Configuration et gestion d'un serveur

Chapitre 9 : Héberger des serveurs en toute sécurité

Objectifs :

Comprendre la relation client serveur.
Maître en œuvre ce qu'il faut pour accéder à un serveur héberger chez soi, depuis Internet.

Plan :

Ressources :

http://www.frameip.com

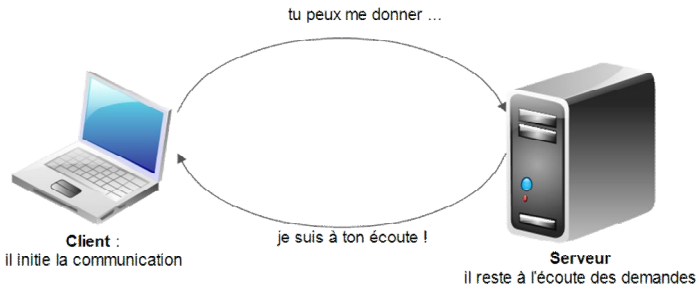
Héberger un serveur suppose l'exposition du réseau local à Internet. Cela aura pour conséquence d'attirer l'attention des hackers.
Pour héberger un serveur, il faut alors observer quelques règles de sécurité.

1. Les architectures client-serveur.

1.1. Qui est le client, qui est le serveur ?

Dès lors qu'une communication est établie entre deux systèmes informatiques, l'un des deux joue le rôle de serveur et l'autre de client.
Par défaut, les serveurs sont de grosses machines puissantes, conçues pour répondre à de fortes sollicitations.

Toutefois, techniquement, ce qui définit un serveur et un client, n'est pas la taille ou la puissance du système, mais son rôle dans la communication :
Le serveur est à l'écoute du réseau, il attend les demandes des clients.
Le client initie la communication. C'est lui qui fait une demande.

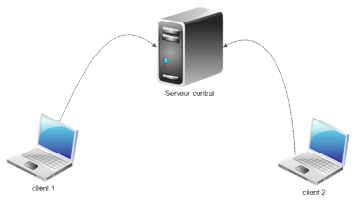


1.2. Les communications évoluées.

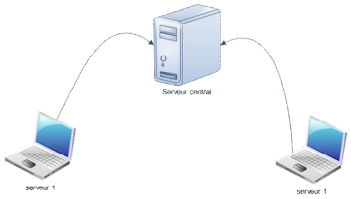
La plupart des échanges client-serveur sont basé sur le modèle précédent, mais pas tous.

Prenons l'exemple du système Skype.

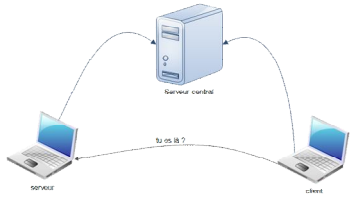
Dans un premier temps, chaque machine qui se connecte, entre en communication avec un serveur central, qui valide l'état connecté du client, et stocke son IP publique.



Une fois cela effectué, cette communication reste en place, pour mettre à jour le statut du client.
Mais en même temps, **chaque système devient serveur** : il est **à l'écoute** des autres systèmes et attend une demande de connexion.



Enfin, dès qu'un système effectue une demande de connexion, pour chater ou faire une visio,
celui-ci prend le rôle de **client** et l'autre système de **serveur** dans cette communication.
Ainsi, chaque système peut jouer le rôle **de client et de serveur**, dans des communications différentes.

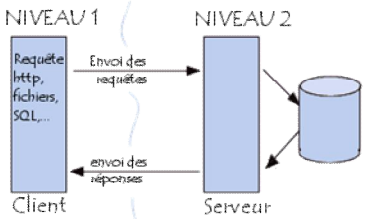


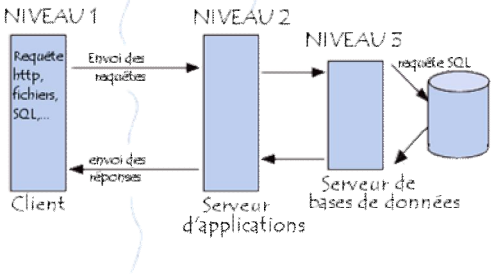
Il en va exactement de même pour les outils de prise en main à distance comme TeamViewer ou UltraVNC.
C'est également le cas pour les outils d'échange en peer to peer

1.3. Les architectures client-serveur.

Les architectures client-serveur sont au moins à 2 niveaux : le client et le serveur. Mais plus de niveaux peuvent être mis en œuvre.

