TIW3 - Administration des systèmes et des bases de données

Un peu de réseau

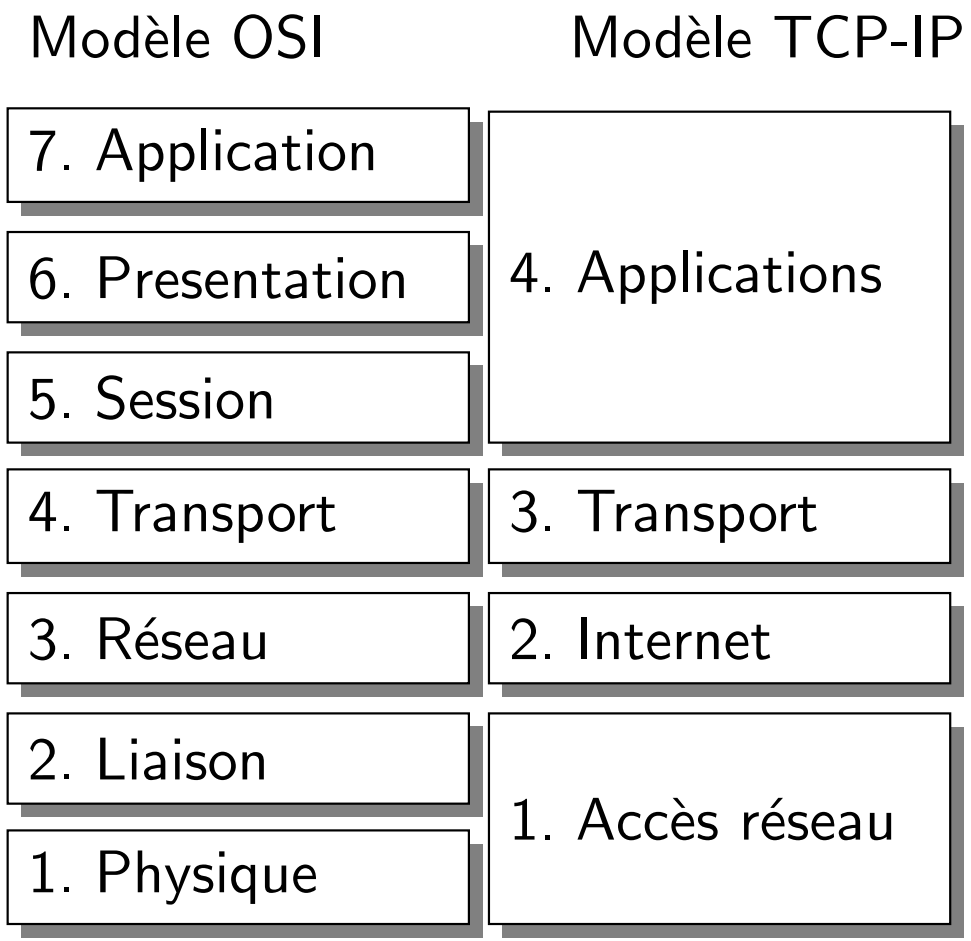
Fabien Rico (fabien.rico@univ-lyon1.fr)

Univ. Claude Bernard Lyon 1

séance 1

Les 7 couches du modèle OSI

- Norme dévellopée en parallèle de TCP/IP et internet.
- Présentation abstraite qui n'a pas forcement été suivie dans la pratique.
- Mais le réseau fonctionne bien par couche
- Pour débugguer il est nécessaire de tester les couches (dans l'ordre croissant).
- Pour mettre en place un service, il est nécessaire de savoir sur quel couche agir (et qui a le droit de le faire).



Exemple

- Vous avez mis en place 3 machines virtuelles,
  - la première propose une base de données, et un docker nginx qui réparti l'adresse `http://lavm/service/` sur les 2 autres avec équilibrage de charge.
  - les 2 VM proposent un service identique via TOMCAT qui utilise la base de données.

  Cela ne fonctionne pas, quel peut en être la cause 

  Comment tester 

Mettre en place un environnement de travail

- Comment passer un pare-feu ?
- Comment travailler à distance de manière sécurisée ?
- Comment le faire de manière simple et efficace ?

Les pare-feux agissent habituellement au niveau 4, il est possible :

- de mettre en place des tunnels,
- d'utiliser des serveurs mandataires (*proxy*),
- d'exécuter des applications à distance,
- de mettre en place des réseaux privés virtuels (VPN).

Ces solutions ont différents niveaux de complexité à la mise en place ou à l'utilisation, une efficacité plus ou moins grande.

