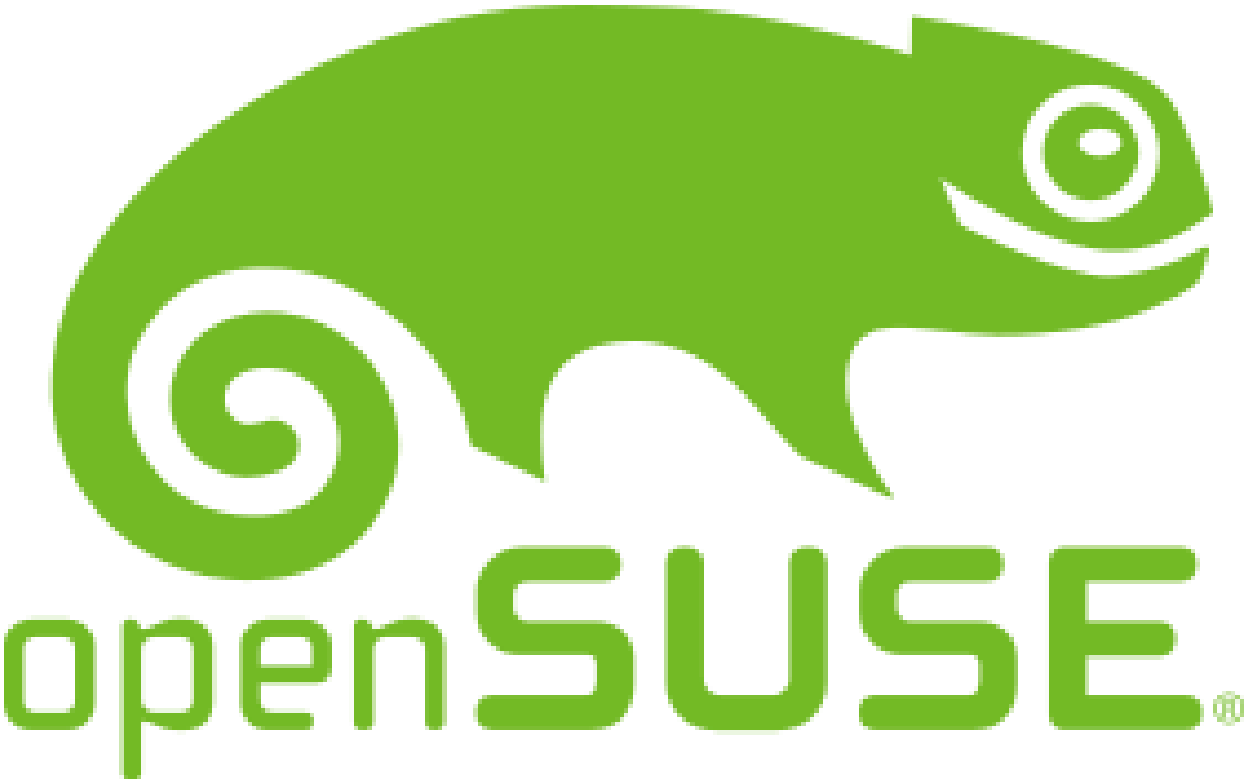
RAPPORT DE LABORATOIRE

Utilisation et Gestion Sécurisée de ftp sur OPENSUSE 42.3



Jebira Sélim

04/05/2020

SYSG5– Gestion

INTRODUCTION

Quand on vient à travailler à plusieurs ou à vouloir **partager des informations** avec les autres on se retrouve toujours confronté à la question de comment faire ?

Soit on envoie un fichier à chaque fois qu'on nous le demande. (Bonne chance !)

Soit on le rend accessible à tous. (Sécurité?)

Dans le premier cas, on se retrouve à envoyer à plein de personnes la même information.

Dans le deuxième cas, on se pose la question de "**qui a accès à ces données?"**.

Dans ce rapport, j'apporterai une réponse automatisée au deuxième cas .

j'aborderai des thèmes comme la **sécurisation de l'information** au travers des **communications** mais aussi au travers d'une politique de **gestion des utilisateurs et des groupes**.

j'y expliquerai

comment **mettre en place un réseau privé virtuel**.

Comment **créer un utilisateur , un groupe** et attribuer les bons droits de manière à l'**isoler du système hébergeant le serveur dans le cadre d'utilisation du protocole FTP**.

Different Principes liés au projet:

Le principe de clé publique et privée et pourquoi cela est important d'en tenir compte dans notre cas.

Le fonctionnement des communications en réseau au travers de l'**adressage privé/publique et des protocole TCP/UDP**.

Le concept de **SNIFFING** et **d'ARP Request**.

Plusieurs **Vulnérabilités** comme:

Une attaque par ManInTheMiddle (MITM) Nommé aussi **ARP POISONING**, le **DoS (Denial of Service)** et comment **s'en prémunir**.

Matériel utilisé

Je travail avec Lenovo IdeaPad 330S , processeur I5 8250 U sous Windows 10 disposant d'un accès à internet.

Prérequis

Les logiciels que j'utiliserai pour mettre à bien ce projet sont multiples, tous gratuits et accessible facilement.

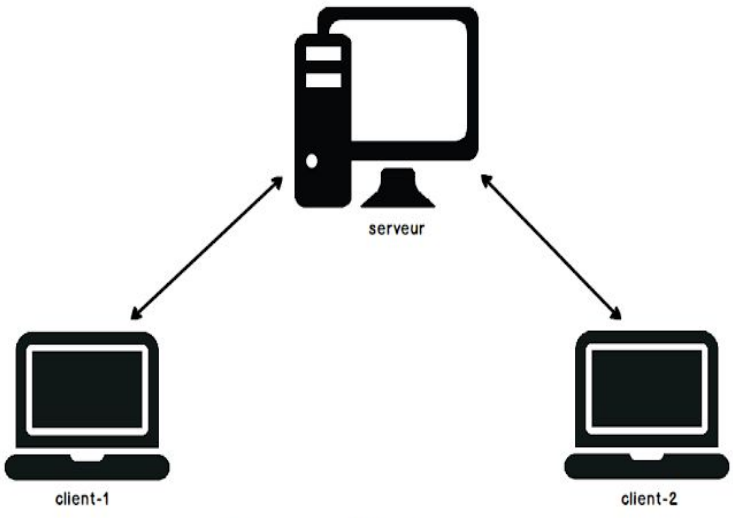
Dans un premier temps il me fallait un logiciel de virtualisation pour travailler sur Opensuse 42.3 .

J'ai choisis le logiciel libre VirtualBox fournit par Oracle.

et ensuite j'ai récupéré l'image ISO servant à installer l'OS via les repos OpenSuse.

Après la création de la première machine virtuelle sa mise à jour et l'ajout des additions Invitées je l'ai cloné 2 fois.

Ce faisant je me retrouve avec 3 machines virtuelles sous opensuse 42.3.



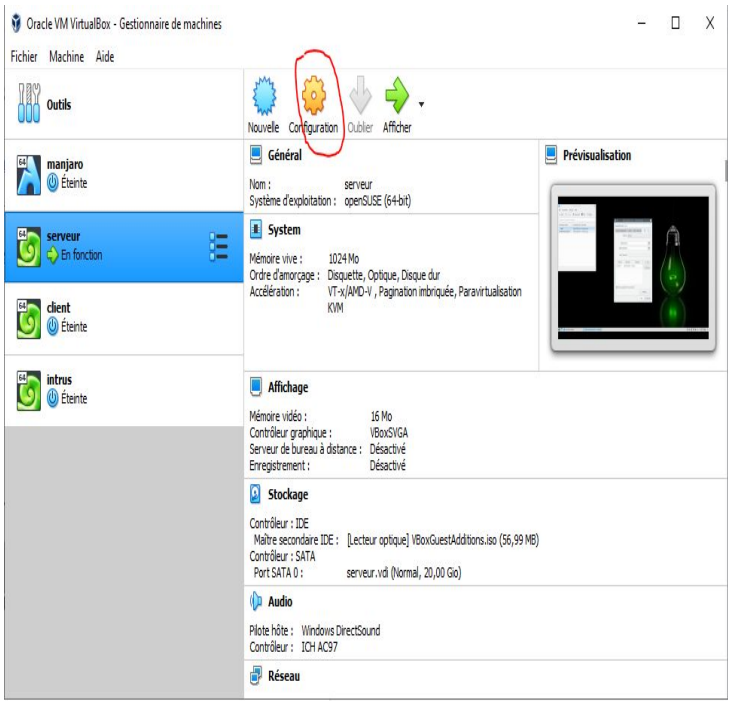
Réseau client-serveur

pour donner accès aux machines à internet (WAN) et leur permettre de communiquer entre elles (LAN).

il faut utiliser le mode **Réseau Nat**

j'expliquerai le WAN et LAN plus loin

la mise en place du réseau local se fait via [l'interface de configuration](#).



