL de SECURITE DES SYSTEMES EN M2SIR**

ANNEE

2019-2020

***Effectué par***

**MAMADOU NDIAYE**

***Sous la supervision du***

**Prof. KONATE**

**Plan**

**I . Ch1 :** **Filtrage avec les ACLs**

1. **Introduction des ACLs**
2. **FONCTIONNEMENT DES ACLs**
   1. **ACL standards**
   2. **ACL étendues**
   3. **ACL nommées**
3. **Étude Théorique**
   1. **Commandes de configuration de base d’un routeur**
   2. **Configuration du routeur NORD**
   3. **Configuration du routeur du SUD**
   4. **Configuration des machines et serveurs**
   5. **Ping entre les différentes machines**
4. **Mise en Œuvre des ACLs**

**4.0 Installation de Packet Tracer**

**4.1 Application aux ACLs**

**4.2 Questions**

**4.3 Solutions**

**4.4 Résultats**

**5. Références**

**I.Chapitre1 : Filtrage avec Les Access-Lists : ACL**

1. **Introduction des ACLs**

Une ACL (access control list) est un ensemble de règles permettant de filtrer ou d’autoriser du trafic sur un réseau en fonction de certains critères (IP source, IP destination, port source, port destination, protocole, …).

On applique une ACL a une interface d’un routeur : in lorsqu’il entre dans le routeur et out lorsqu’il sort. Une ACL est analysé de façon séquentielle.

Il existe deux types d’ACLs : ACL Standard et ACL étendue

**2. FONCTIONNEMENT DES ACLs**

Il est possible de résumer le fonctionnement des ACL de la façon suivante :

On compare la règle avec la condition :

* Si la condition correspond à la première règle, l’action définie est appliquée, le reste de l’ACL n’est pas analysé.
* Sinon le paquet est comparé successivement par rapport aux ACL suivants
* Si la condition ne correspond pas à aucune règle, l'action deny est appliquée

Les règles sont définit sur les informations contenues dans les en-têtes IP, TCP ou UDP Des masques ont été défini pour pouvoir identifier une ou plusieurs adresses IP en une seule définition .Ce masque défini la portion de l'adresse IP qui doit être examinée 0.0.255.255 signifie que seuls les 2 premiers octets doivent être examinés deny 10.1.3.0 avec 0.0.0.255 : refus de toutes les IP commençant par 10.1.3

Remarque: Lorsqu’une ACL contient plusieurs règles il faut partir du particulier au général, et donc les plus génériques en fin de liste.

Il existe deux types d’ACL.

**2-1 : ACL Standard**

Ils ne peuvent pas filtrer selon la destination mais selon la source.

Les ACLs standard sont :

Rapide à exécuter

Facile de mettre en œuvre

Les ACLs standard sont à appliquer le plus proche possible de la destination.

Ils ont des numéros de 1-99 et de 1300-1999

Syntaxe générale

masque generique

un numéro identifie ACL (1-99)

access-list numéro **<action>** <IP source> < wildcard mask.>

Permit/deny

Permet de Définir une ACL

L’adresse IP du réseau source

Exemple

NORD (config)# access-list 10 permit 192.168.2.0 0.0.0.255

**2-2 :ACL Étendues**

Ils peuvent filtrer selon : Adresse IP source, Adresse IP destination, Protocole, Numéro de Port, temps, qualité de service…

Les ACLs étendues sont à appliquer le plus proche possible de la source.

Ils ont des numéros de 100-199 et de 2699-2699

Syntaxe générale

access-list number { deny | permit} protocol @IP-source source-wildcard @IP-destination dest-wildcard

* Le mot clés access-list : qui permet de définir une ACL numéroté.
* number : un numéro permettant d’identifier une ACL.
* L’action à effectuer: deny (supprimer le paquet), permit (autoriser le paquet).
* protocol :on definit le protocol sur lequel on applique l’ACL
* L’adresse IP du réseau source : identifier le réseau source.
* L’adresse IP du réseau destinataire : identifier le réseau destinataire.
* Le wildcard mask (source, destination) : le masque inversé pour savoir quel nombre de bits de l'adresse réseau doit correspondre.

**2-3 :ACL Nommées**

Une ACL numéroté peut être composé de nombreuses régles. La seule façon de la modifier est de faire :

* no access-list number
* Puis de la redéfinir

Avec les ACL nommées, il est possible de supprimer qu'une seule ligne au lieu de toute l'ACL

Sa définition se fait de la manière suivante :

* Router(config)# ip access-list extended bart
* Router(config-ext-nacl)# deny tcp host 10.1.1.2 eq www any
* Router(config-ext-nacl)# deny ip 10.1.1.0 0.0.0.255 any
* Router(config-ext-nacl)# permit ip any any

Pour supprimer une des lignes, il suffit de refaire un

ip access-list extended bart

Puis un no deny ip 10.1.1.0 0.0.0.255 any

**Remarque**

Ils est possible de définir des listes de contrôle d'accès sans les appliquer. Toutefois, les listes de contrôle d'accès n'ont aucun effet tant qu'elles ne sont pas appliquées à l'interface du routeur.

Appliquer(activer) une ACL sur une interface

ip access-group [ number | name [in | out ] ]

Visualiser les ACL

* show access-lists [ number | name ] : toutes les ACL quel que soit l'interface
* show ip access-lists [ number | name ] : les ACL uniquement liés au protocole IP