Contraintes

- L'utilisateur doit avoir un accès autorisé à l'ordinateur
  - gestion de l'utilisateur comme un utilisateur local (généralement sous *nix) ;
  - gestion séparée.
- Environnement hétérogène (Par ex : un étudiant se loggue depuis son portable windows sur le serveur bsd de l'université).
  - problème d'encodage ;
  - problème d'affichage ;
  - problème d'accès aux périphériques ;
  - émulation de terminal ou de bureau.
- Sécurité
  - Que se passe-t'il si un utilisateur distant a accès aux touches du clavier ?
  - Que peut faire un utilisateur une fois loggué sur le vieux serveur de test ?

Exemple connexion en mode texte

- Telnet (1983) : un des premiers standards de l'internet
- Rlogin/Rsh (1991) :
  - permet des accès sans mot de passe,
  - exécution de commandes à distance
- ssh :
  - les communications sont chiffrées ;
  - authentifications par mot de passe ou clefs ;
  - finalisé depuis 2006.

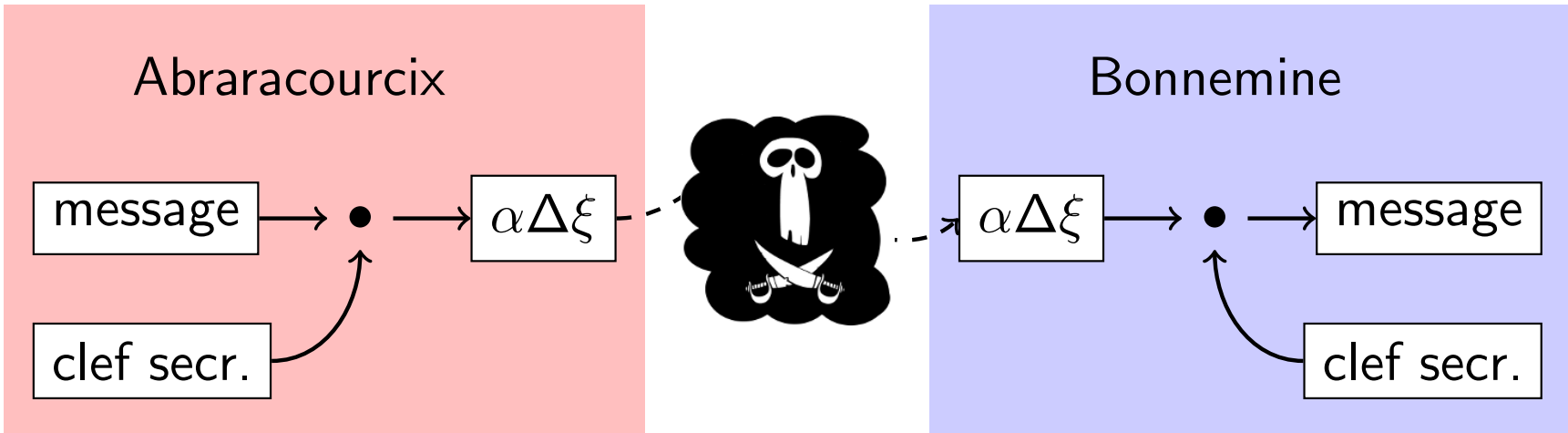
Méthode d'authentification

- Faire confiance à la machine du client :
  - problème si la machine cliente est compromise ;
  - problème si la machine cliente est mal authentifiée (adresse IP).
- Demander le loggin et mot de passe :
  - le mot de passe circule ;
  - la base de mots de passe doit être protégée ;
  - problème si le serveur est compromis.
- Utiliser un ≪ élément d'authentification ≫ *jeton* ou *token* (système à clef public/privée) :
  - la clef secrète est conservée sur le client ;
  - permet les connexions automatiques ;
  - le serveur doit accepter la clef.
- Utiliser une autorité de confiance :
  - le client est redirigé vers l'autorité de confiance ;
  - qui génère un jeton et le *signe* ;
  - le serveur vérifie le jeton et la signature.