Réseau > mode d'accès réseau :

Réseau Nat

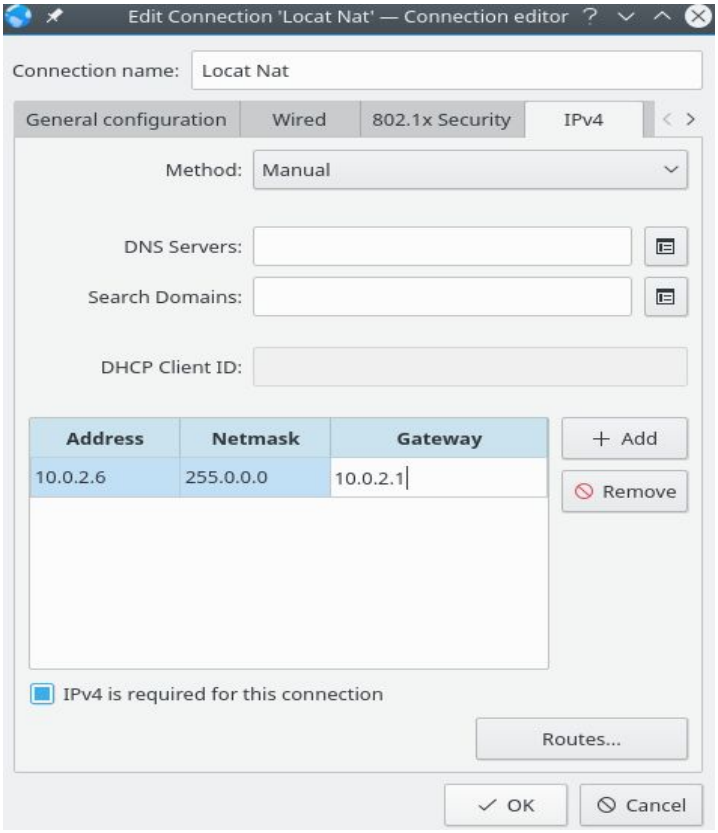
la récupération de la gateway se fait sur une des machines virtuelles via la commande.

sudo route -n

cela donne le subnet

10.0.2.* et GW 10.0.2.1

Il ne reste qu'à configurer le réseau sur les machines en fonction en définissant statiquement les adresses ip.



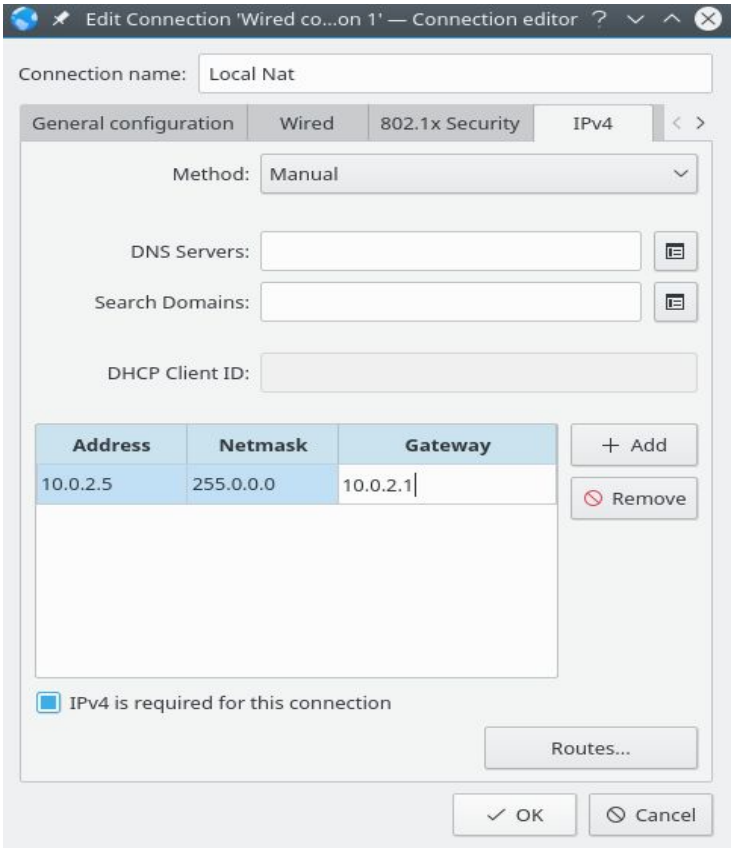
Serveur/routeur :

eth0 : Réseau Nat

Address: 10.0.2.5

NetMask: 255.0.0.0

GateWay :10.0.2.1



Client :

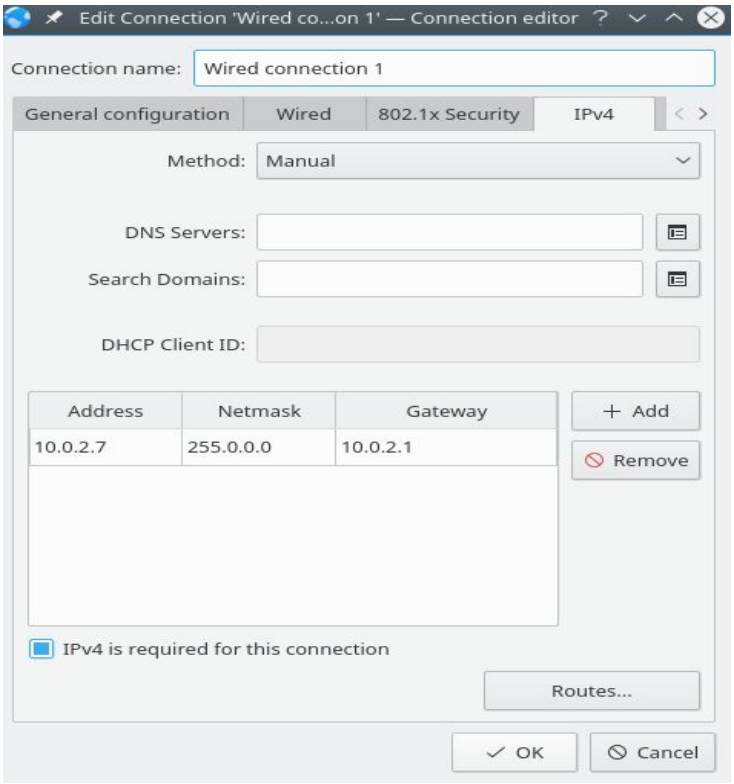
carte Réseau:

eth0: Réseau Nat

Address : 10.0.2.6

NetMask : 255.0.0.0

GateWay:10.0.2.1



Intrus :

Address : 10.0.2.7

NetMask : 255.0.0.0

GateWay:10.0.2.1

IANA Internet Assigned Numbers Authority a réservé 3 espaces d'adressage privé **représentant le LAN**.

10.0.0.0 - 10.255.255.255 (10/8 prefix)

172.16.0.0 - 172.31.255.255 (172.16/12 prefix)

192.168.0.0 - 192.168.255.255 (192.168/16 prefix)

Dans un sous réseau ex: 192.168.16.0/24 , 192.168.16.255 représente le broadcast , toutes les machines (192.168.16.1 -> 254) sur ce sous réseau recevront le message transmis sur ce channel.

Le **Wan est donc l'ensemble des adresses excluant LAN**.

votre adresse ip chez vous à la maison est de type privée, c'est une astuce pour économiser les adresses ip.

La norme IPV4 mise en place à la fin des années 80, n'avaient pas prévu que le parc des machines dépasserait les 4.3 Milliards d'adresses ip disponibles.

IPV4 -> 4 Bytes où 1 Byte -> 8 bit , 8 bit => 2^8 possibilités soit 256. Soit un max de (2^8)^4

Nous sommes passé à L'IPV6 -> 6 Bytes soit un Max de (2^8)^6

mais les 2 doivent coexister car beaucoup de systèmes fonctionnent encore en IPV4.

derrière un routeur il y a un réseau privé contenant toutes vos machines , l'attribution des adresses ip se fait via le DHCP Dynamic Host Configuration Protocol et les messages arrivent à destination grâce au Nat Network Address Translation.

le routeur NAT possède une table de clé-valeur de conversion pour effectuer la translation d'adresse.

La machine interne (à voir ici au sens d'un réseau privé), possédant l'adresse IP 10.101.10.20 est traduite en 193.48.100.174 quand elle converse avec le monde extérieur au travers du routeur NAT.

Le réseau privé n'est pas visible par les autres machines.Il apparait comme une seule machine. c'est à dire que l'ensemble du trafic à destination du réseau privé passe par le routeur.

On remarque que la gateway sur Serveur,Intrus et Client est 10.0.2.1, Cela signifie que

10.0.2.1 est un pont entre le réseau local **Lan (Local Area Network)** et le réseau Internet **Wan (World Area Network)**.