Désactiver une ACL sur une interfacew

R1(config)#interface fastethernet 0/0

R1(config-if)#**no access-group 1 in**

OU

R1(config-if)#**no access-group 1 out**

R1(config-if)#

**3. Étude Théorique**

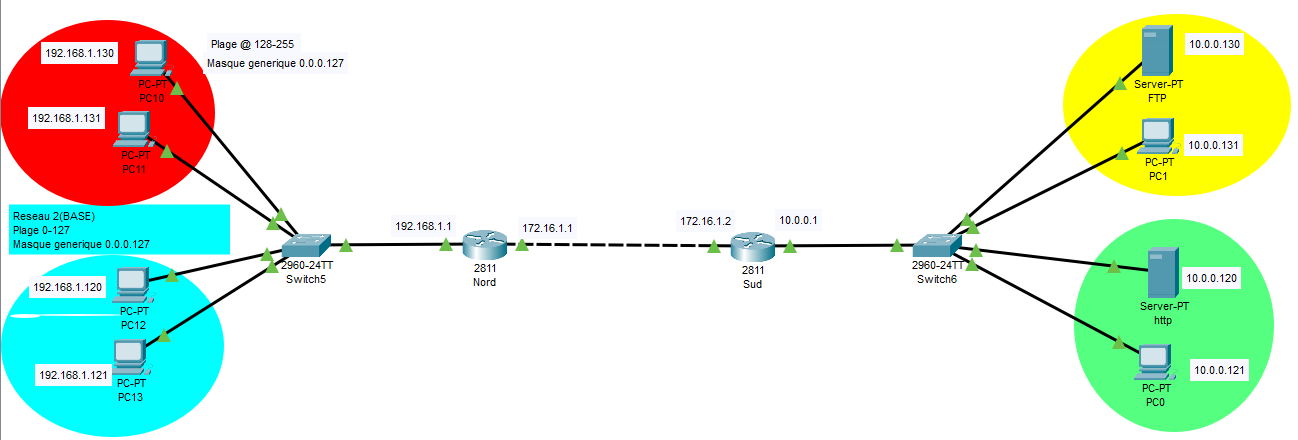
Voici le réseau que nous avons à configurer : il est constitué de deux routeurs nommé Nord et Sud relié respectivement au réseau d’adresses respectives 192.168.1.0 et 10.0.0.0.

L’adresse 192.168.1.0 est décomposé aux sous réseaux 192.168.1.0 et 192.168.1.128 qui portent les machines PC10, PC11, PC12, PC13.

L’adresse 10.0.0.0 est décomposé aux sous réseaux 10.0.0.0 et 10.0.0.128 qui portent les machines PC0, PC1 et les serveurs HTTP, FTP.

C’est dans ce réseau que nous allons appliquer les ACL pour contrôler les accès entre les différents terminaux du réseau.

Le réseau est illustré par le schéma ci-dessous :



* 1. **.Commandes de configuration de base d’un routeur**
* **Changer le nom du routeur:**

*Router#config terminal*

*Router(config)#hostname RouterNORD(nom du routeur)*

* **Mettre un mot de passe pour la console:**

*RouterA#configure terminal*

*RouterA(config)#line console 0*

*RouterA(config-line)#password \*\*\*\*\**

*RouterA(config-line)#login*

*RouterA(config-line)#exit*

*RouterA(config)#*

* **Mettre un mot de passe pour les lignes virtuelles (pour la configuration à travers Telnet) :**

*RouterA(config)#line vty 0 4*

*RouterA(config-line)#password \*\*\*\*\**

*RouterA(config-line)#login*

*RouterA(config-line)#exit*

*RouterA(config)#enable password \*\*\*\*\**

* **Chiffrer les mots de passe:**

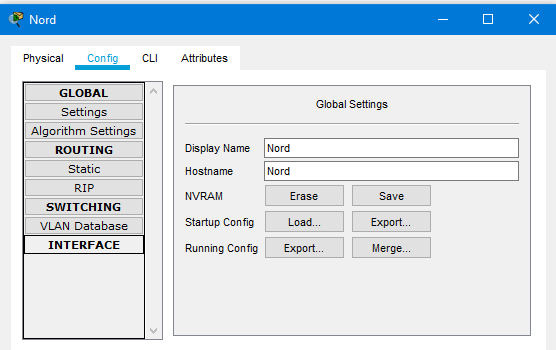
*RouterA(config)#service password-*encryption

* **Enregistrer la configuration**

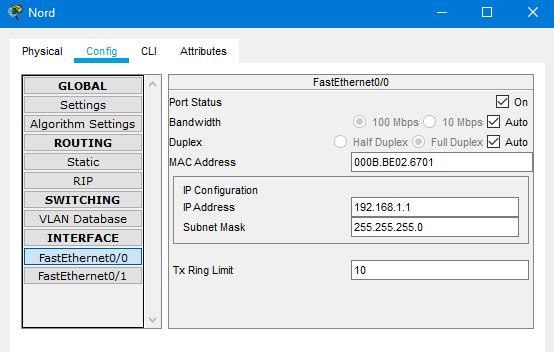
*Router# copy running-config startup-config* OU

*Router# write memory*

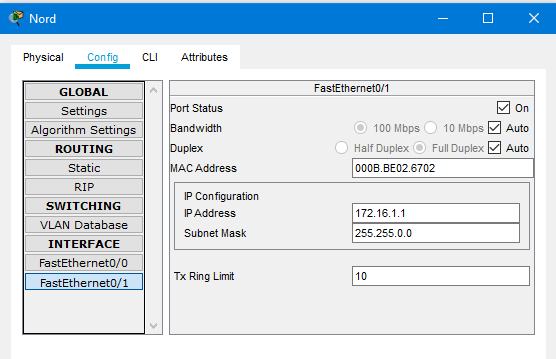
* 1. **Configuration du Routeur Nord**
* **Changement de nom du routeur**

****

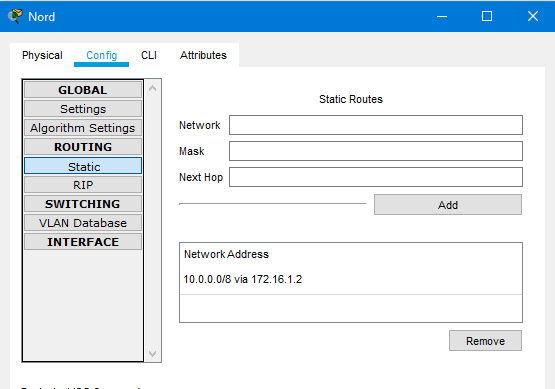
* **Configuration du FastEthernet0/0 du routeur**