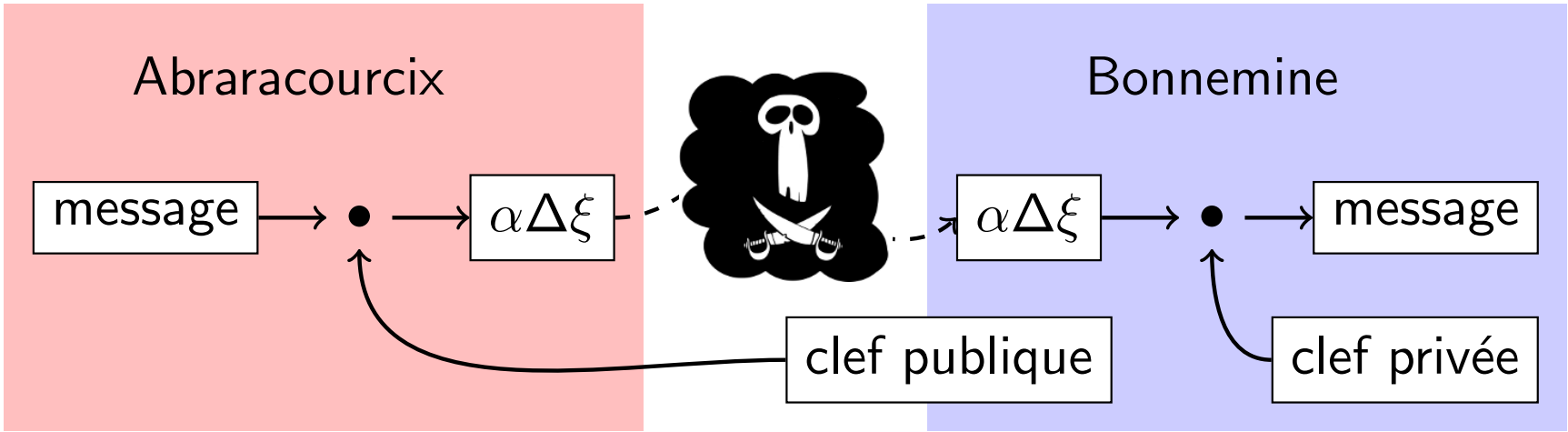
Utilisation de clef

Il y a 2 types d'algorithmes pour chiffrer les échanges entre 2 entitées :

- Système à clef partagée (DES/AES) :
  - + nécessite peu de ressources ;
  - - oblige à partager une clef secrète ;
  - ▸ Longueur des clefs sûres : 128 (recomendation ANSSI).



- Système à clef assymétrique (RSA) :
  - \- nécessite plus de ressources (clef plus grandes) ;
  - \+ pas de partage de clef secrète ;
  - permet l'authentification ;
  - permet la signature.



Fonctionnement des clefs asymétriques

La clef publique est un ensemble de valeurs telles que :

- Elle peut chiffrer un message que seule la clef privée peut déchiffrer
- Elle peut déchiffrer un message s'il a été chiffré par la clef privée.


Comment peut-on s'authentifier



Comment peut-on signer un message


ssh

ssh

ssh est un système de connexion à distance :

- Les communications sont chiffrées (clef symétriques) ;
- Le serveur s'authentifie (clef asymétrique) ;
- plusieurs authentifications sont possibles ;
- permet le transfert de port et l'affichage déporté.

- Openssh implémentation libre de ssh sur système unix
- SSHWindows, Pulse 2 Secure Agent implémentation d'un serveur ssh sous windows grâce à cygwin.
- Sous windows PuTTY client ssh, WinSCP client sftp.
- Client avec une version limité MobaXterm

Utilisation de ssh

- Connexions à distance (style rlogin)
  - ssh -l user hostname
  - ssh user@hostname
- Exécution de commande à distance (style rsh)
  - ssh -l user hostname cmd
  - ssh user@hostname cmd
- Copie de fichiers à distance (style rcp)
  - scp file1 file2 user@hostname:/home/plop/Bureau/
  - scp -r dir user@hostname:/tmp
- Utilisation du protocole sftp c'est à dire copie de fichiers via ssh.
  - Sous windows filezilla ou winscp ;
  - Sous linux sftp est un protocole possible :
    konqueror sftp://fabien.rico@linuxetu.univ-lyon1.fr/tmp/
- Partage de fichiers via ssh et Fuse : *sshfs*/
  - sshfs fabien.rico@linuxetu.univ-lyon1.fr: plop/
  - fusermount -u plop

Authentification pour ssh

Qqs méthodes

- Authentification par la machine du client (généralement désactivée).
- Authentification par mot de passe.
- Authentification par clef.

Pour utiliser l'authentification par clef, il faut :

- Générer une paire de clefs privées/publiques sur le client :
  `ssh-keygen -t rsa`
- Placer la clef publique sur le serveur dans `~/.ssh/authorized_keys`
- *Si la clef a un mot de passe* :
  - lancer un agent d'authentification : `ssh-agent bash`
  - ajouter la clef privée à l'agent : `ssh-add`
- essayer : `ssh toto@serveur`

Si la clef n'a pas de mot de passe, il ne faut pas d'agent. De plus, sous gnome ou kde, un agent est déjà lancé et demande le mot de passe de la clef automatiquement en cas de besoin.