Vérification de l'installation

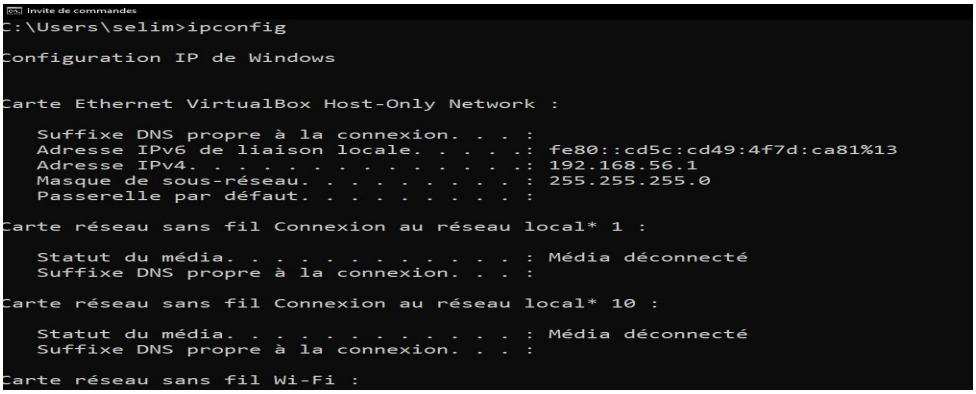
Un outil est mis à votre disposition pour vérifier l'accessibilité à une machine.

la commande ping.

```
serveur@linux-n1x1:~> ping 10.0.2.6
PING 10.0.2.6 (10.0.2.6) 56(84) bytes of data.
64 bytes from 10.0.2.6: icmp_seq=1 ttl=64 time=0.599 ms
64 bytes from 10.0.2.6: icmp_seq=2 ttl=64 time=0.761 ms
64 bytes from 10.0.2.6: icmp_seq=3 ttl=64 time=0.478 ms
```

```
serveur@linux-n1x1:~> ping 10.0.2.7
PING 10.0.2.7 (10.0.2.7) 56(84) bytes of data.
64 bytes from 10.0.2.7: icmp_seq=1 ttl=64 time=0.461 ms
64 bytes from 10.0.2.7: icmp_seq=2 ttl=64 time=0.344 ms
64 bytes from 10.0.2.7: icmp_seq=3 ttl=64 time=0.433 ms
```

on remarque qu'au départ de Serveur on a accès à Client (10.0.2.6) et Intrus (10.0.2.7).



la commande ***ipconfig*** sous windows

ip adresse, subnet

adresse de

broadcast

```
serveur@linux-n1x1:~> ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth9: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 10
00
    link/ether 08:00:27:14:4a:b6 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.5/8 brd 10.255.255.255 scope global eth9
       valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe14:4ab6/64 scope link
       valid_lft forever preferred_lft forever
```

ip addr Sous Linux nous donnes des informations complémentaires sur le réseau

On peut commencer.

Après quelques recherches sur le partage de fichiers sous linux, on se retrouve avec beaucoup d'options dont voici une liste non exhaustive.

Samba ,Cross Plateforme c'est à dire utilisable sur Windows et Linux

NFS Network File System , Uniquement Unix et Linux ou conjointement à Samba

UPNP permet de partager avec des XBox, Playstation ou stations/serveurs Linux.

SSHFS SSH File System , permet le partage au travers d'une connection SSH

FTP File Transfert Protocol, les données transitent en clair (ASCII , "A" vaut 65 , etc..)

HTTP Serveur web local , utilisable avec python -m site 80 dans le dossier à partager

Peer 2 Peer en utilisant le logiciel croc

Après discussion avec mon professeur , j'ai décidé de travailler avec FTP.

Le fait que les données transitent en clair le rend plus intéressant au niveau de l'analyse de la sécurité , la mise en avant de **principe de chiffrement**, la **mise en confinement (CHROOTING)** de certains utilisateurs et l'**importance d'une bonne gestion des fichiers de configuration**.

Attention

Il est obligatoire de désactiver le pare-feu sur la machine serveur, dans le contexte que j'ai mis en place.En effet le parefeu de Opensuse est juste obsolète, on en utilisera un qui fera le travail

démarrer > firewall > Stop Firewall Now & Disable Firewall Automatic Starting

FTP File Transfert Protocol

ftp est installé sur windows 10 et openSuse 42.3 nativement.

l'option -h nous donne une bonne description de ce qu'on peut faire

```
C:\Users\selim>ftp -h

Transfère des fichiers vers et depuis un ordinateur avec un service
de serveur FTP activé (quelquefois appelé un démon : daemon).
Ftp peut être utilisé interactivement.

FTP [-v] [-d] [-i] [-n] [-g] [-s:filename] [-a] [-A] [-x:sendbuffer]
    [-r:recvbuffer] [-b:asyncbuffers] [-w:windowsize] [host]

  -v              Supprime l'affichage des réponses du serveur distant.
  -n              Supprime la connexion automatique au démarrage.
  -i              Désactive l'invite s'il y a plusieurs fichiers à transférer.
  -d              Active le débogage.
  -g              Désactive le globbing du nom de fichier (cf. commande GLOB).
  -s:NomFich      Spécifie un fichier texte contenant les commandes FTP ; ces
                  commandes seront automatiquement exécutées après le
                  démarrage de FTP.
  -a              Utilise n'importe quelle interface locale pour la liaison de
                  la connexion des données.
  -A              Connexion en tant qu'anonyme.
  -x:send sockbuf Remplace la taille SO_SNDBUF de 8192 par défaut.
  -r:recv sockbuf Remplace la taille SO_RCVBUF de 8192 par défaut.
  -b:async count  Remplace le compteur asynchrone de 3 par défaut.
  -w:TailleFenêt Remplace la taille par défaut du tampon de transfert de
                  65535.
  hôte            Spécifie le nom de l'hôte ou l'adresse IP de l'hôte distant
                  auquel se connecter.
```

On est sur la bonne route, **ftp sert au transfert de fichiers vers et depuis un ordinateur**.

Il est aussi fait mention ici, de service ftp appelé Daemon ou ftpd.

un **Daemon est un programme qui s'exécute en arrière plan** plutôt que sous le contrôle direct d'un utilisateur.

On distingue donc 2 "versions" de ftp:

ftp => **client FTP**

ftpd => **Serveur FTP**

comme dit plus haut ftp transfert les données en clair et la question de sécurité s'est posée assez rapidement.

2 réponses furent apportées : SFTP et FTPS

à première vue c'est la même chose mais, il n'en est rien.

SFTP fonctionne au travers d'un service SSH (Secure Shell) fournit par exemple par openssh.

FTPS quant-à-lui fonctionne au travers d'un Secure Socket Layer (SSL) devenu Transport Layer Socket (TLS).

Expliquer les différences entre SSH et SSL pourrait être le sujet d'un rapport, ce qu'il faut retenir c'est :

SSL est une technique de sécurisation des communications fonctionnant avec un système de certificats, de signatures et d'échange de clés. ne nécessitant pas l'authentification côté serveur Exemple HTTPS

SSH est un programme mais aussi un protocole fonctionnant sur un système d'échange de clés.Il a besoin d'un compte authentifié pour fonctionner et s'utilise pour FTP, IMAP,POP,SMTP,TELNET,etc…

En Gros :

SSL permet de sécuriser le transport d'information via le web alors que SSH est une plateforme de sécurisation pour toutes formes de communications électroniques. ref

Au niveau du choix d'implémentation SFTP ou FTPS, ce site nous enjoint à préférer SFTP donc (SSH) pour une bonne raison.Le FIREWALL. Toutes les communications se font au travers d'un seul port (22 par défaut) , ce qui rend la configuration du FW assez simple.

J'ai donc choisis d'implémenter SFTP.

La connection sécurisée depuis le client se fera donc via la commande:

sftp user@ip_server_or_hostname

coté Serveur :

J'ai dit plus haut il nous faut un firewall capable de gérer tout ça, je vais vous en présenter un facile à prendre en main et gratuit.

UFW

on l'installe

sudo zypper in ufw

sudo ufw enable

on définit les règles par défaut

sudo ufw default deny incoming #rien ne rentre

sudo ufw default allow outgoing #tout peut sortir

on définit nos règles pour le ftp et le ssh

sudo ufw allow ftp

sudo ufw allow ssh

sudo ufw reload

et c'est tout il est prêt.

tant qu'on parle réseau , je vais terminer en expliquant que ftp fonctionne sur le protocole tcp.

à la différence de l'udp , le tcp attend un retour (un accusé de réception). si

les paquets n'ont pas tous été envoyés on renvoie. on utilisera udp par exemple pour les jeux vidéos (capture de la souris,appui des touches) (très rapide), et tcp quand on désire vérifier qu'un message a bien été reçu et était complet.