****

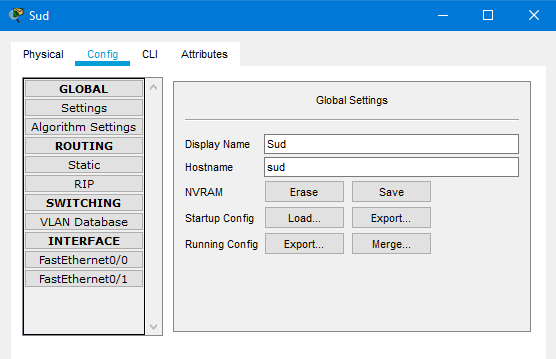
* **Configuration de fa0/1 du routeur**

****

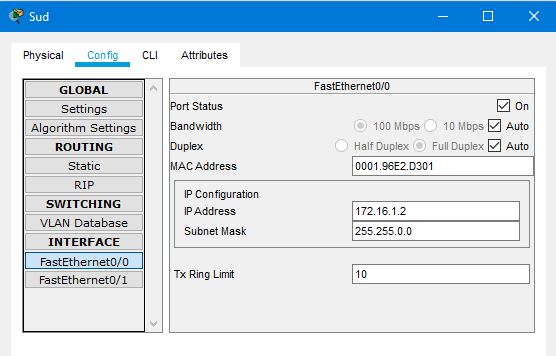
* **Configuration de Static Routes**

****

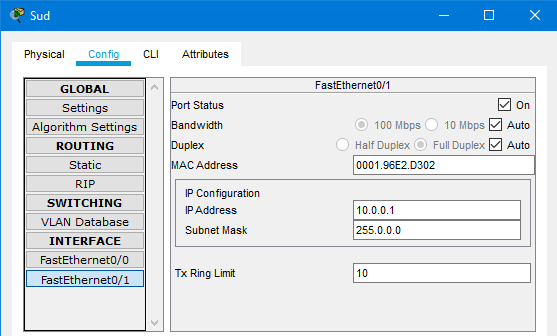
* 1. **Configuration du Routeur Sud**
* **Changement de nom du routeur**

****

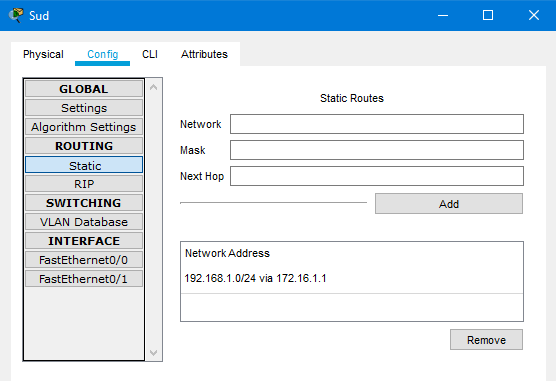
* **Configuration du FastEthernet0/0 du routeur Sud**