Connexion en mode graphique

Pourquoi ?

- Démonstration à distance.
- Logiciels graphiques (gestion, bureautique).
- Logiciels nécessitant d'être sur un réseau local.
- Version simplifiée d'un VPN.

- Hétérogénéité : toutes les applications doivent fonctionner
  - Si les systèmes le permettent, seule la communication avec la partie graphique est concernée (serveur X + ssh).
  - Sinon, utilisation d'une fenêtre locale qui recrée l'environnement graphique distant (Term. Serv., VNC et NX).
- Coût :
  - Transport d'images via le réseau.
  - Contrainte sur la latence.
  - Coût pour le serveur.

Serveur X

Utilisation sous forme de terminal :

- Toutes les applications tournent sur un serveur.
- L'affichage est déporté sur un terminal.
- $\Rightarrow$ Simplification de l'administration.
- $\Rightarrow$ Réduction du coût ?

C'est une utilisation ancienne des serveurs unix qui marche toujours !

Serveur X

Utilisation sous forme de terminal :

- Toutes les applications tournent sur un serveur.
- L'affichage est déporté sur un terminal.
- $\Rightarrow$ Simplification de l'administration.
- $\Rightarrow$ Réduction du coût ?

C'est une utilisation ancienne des serveurs unix qui marche toujours !

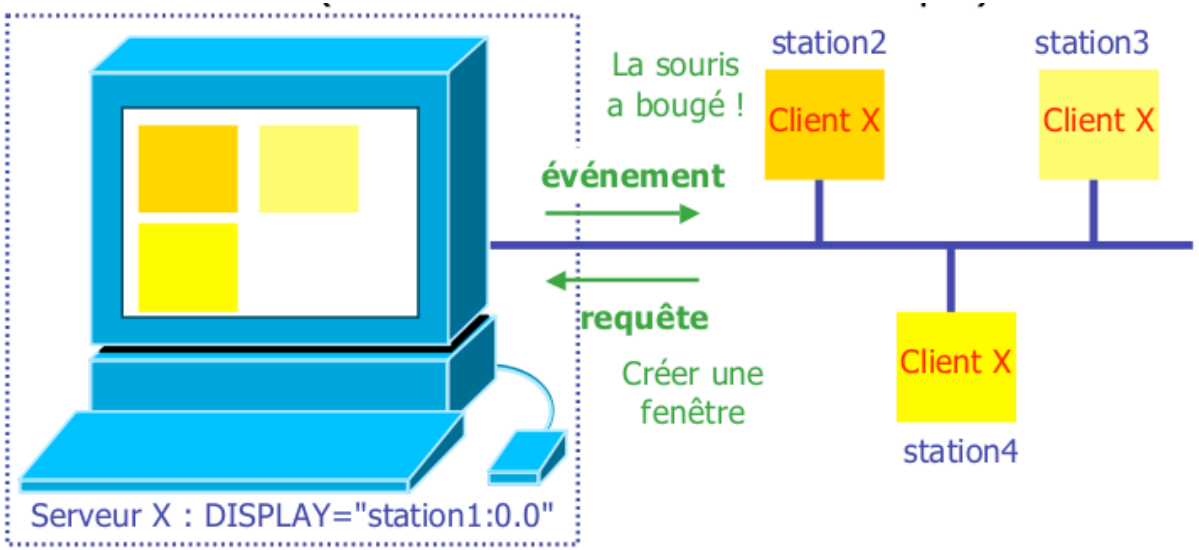
Si

- on configure le part feu
- on enlève les sécurités par défaut de X.

Utilisation du serveur X

Fonctionnement

Sous *nix, la variable d'environnement DISPLAY donne le serveur graphique à utiliser.



Utilisation du serveur X

Exemple (Utilisation de DISPLAY)

- Sous bash

```
$ env | grep DIS
DISPLAY=localhost:1.0
```

Signifie que le serveur graphique est sur le port 6001 de la machine locale.

-

```
$ export DISPLAY=192.168.56.1:0
$ xterm
```

transfert l'affichage sur le port 6000 de la machine 192.168.56.1.

 *Les données sont échangées en clair*

Utilisation de ssh

Fonctionnement

ssh utilise les fonctionnalités réseaux du serveur X mais protège les données dans un tunnel.
Pour cela il faut utiliser les options $-$X ou $-$Y de ssh.