VSFTPD

Il nous faut un service ftp (FTPD) , la commande

sudo systemctl status ftpd

```
serveur@linux-n1x1:~> sudo systemctl status ftpd
● ftpd.service
   Loaded: not-found (Reason: No such file or directory)
   Active: inactive (dead)
```

nous indique que nativement aucun service ftpd existe,on va donc en installer un.

sudo zypper in vsftpd

on va faire en sorte qu'il se lance au démarrage et le configurer

sudo systemctl enable vsftpd

on fait une copie du fichier de configuration

sudo cp /etc/vsftpd.conf /etc/vsftpd.conf.old

en l'état les connections anonymes sont autorisées et personne n'est chrooté.

sudo systemctl start vsftpd

Tentons la connection depuis le client (10.0.2.6)

```
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd /var/www
250 Directory successfully changed.
ftp> get .env
local: .env remote: .env
229 Entering Extended Passive Mode (|||30077|).
150 Opening BINARY mode data connection for .env (0 bytes).
    0        0.00 KiB/s
226 Transfer complete.
ftp> quit
221 Goodbye.
serveur@linux-n1x1:~/Documents> ls -la
total 0
drwxr-xr-x 1 serveur users  28 May 15 01:24 .
drwxr-xr-x 1 serveur users 644 May 15 01:23 ..
-rw-r--r-- 1 serveur users   0 May 15 01:12 .env
```

Cela permet à n'importe qui de télécharger des fichiers sensibles , de configuration ou des bases de données chiffrées ou non. c'est un problème majeur de sécurité qu'on va corriger. rem: .env ne contenait rien OUF! on aura pas toujours cette chance

on modifie le fichier vsftpd.conf

sudo vim /etc/vsftpd.conf

#on veut pouvoir effectuer des commandes d'écritures

write_enable=YES

#on veut recevoir l'information lors d'un changement de répertoire

dirmessage_enable=YES

#on définit la bannière de connection c'est important: par défaut la version est donnée.

```
Connected to 10.0.2.5.
220 (vsFTPd 3.0.2)
331 Please specify the password.
```

ftpd_banner=Welcome to 49853 FTP service.

#accès que via des comptes locaux

local_enable=YES

#permission à enlever à 777 , le masque par défaut vaut 077 => 700

local_umask=022 # mène à des permissions 755 (lire + Exec)

#chroot les utilisateurs à leur répertoire /home/user

chroot_local_user=YES

#on a pas de liste d'exclusion des chroot => tout le monde est chrooté

chroot_list_enable=NO

#on définit le répertoire racine pour le chroot

local_root=/home

#on leur permet d'écrire dans leurs dossiers

allow_writeable_chroot=YES

#on désactive les connections anonymes

anonymous_enable=NO

#on active les logs

syslog_enable=YES

#on confirme qu'on se connecte sur le port 20 (ftp-data)

connect_from_port_20=YES

#on dit qu'on fonctionne en ipv4

listen=YES

#on doit rejeter ipv6

listen_ipv6=NO

on sauvegarde , et on relance le service

sudo systemctl restart vsftpd

```
● vsftpd.service - Vsftpd ftp daemon
   Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; enabled; vendor preset: disabled)
   Active: active (running) since Fri 2020-05-15 00:19:42 CEST; 7s ago
Main PID: 6192 (vsftpd)
    Tasks: 1 (limit: 512)
   CGroup: /system.slice/vsftpd.service
           └─6192 /usr/sbin/vsftpd /etc/vsftpd.conf
```

Nous avons posé les bases:

nous devons créer un groupe pour les ftpusers, un utilisateur et lui assigner un espace dans /home avec les bons droits.

on créer le groupe ftpusers

sudo groupadd ftpusers

on rend notre serveur dissociable des ftpusers par règle d'accès

sudo chmod 750 ~

bien sûr on aura vérifié que ~ appartient à un groupe différent de ftpusers

//sudo chgrp nonftpusers ~

on ajoute l'utilisateur toto

sudo useradd toto

on créer le dossier de toto

sudo mkdir /home/toto

on définit les règles d'accès

7 donne tous les droits au propriétaire 5 donne le droit en lecture et en parcours à son groupe 0 on refuse aux autres l'accès en lecture/écriture ou parcours (on pourrait imaginer que plusieurs groupes existent sur la même machine ex: Étudiants/Professeurs)

sudo chmod 750 /home/toto

on rend à toto ce qui appartient à toto, toto devient propriétaire du dossier

sudo chown toto /home/toto

on fait en sorte que toto soit dans le groupe des ftp users

sudo chgrp ftpusers /home/toto

```
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd /var/www
550 Failed to change directory.
ftp> pwd
Remote directory: /
ftp>
```

voilà que toto n'a plus accès aux informations sensibles à la racine Mais aussi aux fichiers de notre serveur.

```
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> dir
229 Entering Extended Passive Mode (|||30056|).
150 Here comes the directory listing.
drwxr-xr-x    1 1003     1000           12 May 15 00:38 selim
drwx------    1 1000     100           652 May 15 00:02 serveur
drwxr-xr-x    1 1002     1000           22 May 15 00:43 toto
226 Directory send OK.
ftp>
```

selim et toto peuvent échanger des données sans problème, tout en garantissant que personne d'autre que selim ne peut écrire dans son dossier.

toto peut uploader dans son répertoire

```
serveur@linux-n1x1:~/Documents> ftp toto@10.0.2.5
Connected to 10.0.2.5.
220 Welcome to 49853 FTP service.
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd selim
250 Directory successfully changed.
ftp> put msg_de_toto.txt
local: msg_de_toto.txt remote: msg_de_toto.txt
229 Entering Extended Passive Mode (|||30044|).
553 Could not create file.
ftp> quit
221 Goodbye.
serveur@linux-n1x1:~/Documents>
```

toto ne peut pas upload dans le répertoire de sélim

```
serveur@linux-n1x1:~/Documents> ftp selim@10.0.2.5
Connected to 10.0.2.5.
220 Welcome to 49853 FTP service.
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd toto
250 Directory successfully changed.
ftp> get msg_de_toto.txt
local: msg_de_toto.txt remote: msg_de_toto.txt
229 Entering Extended Passive Mode (|||30055|).
150 Opening BINARY mode data connection for msg_de_toto.txt (34 bytes).
100% |***********************************************************************|    34      638.52 KiB/s    00:00 ETA
226 Transfer complete.
34 bytes received in 00:00 (32.67 KiB/s)
ftp> quit
221 Goodbye.
serveur@linux-n1x1:~/Documents>
```

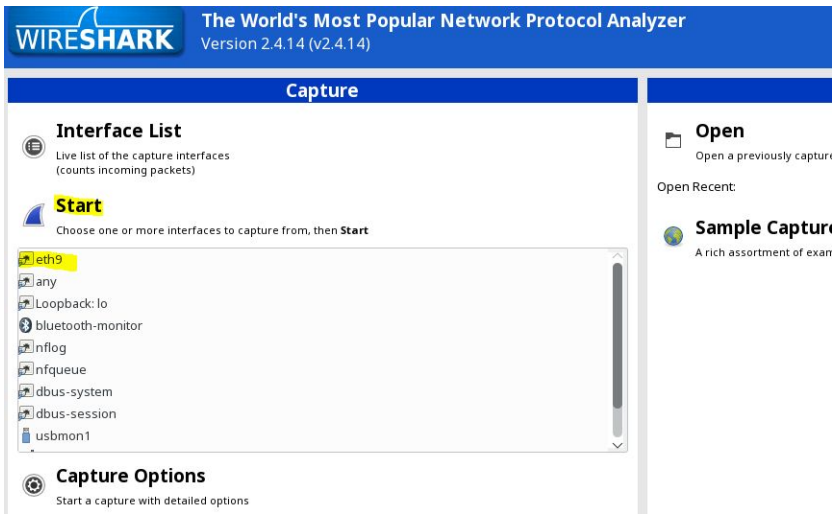
selim peut récupérer le message de toto dans le répertoire de toto.

a ce stade on a permis l'échange de données avec un niveau satisfaisant de sécurité au niveau des permissions/accès mais qu'en est il du réseau ?

Voyons ça avec un logiciel gratuit.