****

* **Configuration de fa0/1 du routeur Sud**

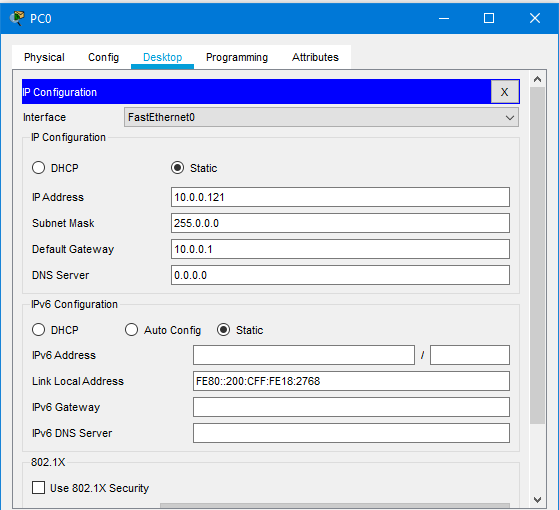


* **Configuration de Static Routes**

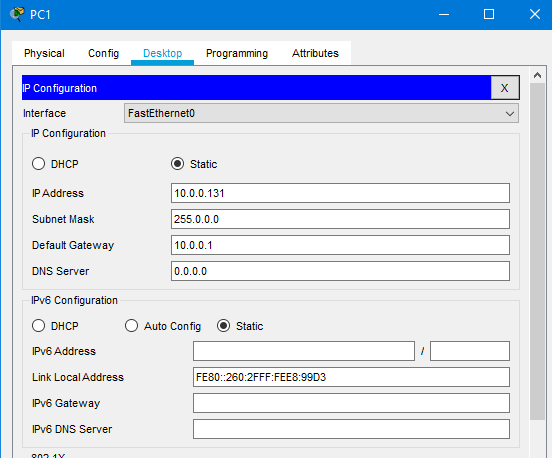
****

**3.4)Configuration des machines et des serveurs**

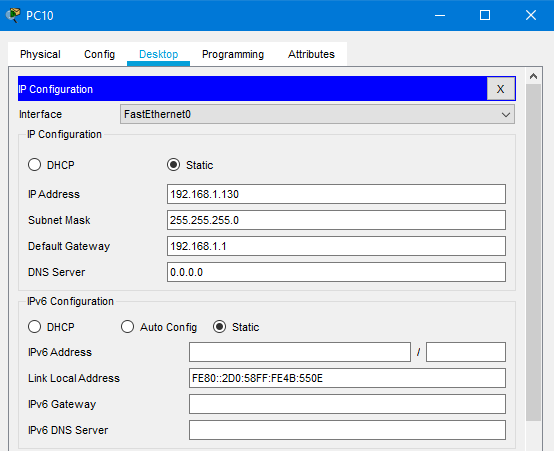
* **PC0**

****

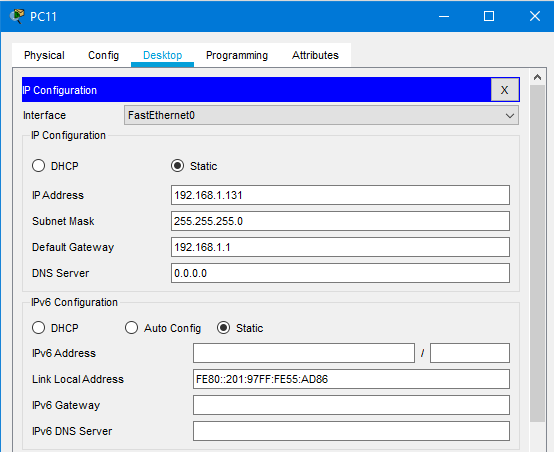
* **PC1**

****

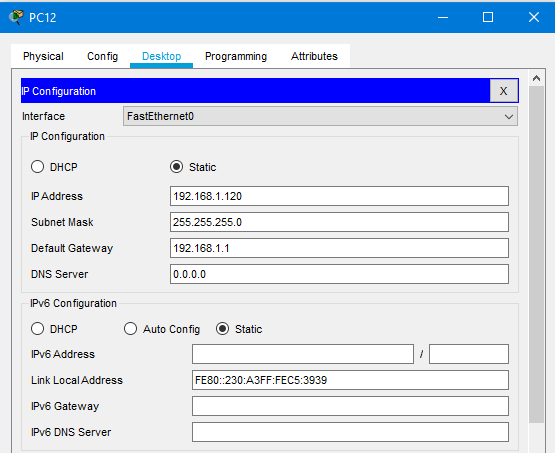
* **PC10**

****

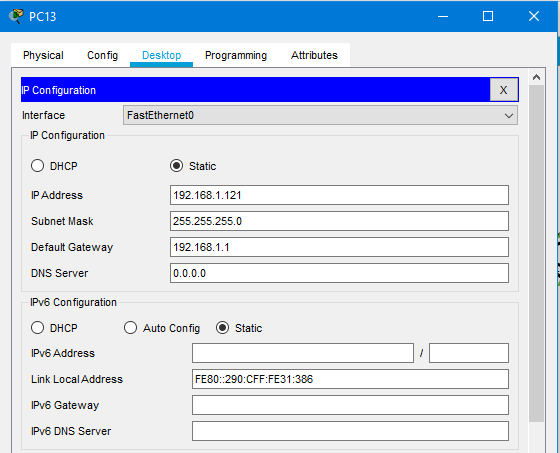
* **PC11**

****

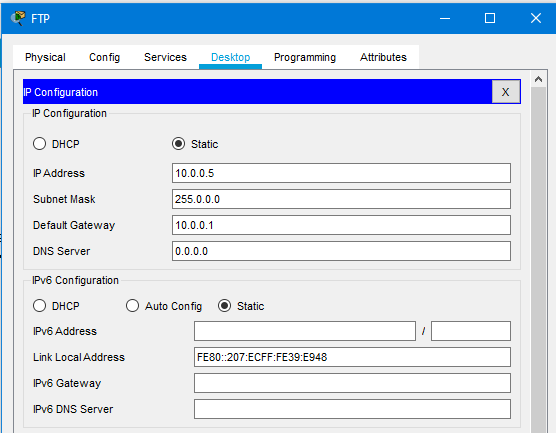
* **PC 12**

****

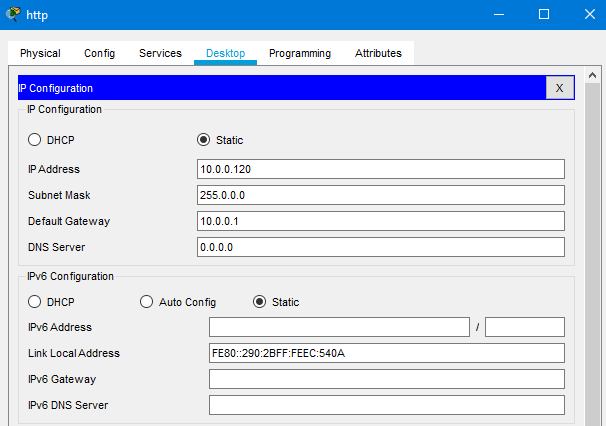
* **PC13**

****

* **FTP**

****

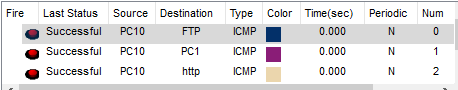
* **http**

****

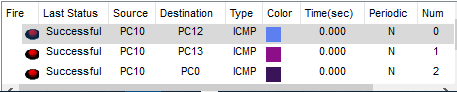
**3.5)Ping entre les différentes machines**

**Ping PC 10**

* + **PC10 vers PC1, FTP et http :**

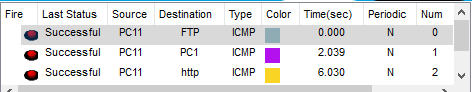
****

* + **PC10 vers PC12, PC13 et PC0 :**

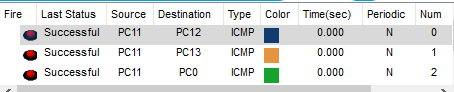
****

**Ping PC11 :**

* + **PC11 vers PC0, FTP et http :**

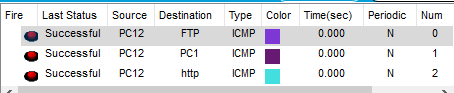
****

* + **PC11 vers PC12, PC13, PC1**

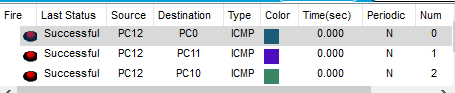
****

**Ping PC12 :**

* + **PC12 vers PC1, FTP et http**

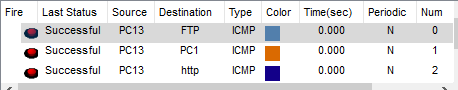
****

* + **PC12 vers PC10, PC11 et PC0**

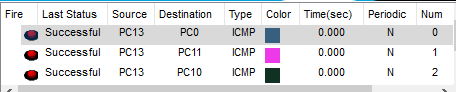
****

**Ping PC13 :**

* + **PC13 vers PC1, FTP et http**

****

* + **PC13 vers PC0, PC11 et PC10**

****

1. **Application des ACL**

Les ACL permettent de gérer le contrôle d’accès dans un réseau, à une machine, à un serveur…

Dans ce réseau nous avons vu que toutes les machines se voient entre elles et peuvent accéder aux serveurs FTP et HTTP. Nous allons gérer les contrôles d’accès suivant les questions ci-dessous :

**4.2 ) Questions**

1. Toutes les machines de la partie haute (192.168.1.128) de Nord peuvent accéder à toutes les machines la partie haute (10.0.0.128) de Sud.

2. Toutes les machines de la partie haute (192.168.1.128) de Nord peuvent accéder uniquement au serveur ftp (10.0.0.130) de la partie haute de Sud.

3. Toutes les machines de la partie haute (192.168.1.128) de Nord ne peuvent pas accéder à la partie base (10.0.0.0) de Sud.

4. Toutes les machines de la partie haute (192.168.1.128) de Nord ont accès au Serveur http(10.0.0.130) de Sud

5. Toutes les machines de la partie basse (192.168.1.0) de Nord peuvent accéder à la partie base (10.0.0.0) de Sud.

6. Toutes les machines de la partie base (192.168.1.0) de Nord peuvent accéder uniquement au serveur http (10.0.0.120) de Sud.

7. Toutes les machines de la partie base (192.168.1.128) de Nord ne peuvent pas accéder à la partie haute (10.0.0.128) de Sud.