L'architecture à deux niveaux (aussi appelée *architecture 2-tier*, *tier* signifiant *rangée* en anglais) caractérise les systèmes clients/serveurs pour lesquels le client demande une ressource et le serveur la lui fournit directement, en utilisant ses propres ressources. Cela signifie que le serveur ne fait pas appel à une autre application afin de fournir une partie du service.





Dans l'architecture à trois niveaux (appelée *architecture 3-tier*), il existe un niveau intermédiaire, c'est-à-dire que l'on a généralement une architecture partagée entre :

1. Un client, c'est-à-dire l'ordinateur demandeur de ressources, équipée d'une interface utilisateur (généralement un navigateur web) chargée de la présentation ;
2. Le serveur d'application (par exemple un serveur web), chargé de fournir la ressource mais faisant appel à un autre serveur, par exemple un serveur de base de données.
3. Le serveur de données, fournissant au serveur d'application les données dont il a besoin.

2. Le filtrage des ports.

2.1. Le filtrage des ports d'écoute.

Le filtrage des ports se fera à l'aide d'un service réseau nommé Firewall. Ce service est monté sur un routeur, permettant d'isoler le réseau local d'internet, et d'appliquer les règles de filtrage.

Ainsi, le firewall aura pour mission de vérifier les ports de toutes les demandes de connexion, de valider leur autorisation (en fonction d'une liste établie) et de laisser passer ou détruire la trame selon le cas.

Généralement, la règle sécuritaire de niveau maximal est appliquée : Tout ce qui n'est pas explicitement autorisé est interdit. Il faut alors ouvrir uniquement les ports nécessaires.

2.2. Le filtrage applicatif.

Le filtrage applicatif, effectué par un service proxy, permet de filtrer les communications application par application, ce qui signifie qu'il travaille au niveau de la couche 4 du modèle DOD. Le filtrage applicatif suppose donc une connaissance de l'application, et notamment de la manière de laquelle elle structure les données échangées.

Les services proxy sont plusieurs fonctionnalités :
- *La fonction de cache*
- *Le filtrage*
- *L'authentification*

La fonction de cache permet de délivrer au réseau local des données directement, conservées en mémoire, plutôt que d'aller les chercher à chaque fois sur Internet. Cela fonctionne avec les pages web, mais également avec les paquets téléchargés vi `apt-get install` par exemple, et avec tout ce qui passe par le routeur.

La fonction de filtrage permet de filtre les sites web autorisés, en fonction d'une liste blanche de sites autorisés , ou d'une liste noire de sites interdits, éventuellement complétée d'une liste de mots clés interdits.

La fonction d'authentification permet de connaître l'identité des utilisateurs d'internet, e donc de savoir ce qu'ils y font. Ceci étant à la limite de ce qui est autorisé par la loi informatique et libertés du 6 janvier 1978…
Il sera par contre possible, en toute légalité, de contraindre les utilisateurs dans des plages horaires.

2.3. Héberger soi-même son verseur.

Si vous envisagez d'héberger vous-même un serveur, il faut respecter quelques règles :
- Avoir conscience qu'une connexion ADSL n'est pas une bonne solution, car la banque passante montante est faible, et c'est pourtant celle qui est mobilisée pour répondre aux demandes des clients.
- mettre en place un firewall qui sécurise au mieux votre réseau. La bonne nouvelle est que toutes les box intègrent un firewall, qui d'ailleurs bloque toute connexion entrante par défaut.
- ouvrir un port sur votre firewall et le rediriger vers le serveur concerné.
- avoir une adresse stable : soit une adresse IP publique fixe soit un nom de domaine dynamique, c'est-à-dire qui met à jour son serveur DNS en cas de changement d'adresse IP publique (noip, …), configurée dans la box pour qu'elle soit toujours à jour.