WireShark

sudo zypper in wireshark



on lance la capture sur notre seule interface eth 9

No.	Time	Source	Destination	Protocol	Length	Info
3	0.000626141	10.0.2.6	10.0.2.5	TCP	68	41646 → 21 [ACK] Seq=1 Ack=1 Win=29200 Len=0 TSval=3959059 TSecr=3959770
4	0.003335926	10.0.2.5	10.0.2.6	FTP	103	Response: 220 Welcome to 49853 FTP service.
5	0.003882147	10.0.2.6	10.0.2.5	TCP	68	41646 → 21 [ACK] Seq=1 Ack=36 Win=29200 Len=0 TSval=3959060 TSecr=3959771
6	0.004218491	10.0.2.6	10.0.2.5	FTP	79	Request: USER toto
7	0.004239704	10.0.2.5	10.0.2.6	TCP	68	21 → 41646 [ACK] Seq=36 Ack=12 Win=29056 Len=0 TSval=3959771 TSecr=3959060
8	0.004320823	10.0.2.5	10.0.2.6	FTP	102	Response: 331 Please specify the password.
9	0.043541899	10.0.2.6	10.0.2.5	TCP	68	41646 → 21 [ACK] Seq=12 Ack=70 Win=29200 Len=0 TSval=3959070 TSecr=3959771
10	3.803339637	10.0.2.6	10.0.2.5	FTP	84	Request: PASS selim1234
11	3.841461929	10.0.2.5	10.0.2.6	TCP	68	21 → 41646 [ACK] Seq=70 Ack=28 Win=29056 Len=0 TSval=3960731 TSecr=3960009
12	3.870531816	10.0.2.5	10.0.2.6	FTP	91	Response: 230 Login successful.
13	3.870832102	10.0.2.6	10.0.2.5	TCP	68	41646 → 21 [ACK] Seq=28 Ack=93 Win=29200 Len=0 TSval=3960026 TSecr=3960738
14	3.870843517	10.0.2.6	10.0.2.5	FTP	74	Request: SYST
15	3.870848948	10.0.2.5	10.0.2.6	TCP	68	21 → 41646 [ACK] Seq=93 Ack=34 Win=29056 Len=0 TSval=3960738 TSecr=3960026
16	3.870929134	10.0.2.5	10.0.2.6	FTP	87	Response: 215 UNIX Type: L8

Ay ay ! c'est le drame voilà qu'on récupère un identifiant et un mot de passe.

```
72 143.0379365 10.0.2.6          10.0.2.5          FTP-DAT  102 FTP Data: 34 bytes
73 143.0379494 10.0.2.6          10.0.2.5          TCP       68 35276 → 30001 [FIN, ACK] Seq=35 Ack=1 Win=29200 Len=0 TSval=3994818 TSecr=3995529

0000  00 00 00 01 00 06 08 00  27 71 87 74 00 00 08 00   ........ 'q.t....
0010  45 08 00 56 01 3e 40 00  40 06 21 52 0a 00 02 06   E..V.>@. @.!R....
0020  0a 00 02 05 89 cc 75 31  c7 22 f5 ff e4 64 68 4c   ......u1 ."...dhL
0030  80 18 39 08 62 b8 00 00  01 01 08 0a 00 3c f4 c2   ..9.b... .....<..
0040  00 3c f7 89 42 6f 6e 6a  6f 75 72 20 76 6f 69 6c   .<..Bonj our voil
0050  c3 a0 20 75 6e 20 66 69  63 68 69 65 72 20 64 65   .. un fi chier de
0060  20 74 6f 74 6f 0a                                  toto.
```

Oula ! et les contenus des fichiers qui transitent sont pratiquement lisibles , **Bonj our voil un fi chier de toto.**

On voit donc ici que la communication entre 10.0.2.5 et 10.0.2.6 n'est pas du tout sécurisée , cela peut se résoudre assez simplement il nous faudra un outil coté Serveur

OpenSSH : logiciel libre de SSH

sudo zypper in openssh

sudo systemctl enable sshd

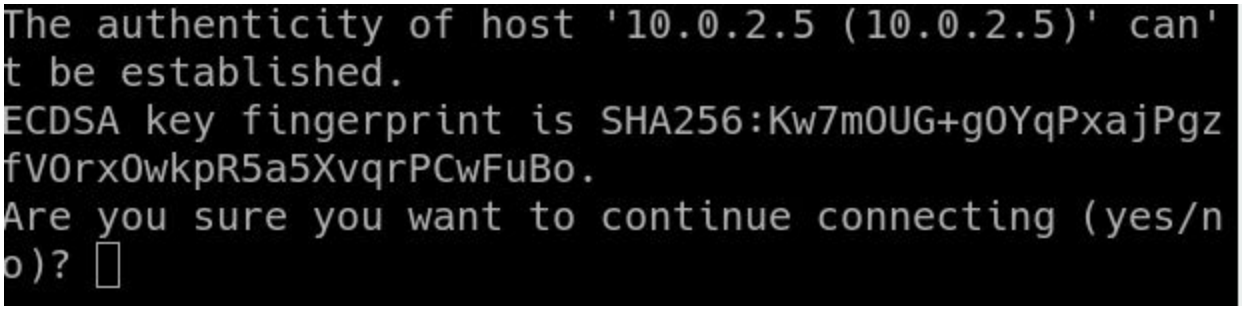
Le seul paramètre un peu gênant c'est le permit root login , qui nous permet de se connecter en root à la machine.

Je laisse ça à la convenance du lecteur, en fonction de ses besoins il convient de changer cela à NO dans /etc/ssh/sshd_config.

on peut lancer le service et l'essayer

sudo systemctl start sshd

sftp toto@10.0.2.5 depuis 10.0.2.6 (Client) nous donne ceci



yes l'ajoute à nos known_hosts tandis que no casse la connection.

Alors comment ça fonctionne ?

Le principe est assez simple , Il y a 2 clefs.

Une privée qu'on garde précieusement et une publique qu'on donne à qui la veut.

La clef Publique est utilisé par notre correspondant pour chiffrer le message qui nous est destiné.

Grâce à notre clef privée nous pouvons déchiffrer le message.

une des techniques les plus souvent utilisée est de placer la clef publique du serveur en dur sur les machines clientes.

Disposant de la clef elles chiffrent à la première transmission, ce qui n'est pas le cas quand on a le handshake (Phase d'échange de clefs).

Il faut savoir que ce système est dit asymétrique, il y a 2 clefs et les calculs sont gourmands.

Cette phase Asymétrique (publique/privée) est souvent remplacé au fil de la transmission par une transmission Symétrique.

les 2 partis échangent de manière crypté via le premier système , les informations nécessaire à la création de la clef de session/symétrique qui sera utilisée pour chiffrer et déchiffrer.

Ce qu'il faut savoir c'est qu'ici on se partage une information sensible, si on ne communique pas avec la bonne personne tout ce qu'on a fait avant tombe à l'eau.

soit un Attaquant sur le réseau , au moment ou vous échanger les clés publique il vous donne la sienne et vous vous lui donner la vôtre pensant que c'est le serveur.

Il est donc en mesure de déchiffrer tout ce que vous lui envoyer grâce à sa clé privée.

Il va rechiffrer avec la clé publique du serveur, transmettre la requête et attendre la réponse.

Il va ensuite déchiffrer la réponse du serveur et vous envoyer la réponse chiffrer avec votre clé.

Ni vu ni Connu.

L’attaquant est un relai entre vous et votre correspondant capable de lire tout ce que vous échangé.

L’attaque présentée ici est un version améliorée de la MITM (Man In The Middle) aussi appelé ARP Poisoning/Spoofing.

Elle utilise une vulnérabilité liée au Protocole ARP.

ARP

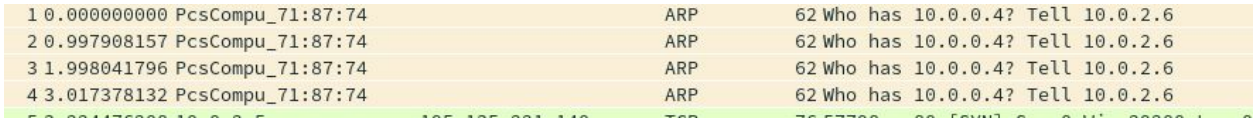
Address Request Protocol

L'adresse MAC est l'identifiant unique de votre carte réseau.

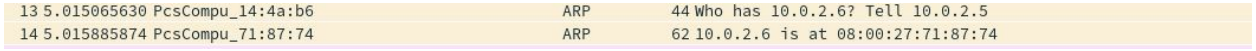
L’adresse IP est l’identifiant sur le réseau on l’a vu plus haut.

Sur un réseau privé (RFC1918) donc de forme 10.0.0.0/8 par exemple,

les machines savent qu’elles ne doivent pas utiliser la passerelle, elles font donc une Requête sur le channel Broadcast FF:FF:FF:FF:FF:FF destiné donc à l’ensemble du Subnet pour savoir qui a l’adresse ex : 10.0.0.4. Host unreachable dans mon contexte



si j’essaie avec 10.0.2.6 depuis serveur



Là une réponse est envoyée 10.0.2.6 est à 08:00:27:71:87:74

cette valeur peut changer aucune machine n’est au courant si le réseau est statique.

à moins de faire un ARP binding

arp -s [ip] [mac Address]

```
serveur@linux-n1x1:~> sudo arp -s 10.0.2.1 52:54:00:12:35:00
[sudo] password for root:
serveur@linux-n1x1:~> sudo arp -a
? (10.0.2.6) at 08:00:27:71:87:74 [ether] on eth9
? (10.0.2.1) at 52:54:00:12:35:00 [ether] PERM on eth9
serveur@linux-n1x1:~>
```

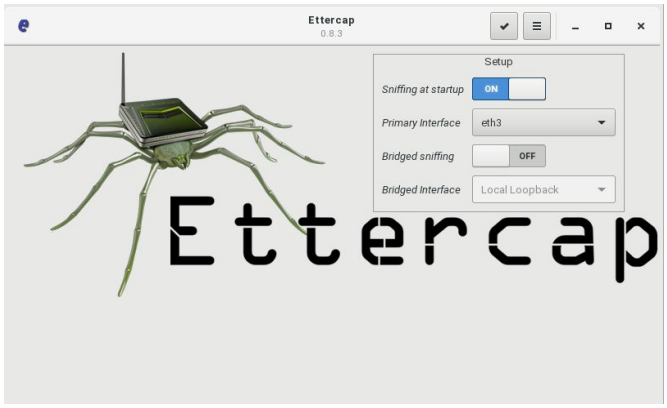
il n'y aura plus de mise à jour de la table ARP pour joindre 10.0.2.1

Mais voyons plus en détail ce qu'est un ARP Spoofing

J'installe sur la machine intruse un logiciel Ettercap,

on nous souffle quelques "à vos risque et périls" durant l'installation.