8. Toutes les machine de la partie base ont accès au Serveur FTP(10.0.0.130) de Sud.

**4.3) Solution**

Pour appliquer ces access lists contrôles nous avons fait les commandes suites à la face FastEthernet0/0 du router Nord:

Création du groupe :

Nord(config)# interface FastEthernet0/0

Nord(config)# ip access-group 101 in

Les différentes commandes du contrôle d’accès :

Nord (config)# Access-list 101 deny tcp 176.16.10.128 0.0.0.127 host 192.168.1.131 eq ftp

Nord (config)# Access-list 101 permit tcp 176.16.10.0 0.0.0.127 host 192.168.1.132 eq www

STLouis(config)# Access-list 101 deny ip 176.16.10.128 0.0.0.127 192.168.1.0 0.0.0.127

STLouis(config)# Access-list 101 deny ip 176.16.10.0 0.0.0.127 192.168.1.128 0.0.0.127

STLouis(config)# Access-list 101 permit ip any any

**Chapitre II : RAPPORT DE CONFIGURATION D’UN FIREWALL AVEC NETFILTER-IPTABLES**

1. **Définition de Netfilter**

**Netfilter** est un [framework](http://dictionnaire.sensagent.leparisien.fr/Framework/fr-fr/) implémentant un [pare-feu](http://dictionnaire.sensagent.leparisien.fr/Pare-feu%20(informatique)/fr-fr/) au sein du [noyau Linux](http://dictionnaire.sensagent.leparisien.fr/Noyau%20Linux/fr-fr/) à partir de la version 2.4 de ce dernier. Il prévoit des accroches dans le [noyau](http://dictionnaire.sensagent.leparisien.fr/Noyau%20de%20syst%C3%A8me%20d'exploitation/fr-fr/)

pour l'interception et la manipulation des [paquets réseau](http://dictionnaire.sensagent.leparisien.fr/Paquet%20(r%C3%A9seau)/fr-fr/) lors des appels des routines de réception ou d'émission des paquets des [interfaces réseau](http://dictionnaire.sensagent.leparisien.fr/Carte%20r%C3%A9seau/fr-fr/).

La version 1.4.2 a reçu un Certificat de Sécurité de Premier Niveau (CSPN) par l'[Agence nationale de la sécurité des systèmes d'information](http://dictionnaire.sensagent.leparisien.fr/Agence%20nationale%20de%20la%20s%C3%A9curit%C3%A9%20des%20syst%C3%A8mes%20d'information/fr-fr/).

**2. Définition d’Iptable :**

**Iptable** est un [logiciel libre](https://fr.wikipedia.org/wiki/Logiciel_libre) de l'[espace utilisateur](https://fr.wikipedia.org/wiki/Espace_utilisateur) [Linux](https://fr.wikipedia.org/wiki/Linux) grâce auquel l'[administrateur système](https://fr.wikipedia.org/wiki/Administrateur_syst%C3%A8me) peut configurer les chaînes et règles dans le [pare-feu](https://fr.wikipedia.org/wiki/Pare-feu_(informatique)) en [espace noyau](https://fr.wikipedia.org/wiki/Espace_noyau) (et qui est composé par des modules [Netfilter](https://fr.wikipedia.org/wiki/Netfilter)).

Différents programmes sont utilisés selon le protocole employé : Iptables est utilisé pour le protocole [IPv4](https://fr.wikipedia.org/wiki/IPv4), *Ip6tables* pour [IPv6](https://fr.wikipedia.org/wiki/IPv6), *Arptables* pour ARP ([Address Resolution Protocol](https://fr.wikipedia.org/wiki/Address_Resolution_Protocol)) ou encore *Ebtables*, spécifique aux trames [Ethernet](https://fr.wikipedia.org/wiki/Ethernet).

Ce type de modifications doit être réservé à un administrateur du système. Par conséquent, son utilisation nécessite l'utilisation du compte [root](https://fr.wikipedia.org/wiki/Utilisateur_root). L'utilisation du programme est refusée aux autres utilisateurs.

Sur la plupart des [distributions Linux](https://fr.wikipedia.org/wiki/Distribution_Linux), Iptables est lancé par la commande /usr/bin/iptables et documenté par sa page de manuel iptableset ip6tables, laquelle peut être visualisée via la commande «man iptables ».

*Iptables* est également fréquemment utilisé pour faire référence aux composants de bas niveau (niveau kernel). *X\_tables* est le nom du module noyau, plus générique, qui contient le code partagé pour les quatre protocoles. C'est aussi le module qui fournit l'[API](https://fr.wikipedia.org/wiki/Interface_de_programmation) des extensions. Par conséquent, *Xtables* désigne usuellement le pare-feu complet (IPv4, IPv6, arp, eb).

**3. Configuration du Firewall**

Le pare-feu Iptables Linux est utilisé pour surveiller le trafic entrant et sortant vers un serveur et le filtrer en fonction des règles définies par l’utilisateur, afin d’empêcher toute personne d’accéder au système. En utilisant Iptables, on peut définir des règles qui n’autoriseront que le trafic sélectionné sur notre serveur.

Toutes les données sont envoyées sous forme de paquets sur Internet. Le noyau Linux fournit une interface pour filtrer les paquets de trafic entrants et sortants à l’aide de tableaux de filtres de paquets. Iptables est une application de ligne de commande et un pare-feu Linux qu’on peut utiliser pour configurer, maintenir et inspecter ces tableaux. Plusieurs tableaux peuvent être définis. Chaque tableau peut contenir plusieurs chaînes. Une chaîne n’est qu’un ensemble de règles. Chaque règle définit ce qu’il faut faire avec le paquet, s’il correspond à ce paquet. Lorsque le paquet correspond, il lui est attribué une cible. Une cible peut être une autre chaîne, pour correspondre à l’une ou l’autre des valeurs spéciales suivantes :

* **ACCEPT** : Cela signifie que le paquet sera autorisé à passer.
* **DROP** : Cela signifie que le paquet ne sera pas autorisé à passer.
* **RETURN** : Cela signifie ignorer la chaîne actuelle et revenir à la règle suivante de la chaîne dans laquelle elle a été appelée.