2.4. La fragmentation du réseau en zones de sensibilités.

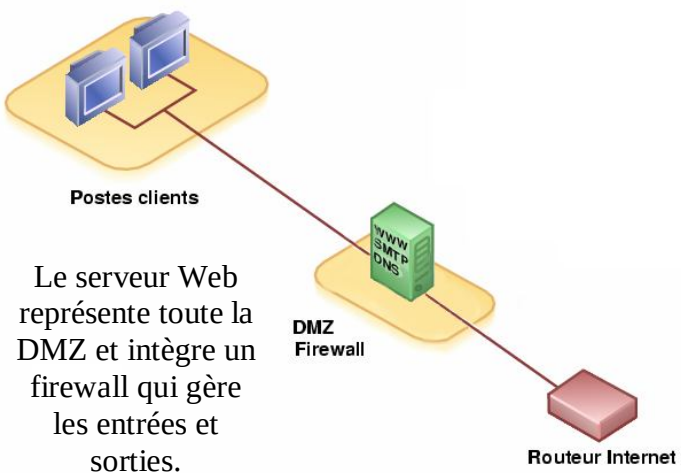
L'utilisation judicieuse d'un ou plusieurs firewall peut permettre la création de zones dites démilitarisées.

Une zone démilitarisée ou DMZ pour DeMilitared Zone est un réseau indépendant sur LAN de l'entreprise. C'est pour cela que ce réseau est également nommé réseau externe.

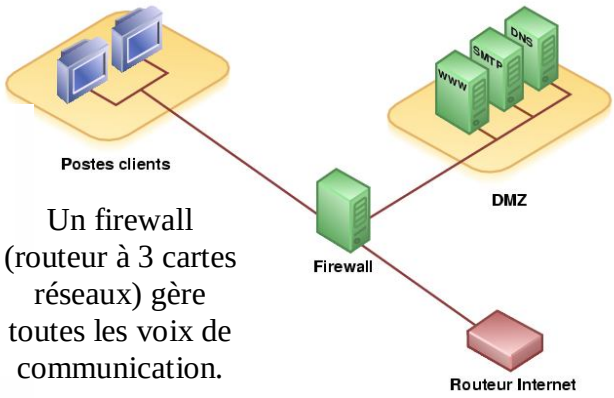
Les serveurs sensibles et les clients sont placés dans le réseau local, hautement sécurisé, et les serveurs publiques dans la DMZ.

Le concept de base est simple : toutes les connexions entrantes (depuis le web) sont dirigées vers la DMZ. Ainsi, aucune connexion ne peut atteindre le LAN.
Cette solution est théoriquement ultra sécurisante, mais cela suppose que les serveurs de la DMZ ne deviennent pas des relais pour les hackers. Si l'un de ces serveurs a une porte d'entrée vers le LAN, un hacker, qui prend la main sur ce serveur, accède au LAN. Il est donc préférable que la séparation DMZ – LAN soit étanche.
Il est toutefois possible d'ouvrir des sockets (un port réservé à une adresse IP), pour accéder à un serveur SGBDR du LAN depuis la DMZ, ce qui minimise quand même les risques.

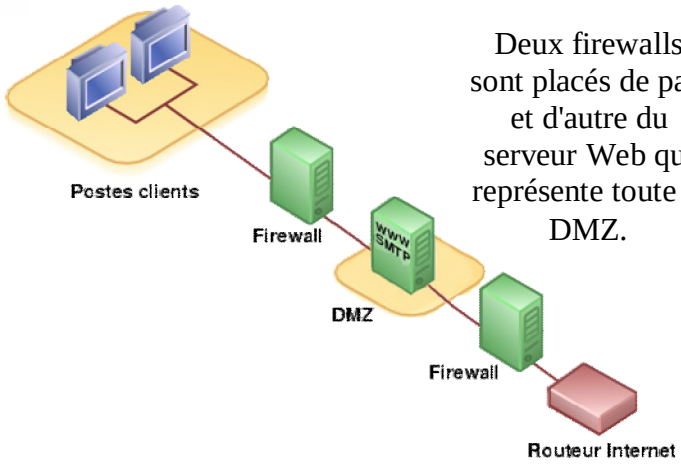
Les différentes formes de DMZ.



Le serveur Web représente toute la DMZ et intègre un firewall qui gère les entrées et sorties.



Un firewall (routeur à 3 cartes réseaux) gère toutes les voix de communication.



Deux firewalls sont placés de part et d'autre du serveur Web qui représente toute la DMZ.