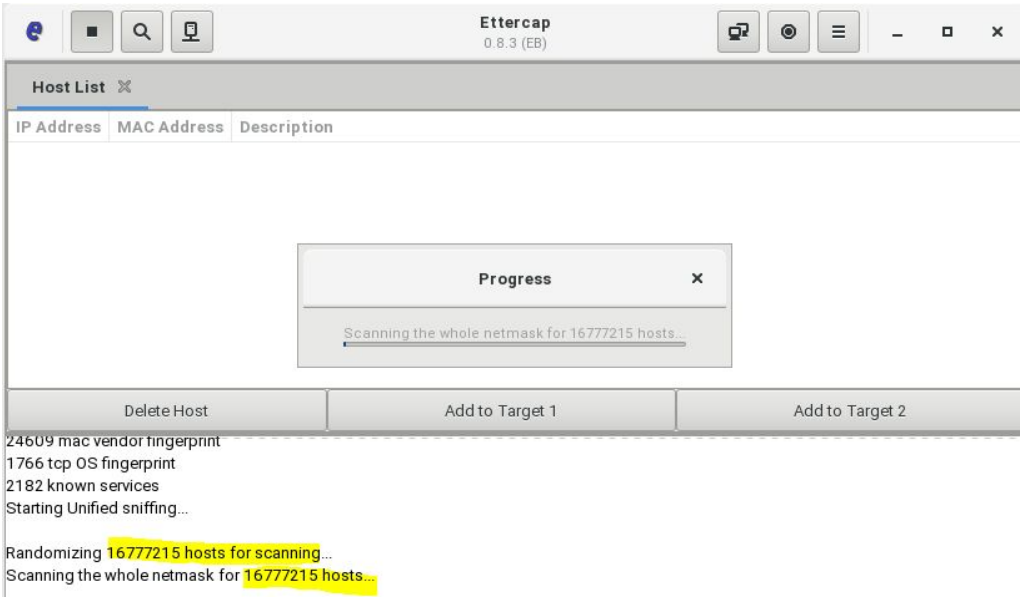
installable via yast à cette [adresse](#).



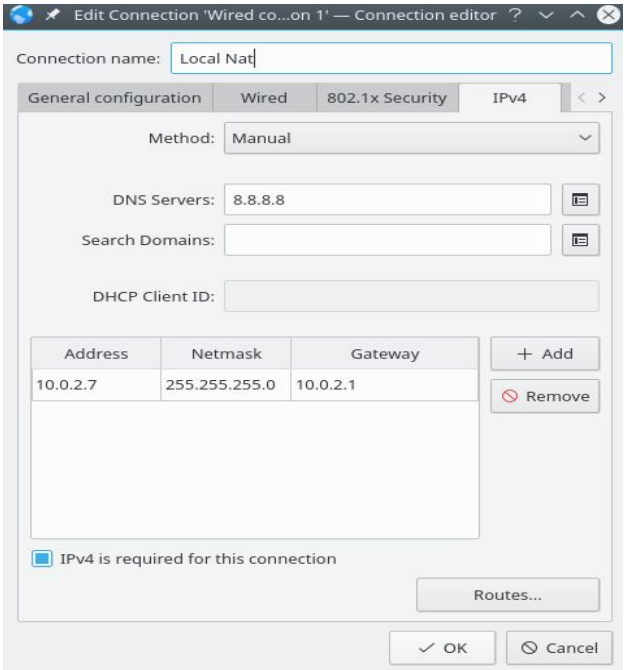
maintenant qu'on a fait entrer le loup dans la bergerie, observons depuis le serveur via WireShark les paquets qui vont transiter sur le réseau pour comprendre.

309	157.2606813	PcsCompu_f9:c7:35		ARP	62 Who has 10.149.244.33? Tell 10.0.2.7
310	157.2707790	PcsCompu_f9:c7:35		ARP	62 Who has 10.139.83.6? Tell 10.0.2.7
311	157.2819691	PcsCompu_f9:c7:35		ARP	62 Who has 10.172.162.43? Tell 10.0.2.7
312	157.2920643	PcsCompu_f9:c7:35		ARP	62 Who has 10.193.187.191? Tell 10.0.2.7
313	157.3021712	PcsCompu_f9:c7:35		ARP	62 Who has 10.223.208.182? Tell 10.0.2.7
314	157.3122432	PcsCompu_f9:c7:35		ARP	62 Who has 10.164.156.38? Tell 10.0.2.7
315	157.3227320	PcsCompu_f9:c7:35		ARP	62 Who has 10.87.139.210? Tell 10.0.2.7
316	157.3333833	PcsCompu_f9:c7:35		ARP	62 Who has 10.135.114.118? Tell 10.0.2.7
317	157.3434749	PcsCompu_f9:c7:35		ARP	62 Who has 10.174.146.115? Tell 10.0.2.7
318	157.3535649	PcsCompu_f9:c7:35		ARP	62 Who has 10.107.236.6? Tell 10.0.2.7
319	157.3636764	PcsCompu_f9:c7:35		ARP	62 Who has 10.50.3.60? Tell 10.0.2.7
320	157.3737665	PcsCompu_f9:c7:35		ARP	62 Who has 10.205.229.251? Tell 10.0.2.7
321	157.3843158	PcsCompu_f9:c7:35		ARP	62 Who has 10.82.131.144? Tell 10.0.2.7
322	157.3944389	PcsCompu_f9:c7:35		ARP	62 Who has 10.39.127.22? Tell 10.0.2.7
323	157.4044983	PcsCompu_f9:c7:35		ARP	62 Who has 10.40.237.92? Tell 10.0.2.7
324	157.4149629	PcsCompu_f9:c7:35		ARP	62 Who has 10.208.43.115? Tell 10.0.2.7
325	157.4251710	PcsCompu_f9:c7:35		ARP	62 Who has 10.249.184.206? Tell 10.0.2.7
326	157.4357044	PcsCompu_f9:c7:35		ARP	62 Who has 10.142.130.140? Tell 10.0.2.7
327	157.4457871	PcsCompu_f9:c7:35		ARP	62 Who has 10.171.10.104? Tell 10.0.2.7
328	157.4559020	PcsCompu_f9:c7:35		ARP	62 Who has 10.127.139.216? Tell 10.0.2.7
329	157.4660309	PcsCompu_f9:c7:35		ARP	62 Who has 10.8.85.16? Tell 10.0.2.7
330	157.4760916	PcsCompu_f9:c7:35		ARP	62 Who has 10.236.66.4? Tell 10.0.2.7
331	157.4866540	PcsCompu_f9:c7:35		ARP	62 Who has 10.255.105.18? Tell 10.0.2.7
332	157.4966630	PcsCompu_f9:c7:35		ARP	62 Who has 10.147.201.189? Tell 10.0.2.7
333	157.5068720	PcsCompu_f9:c7:35		ARP	62 Who has 10.48.254.1? Tell 10.0.2.7
334	157.5169582	PcsCompu_f9:c7:35		ARP	62 Who has 10.155.200.255? Tell 10.0.2.7
335	157.5270121	PcsCompu_f9:c7:35		ARP	62 Who has 10.128.154.115? Tell 10.0.2.7
336	157.5375800	PcsCompu_f9:c7:35		ARP	62 Who has 10.61.58.77? Tell 10.0.2.7