Le tableau des filtres comporte trois chaines :

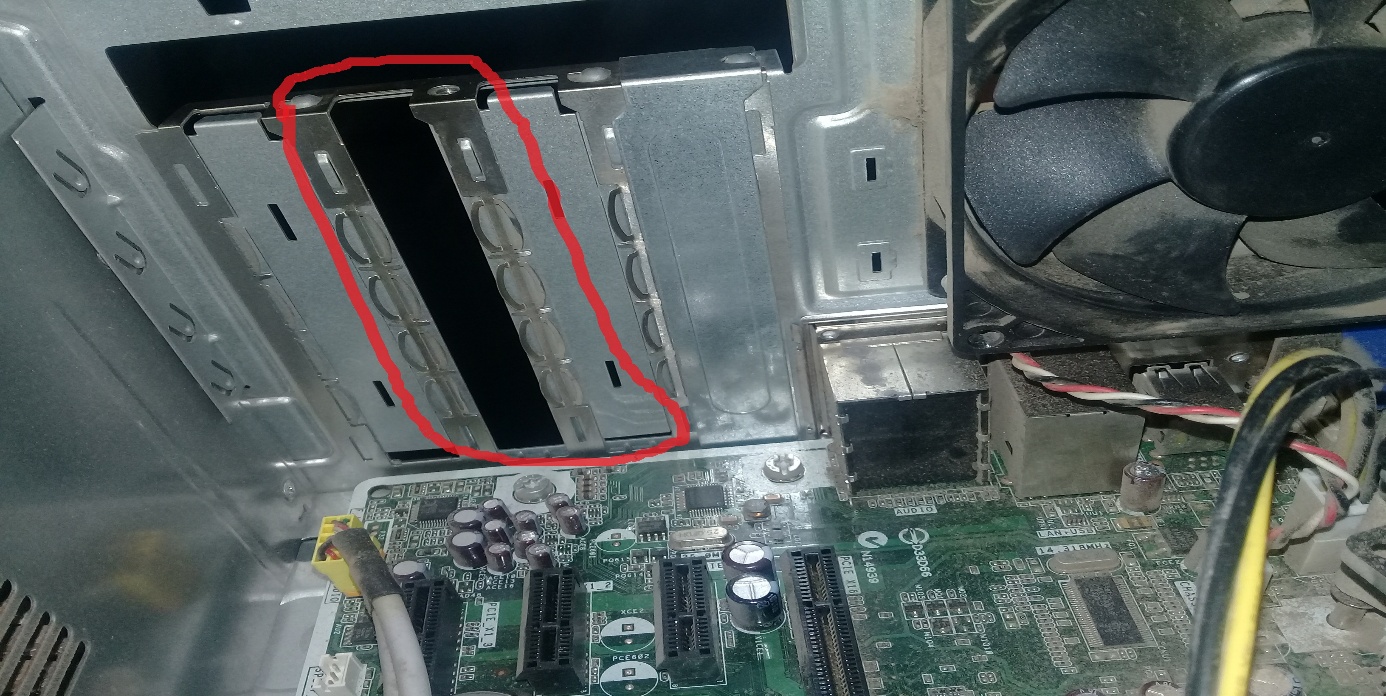
* **INPUT** – Cette chaîne est utilisée pour contrôler les paquets entrants sur le serveur. On peut bloquer/autoriser les connexions en fonction du port, du protocole ou de l’adresse IP source.
* **FORWARD** – Cette chaîne est utilisée pour filtrer les paquets qui entrent sur le serveur mais doivent être transférés ailleurs.
* **OUTPUT** – Cette chaîne est utilisée pour filtrer les paquets qui sortent de votre serveur.

**3.1Ajout d’une carte réseau**

La machine dont nous avons utilisé comme Firewall est un Desktop voici la **carte mère :**