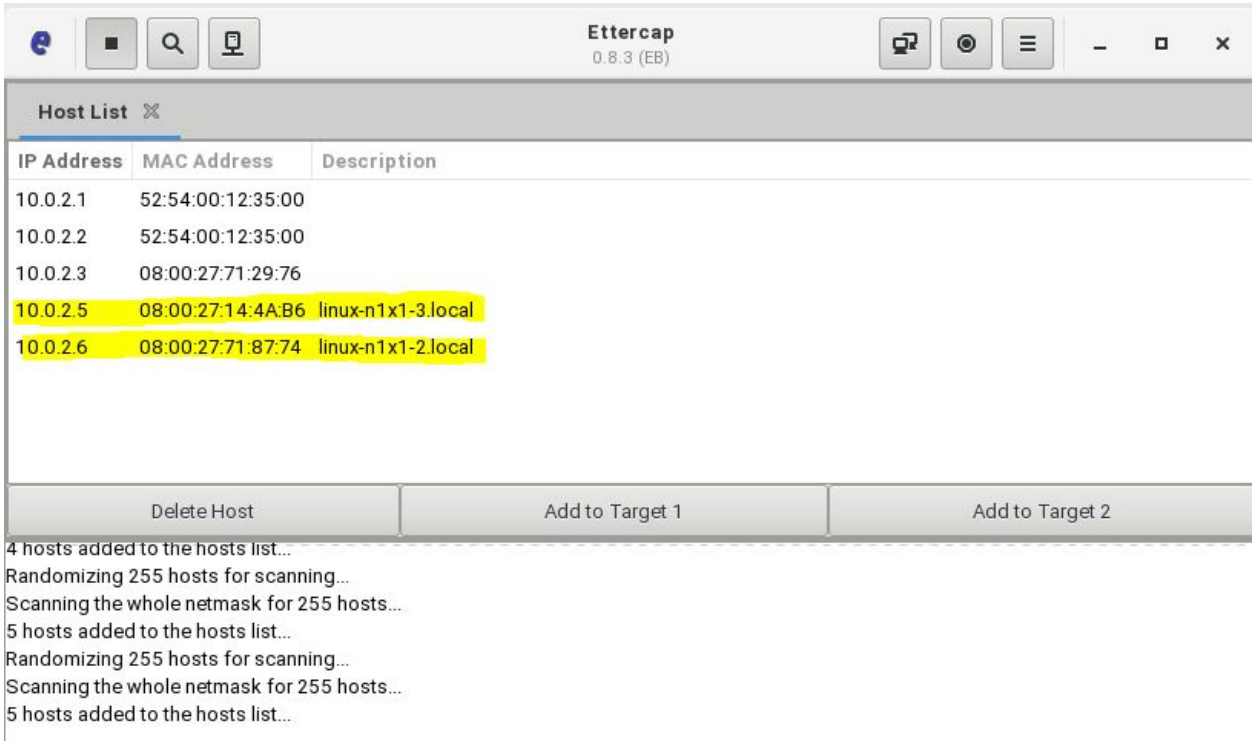
On remarque ici que le loup fait des recherches randomisée sur l'ensemble du subnet 10.0.0.0/8 afin de trouver les hotes potentiels.



ça risque de lui prendre du temps, surtout qu'il n'y a pas beaucoup de machines sur le réseau. **Ce qu'il fait s'appelle du Sniffing,** il cartographie le réseau et tente de constituer une liste d'hôtes. j'ai choisis un tout petit masque 255.0.0.0 cela veut dire qu'il y a environ 255^3 soit 16 581 375 possibilités.



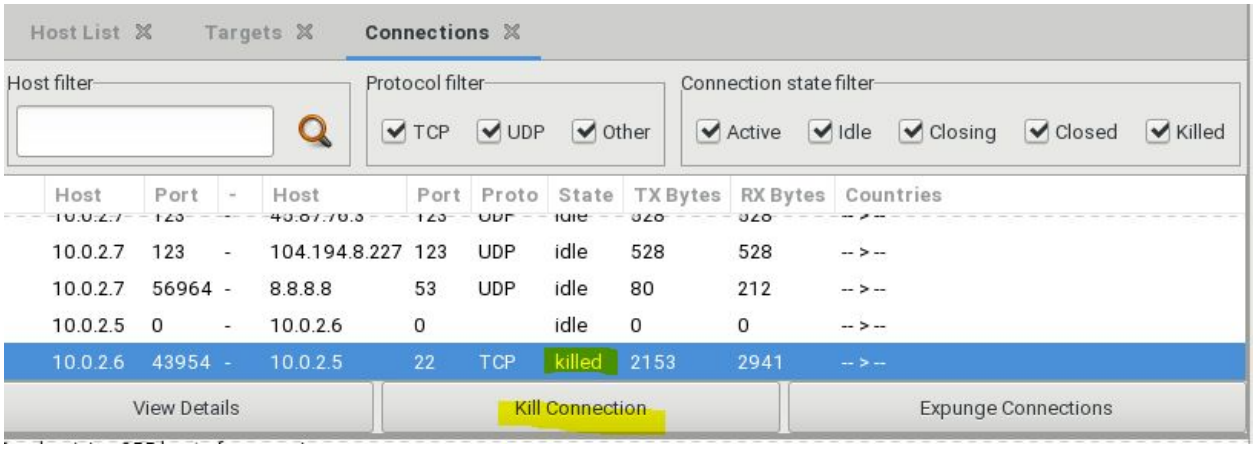
on va modifier le netmask ça va l'aider.



le résultat est immédiat.

Je sélectionne 10.0.2.5 et 10.0.2.6 comme target 1 et target 2 et je lance la MITM.

Je tente une connection depuis client vers serveur en sftp.



dès lors depuis la machine Intruse je peux interrompre la connection entre le client et le serveur.

```
serveur@linux-n1x1:~> sftp selim@10.0.2.5
Password:
Connected to 10.0.2.5.
sftp> packet_write_wait: Connection to 10.0.2.5 port 22: Broken pipe
```

mais ce n'est pas tout, je peux aussi empêcher le client de se connecter au serveur.

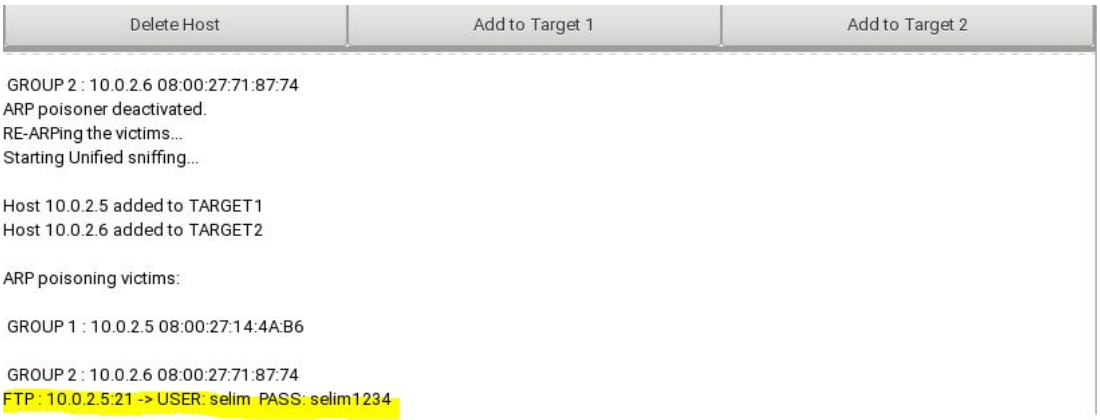
c'est ce qu'on appelle un DoS (denial of service), déni de service.

et il a lui aussi plusieurs formes.

on peut faire un déni de service en surchargeant un des services jusqu'à ce qu'il ne puisse plus apporter de réponses aux requêtes.

Ou comme ici , se placer entre le client et le serveur et faire en sorte que les paquets n'arrivent jamais au serveur.

```
serveur@linux-n1x1:~> sftp selim@10.0.2.5
```



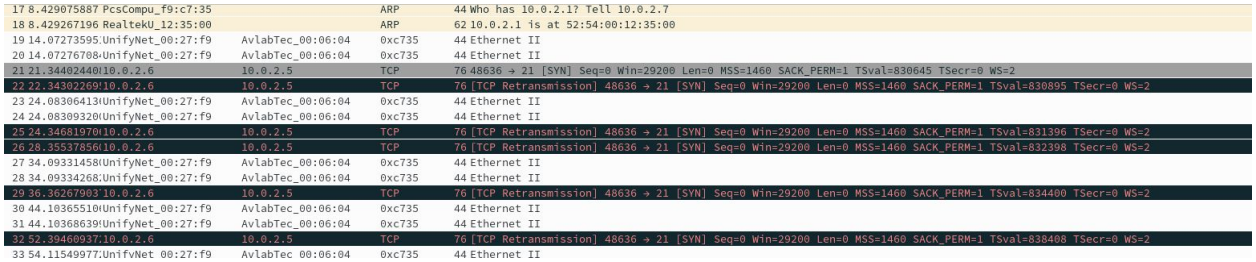
grâce à la MITM , on récupère les identifiants de connection de selim, qui essaie de se connecter via ftp. les paquets transitent par Intruse avant d'atteindre serveur ftp n'étant pas sécurisé il suffit de lire.