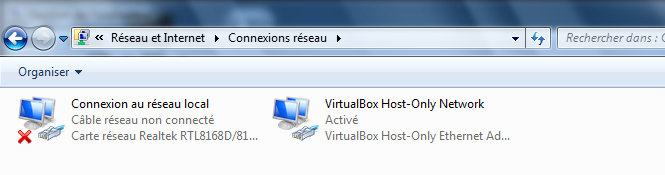


Figure 1.1 :Carte mère Desktop

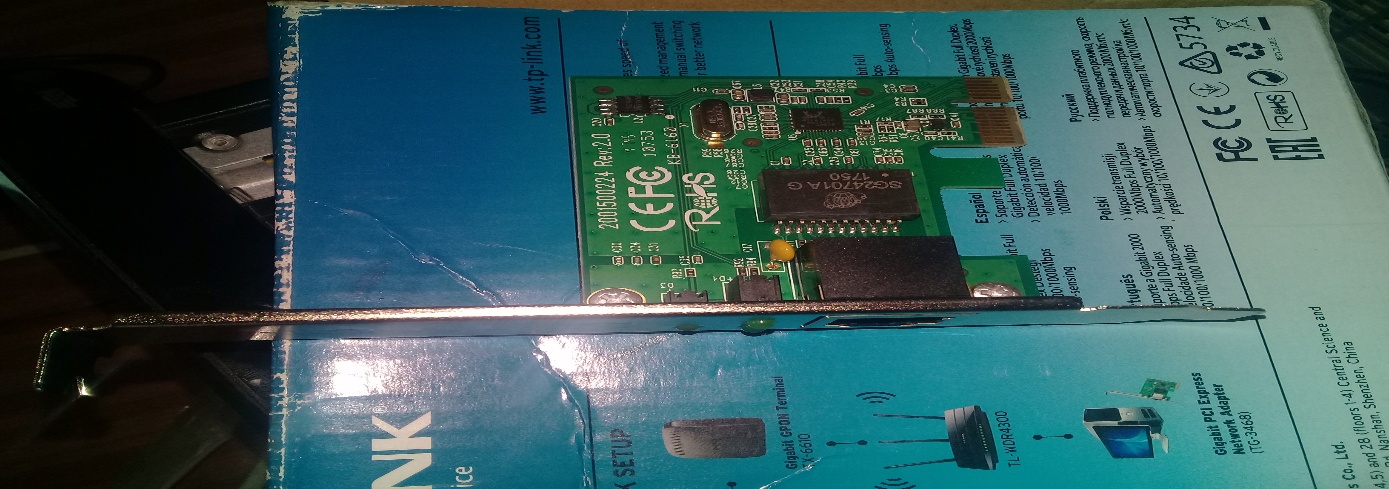
****

Point de montage de la carte mère

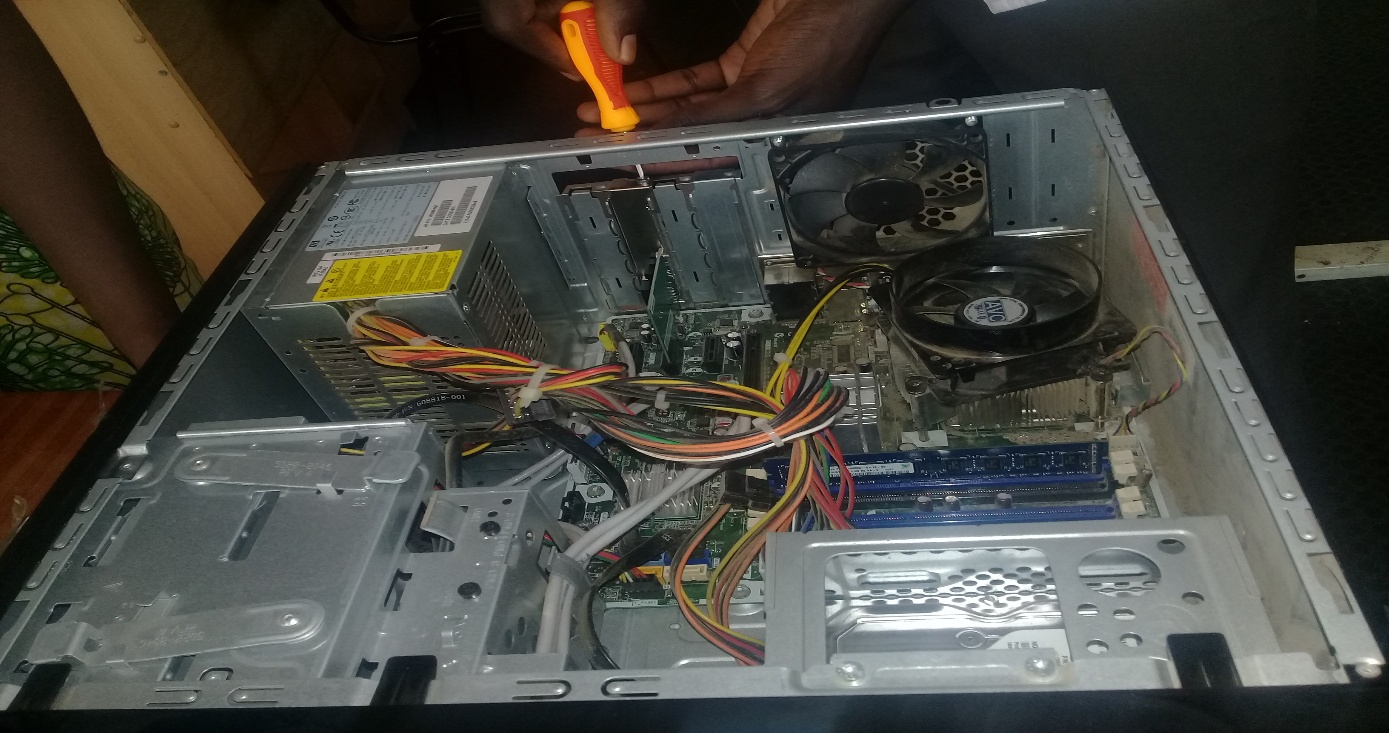
**Avant montage de carte réseau :** nous avons dansPanneau de configuration »Réseau et Internet »Connexions réseau »de l’OS de la machinel’architecture suivante :



**Voici la carte réseau :**

****

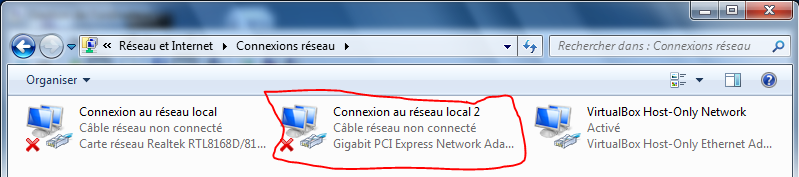
**Montage de la carte**

****

**Carte prête :**



**Après montage de la carte réseau :** nous avons dans Panneau de configuration »Réseau et Internet »Connexions réseau » de l’OS de la machine l’architecture suivante :



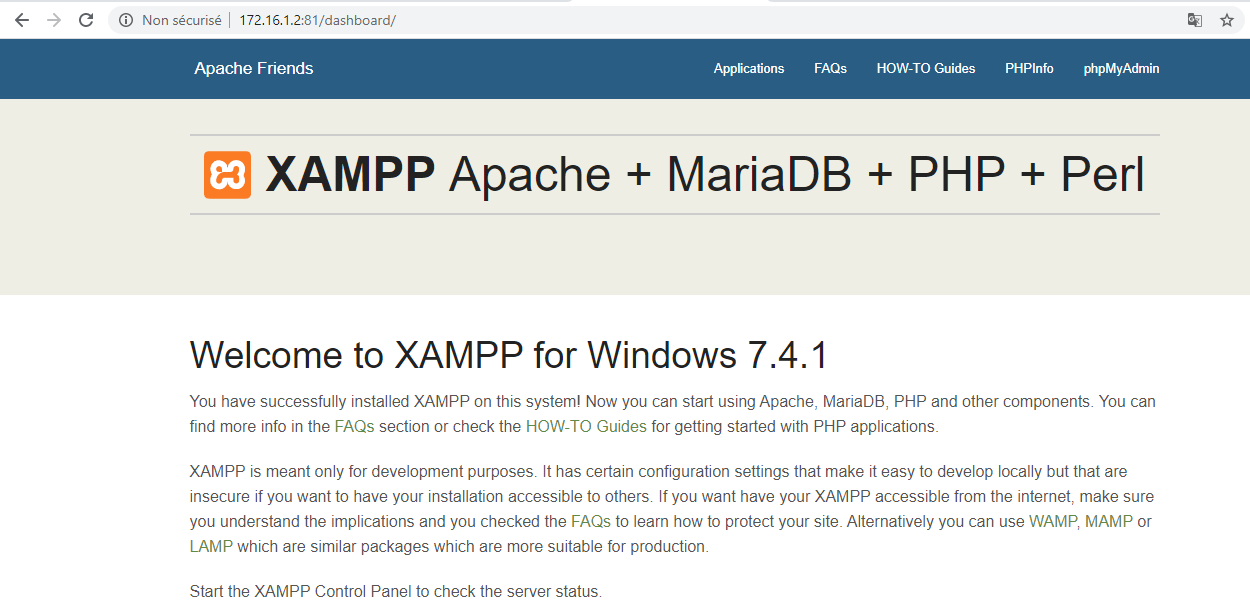
Ainsi nous avons fait toutes les configurations nécessaires pour le bon fonctionnement de la carte réseau.

Donc notre Firewall dispose maintenant de deux cartes réseaux.

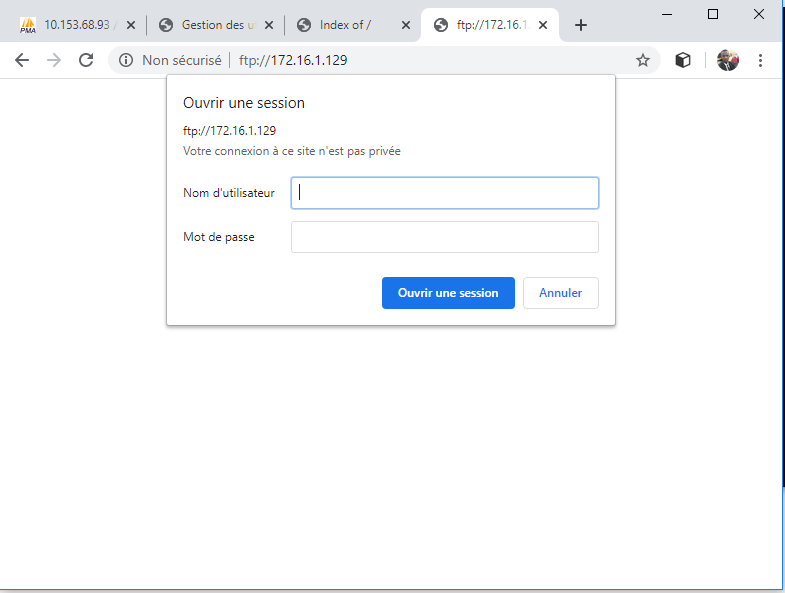
Avant utilisation d’Iptables

* Accès sur le même réseau

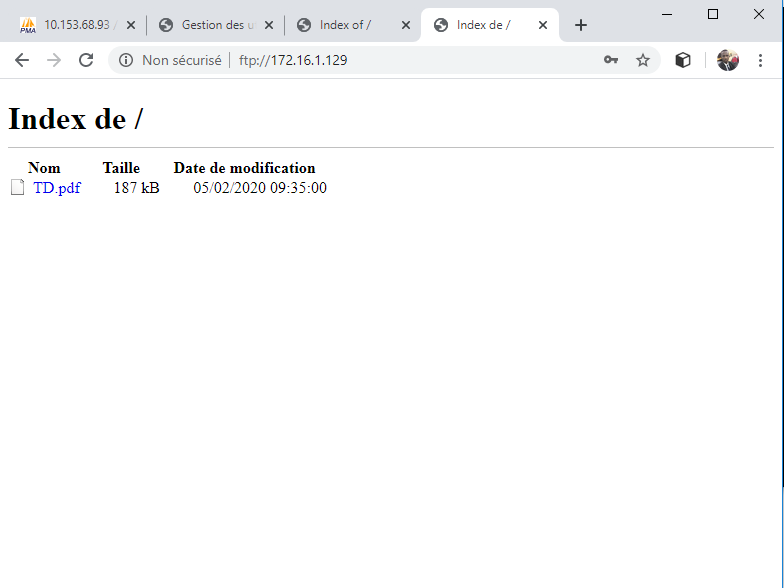
L’hôte 172.16.1.1 vers http 172.16.1.2



* L’hôte 172.16.1.1 vers FTP 172.16.1.129

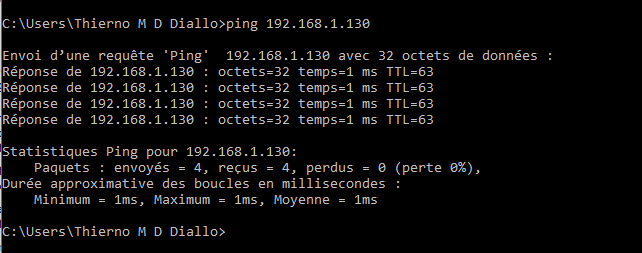


Pour voir les données qui se trouve sur le serveur ftp on se logue avec le nom de l’utilisateur et son mot de passe.

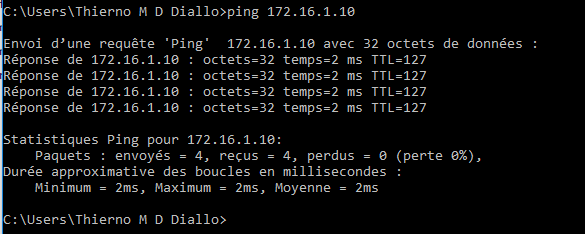


Ainsi on peut récupérer le fichier, on fait un double clic sur TD.pdf.

* Ping entre l’hôte 172.16.1.10 et l’interface 192.168.1.130



* Ping entre l’hôte 192.168.1.129 et l’interface 172.16.1.10



* Accès entre les deux réseaux différents : pour se faire on doit :

Editer le fichier /etc/sysctl.conf avec la commande suivante

root@srv-01:# vim /etc/sysctl.conf

puis on décommente la ligne net.ipv4.ip\_forward=1

et pour que la machine prend en compte cette modification à chaque ouverture on tape la commande suivante

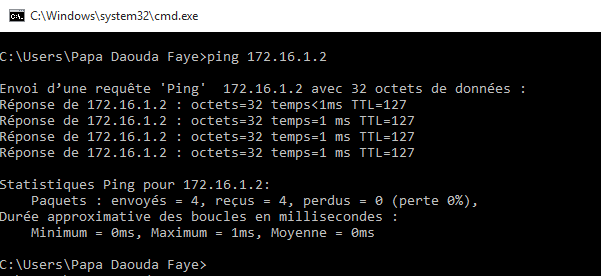
root@srv-01:# sysctl –p

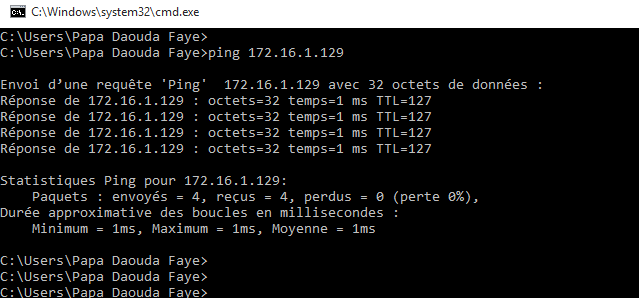
et après on tape les commandes suivantes pour le routage entre les réseaux

root@srv-01:# route add –net 172.16.1.0 netmask 255.255.255.0 gw 172.16.12.254

root@srv-01:# route add –net 192.168.1.0 netmask 255.255.255.0 gw 192.168.1.254

Ping entre l’hôte 192.168.1.129 et l’hôte 172.16.1.2/ 176.16.10.129





<http://dictionnaire.sensagent.leparisien.fr/Netfilter/fr-fr/>

<https://fr.wikipedia.org/wiki/Iptables>

<https://www.hostinger.fr/tutoriels/iptables/>