à l’inverse on remarque que si il se connecte par sftp , les informations ne sont pas récupérables mais on peut interrompre la connection.

voyons le comportement de l’application après un **ARP binding**

Uniquement Sur le serveur :

arp -s 10.0.2.6 08:00:27:71:87:74

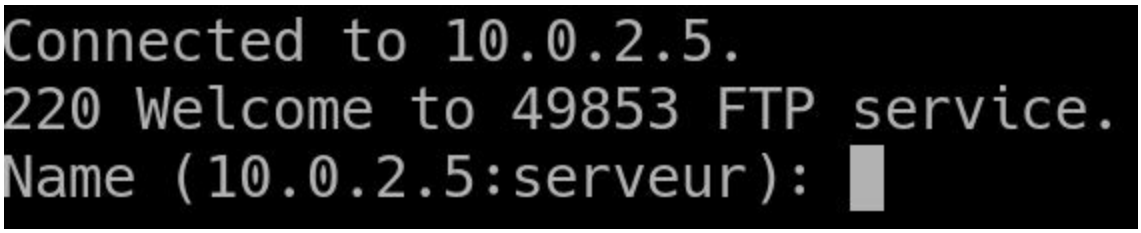


17	8.429075887	PcsCompu_f9:c7:35		ARP	44 Who has 10.0.2.1? Tell 10.0.2.7
18	8.429267196	RealtekU_12:35:00		ARP	62 10.0.2.1 is at 52:54:00:12:35:00
19	14.07273595	UnifyNet_00:27:f9	AvlabTec_00:06:04	0xc735	44 Ethernet II
20	14.07276708	UnifyNet_00:27:f9	AvlabTec_00:06:04	0xc735	44 Ethernet II
21	21.34402440	10.0.2.6	10.0.2.5	TCP	76 48636 → 21 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=830645 TSecr=0 WS=2
22	22.34302269	10.0.2.6	10.0.2.5	TCP	76 [TCP Retransmission] 48636 → 21 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=830895 TSecr=0 WS=2
23	24.08306413	UnifyNet_00:27:f9	AvlabTec_00:06:04	0xc735	44 Ethernet II
24	24.08309320	UnifyNet_00:27:f9	AvlabTec_00:06:04	0xc735	44 Ethernet II
25	24.34681970	10.0.2.6	10.0.2.5	TCP	76 [TCP Retransmission] 48636 → 21 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=831396 TSecr=0 WS=2
26	28.35537856	10.0.2.6	10.0.2.5	TCP	76 [TCP Retransmission] 48636 → 21 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=832398 TSecr=0 WS=2
27	34.09331458	UnifyNet_00:27:f9	AvlabTec_00:06:04	0xc735	44 Ethernet II
28	34.09334268	UnifyNet_00:27:f9	AvlabTec_00:06:04	0xc735	44 Ethernet II
29	36.36267903	10.0.2.6	10.0.2.5	TCP	76 [TCP Retransmission] 48636 → 21 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=834400 TSecr=0 WS=2
30	44.10365510	UnifyNet_00:27:f9	AvlabTec_00:06:04	0xc735	44 Ethernet II
31	44.10368639	UnifyNet_00:27:f9	AvlabTec_00:06:04	0xc735	44 Ethernet II
32	52.39460937	10.0.2.6	10.0.2.5	TCP	76 [TCP Retransmission] 48636 → 21 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=838408 TSecr=0 WS=2
33	54.11549977	UnifyNet_00:27:f9	AvlabTec_00:06:04	0xc735	44 Ethernet II

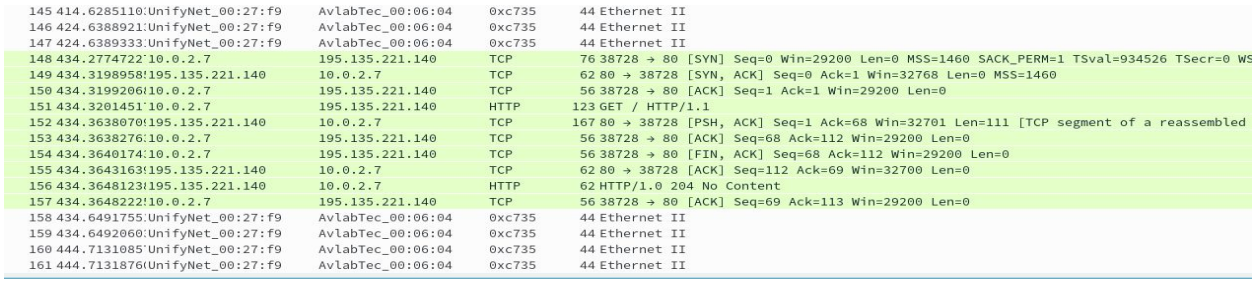
le serveur n’est plus accessible et il y a beaucoup de retransmissions.

si on ajoute un bind sur le client:

arp -s 10.0.2.5 08:00:27:14:4a:b6



il n'y a plus aucun soucis.



Au travers de ces quelques exemples on entrevoit les possibilités et les dangers de ce genre de programmes sur un réseau.

Mais aussi l'importance d'avoir des réseaux publics dissociés de notre réseau privé.

Exemple d'un restaurant qui fournit le wifi

et aussi, le danger derrière les réseaux publics, on ne sait pas qui d'autres est sur le réseau.

=> l'importance d'un réseau de confiance, bien configuré.

Récapitulatif

	Vulnérabilité	Résolution
FTP	Accès données sensibles	CHROOT /Permissions/SFTP
ARP	ARP Poisoning	ARP Binding Bilateral

Conclusion

Au travers de l'installation du parc virtuel , je me suis rendu compte de la charge colossale du travail nécessaire à la sécurisation et la mise en place des différents composants.

A chacune des étapes , des nouveaux problèmes , des nouveaux défis.

Nous avons vu :

Le choix de la configuration réseau optimale tant au niveau de la définition du subnet que des différentes possibilitées de réseau offertes par virtualbox .

Le firewall ufw successeur de SuseFirewall2 .

La gestion des permissions sur les fichiers et les dossiers.

Une manière de Chrooter les utilisateurs tout en leur permettant d'échanger entre eux. Le tout se faisant via l'ajout et la suppression de détails très fins dans les fichiers de configuration.

Chaque maillon de la chaîne est aussi important que les autres et il ne faut en négliger aucun sans quoi on voit son travail réduit à néant.

Beaucoup d'aspects de l'informatique paraissent acquis alors qu'on peut descendre couche après couche IP>ARP>Port>FireWall>MAC>SSH>SSH_CONFIG>CHROOT>TCP

sans savoir du tout où on met les pieds. Je n'avais aucune idée du temps que ça allait me prendre de réaliser ce projet, j'en ai appris énormément sur beaucoup de sujets différents.

Et j'ai pu apporter une réponse automatisée au problème soulevé **Annexe 1**.

partager des fichiers avec certains utilisateurs et chiffrer la communication lors du transport de ces fichiers.

J'ai aussi mis en avant, l'importance de bien évaluer un réseau avant de le juger de confiance et quelques techniques pour se protéger.

la meilleure défense reste notre cerveau et j'espère lui avoir appris quelques petites choses.

Annexe 1 : Solution Automatisée

```
zypper rm -y vsftpd ufw && zypper in -fy vsftpd ufw openssh

systemctl disable SuSEfirewall2 && systemctl stop SuSEfirewall2

systemctl enable ufw && systemctl start ufw

ufw enable

#default rules

ufw default deny incoming && ufw default allow outgoing

#allow ftp and ssh

ufw allow ftp && ufw allow ssh

ufw reload

systemctl enable vsftpd

#copy old version of vsftpd.conf

cp /etc/vsftpd.conf /etc/vsftpd.conf.old && mv vsftpd.conf  /etc/vsftpd.conf

#start service

systemctl start vsftpd

echo "creation group ftpusers"

#create group

groupadd ftpusers

chmod 750 ~

systemctl enable sshd && systemctl start sshd
```

Ajout d'un Ftp User

```
read -p "Name of the user ? " name
cd /home
if [[ -d $name ]]
then
    echo "name already taken"
    exit 1
fi
if [ $(getent group ftpusers) ]
then
    useradd $name
    usermod -aG ftpusers $name
    passwd $name
    mkdir /home/$name
    chown $name /home/$name
    chgrp ftpusers /home/$name
    chmod 755 /home/$name
else
    echo “no group ftpusers”
    exit 1
fi
```

RÉFÉRENCES

1. 7 Default OpenSSH Security Options You Should Change in /etc/ssh/sshd_config 7 feb 2020
2. SSH : Installer et configurer un serveur SSH - Wiki - Wiki 7 feb 2020
3. What is a SSH key fingerprint and how is it generated? 7 feb 2020
4. https://medium.com/bob-kfir-tech/how-to-use-ssh-as-a-vpn-2c16f69dd13 7 feb
5. Documentation - Zone - Predefined Zones 5 feb 2020
6. RFC 1918 - Address Allocation for Private Internets 5 feb 2020
7. The slash after an IP Address - CIDR Notation 5 feb 2020
8. UFW - Community Help Wiki 5 feb 2020
9. https://linuxfr.org/news/creation-d-un-serveur-de-fichiers-sous-ubuntu 13 feb
10. Samba | Reference | openSUSE Leap 15.1 13 feb 2020
11. Mounting samba shares from a unix client 3 mar 2020
12. Penetration Testing of an FTP Server 3 mar 2020
13. How to Prevent Sniffer Attacks with Encrypted FTP 16 mar 2020
14. https://fr.wikipedia.org/wiki/Proxy 16 mar 2020