 *L'accès au serveur X d'une machine par une autre est un trou de sécurité car il permet (entre autre) d'installer un keylogger.*

- Sur certaines distributions (ni débian, ni ubuntu), l'option $-$X est sécurisée mais moins fiable.
- L'option $-$Y supprime la sécurité (mais c'est dangereux).

Utilisation de ssh

Exemple (Utilisation avec ssh)

ssh −X machineloin permet un login sur le serveur « machineloin » en déportant l'affichage.

- ```
  $ ssh -X 192.168.56.1
  rico@192.168.56.1 s password:
  Last login: Mon Oct 18 15:03:21 2010 from 192.168.56.2
  [rico@horus ~]$ env | grep DISPLAY
  DISPLAY=localhost:10.0
  $ emacs &
  ```

  Cela fonctionne car le port 6010 du serveur est envoyé sur le port 6000 de la machine cliente.
  ```

Fenêtre incluse

Si le gestionnaire graphique local ou distant ne permet pas l'utilisation du réseau.

- Utilisation de logiciel recréant le bureau dans une fenêtre locale.
- La totalité de l'image du bureau doit passer sur le réseau.
- Possibilité de compresser le flux.
- Protocols RDP (Windows) VNC (tout).

Logiciels :

- « Connexion Bureau à distance » par défaut sous windows.
- rdesktop client RDP sous linux.
- NX/FreeNX (encapsule plusieurs protocoles).
- ...

Protocol RDP

RDP

Cela correspond au *serveur de terminal* sous windows

- Capable de mapper des ressources locales :
  - imprimante,
  - disque,
  - sond.
- Souvent limité :
  - à la prise en main à distance sur les versions grand public ;
  - à un seul utilisateur pour les versions entreprises.

- Client windows
  Menu Démarer→application→Bureau à distance
- Client linux

```
rdesktop -k fr -g 1280x1024 -r sound:local
         -r printer:"ImpLabo"="HP LaserJet 3050 PCL5"
         -r disk:home="$HOME"
         -r sound:local:alsa 192.168.56.2
```

VNC

- Partage de bureau
  - utile pour démonstration et assistance ;
  - pas vraiment un système de connexions.
- Disponible sur de nombreux systèmes d'exploitation.
  - windows : RealVNC, ...
  - Linux : vino, kfrb, ...
  - MAC OS : Apple Remote Desktop
- Partage d'écran par le protocole RFB (Remote Frame Buffer)
  - assez gourmand en bande passante mais peut être compressé ;
  - les versions commerciales ont des algorithmes plus efficaces.
- Problème de sécurité :
  - pas de vrai authentification ;
  - pas de chiffrage des échanges ;
  - possibilité de l'utiliser à travers un tunel ssh ou un vpn.

Adresse IP

Sur internet, toutes les machines ont une adresse IP :

- Une adresse ipv4 est une valeur de 32 bits, qu'on représente souvent via ses 4 octets en décimal séparés par des points. Par exemple 134.214.102.5
- Certaines adresses ne sont utilisées que localement (*non routable* ou *privées*)
  - 192.168.x.y
  - 172.16.x.y à 172.32.x.y
  - 10.x.y.z

  D'autres sont utilisées pour des cas particuliers (diffusion, ...)
- En dehors de ces cas, un seul appareil peut avoir une adresse (problème vu le nombre d'objets connectés).
- Pour trouver une solution à la pénurie d'adresse, définition de ipv6 (4×32 octets).

Adresses privées

Les adresses privées sont très utilisées :

- pour palier la pénurie d'adresses ipv4 (via le NAT) ;
- pour gérer les connexions internes de docker/VM/point à point ;
- pour créer des services qui restent confinés.

Par exemple, à l'université, ces adresses sont utilisées :

- pour attribuer des adresses wifi ;
- pour certains services internes (DNS, vpn, ...) ;
- pour les machines du serveur openstack (d'où l'utilité du proxy).

Que se passe-t-il si un docker utilise l'adresse interne 172.18.0.1 et qu'il doit contacter un serveur DNS de l'université à l'adresse 172.18.0.1 ?

Routage

Les adresses IP servent essentiellement à contacter les machines via un routage de proche en proche.

- Lorsqu'un appareil doit contacter une adresse, il compare cette valeur à une table de routage.
- La comparaison se fait en utilisant un certain nombre de bits au début de l'adresse.
- La table lui donne l'interface par laquelle envoyer le paquet.
  - si la destination est directement connectée, le paquet est envoyé (passage au niveau 2 OSI) ;
  - sinon, le paquet est transmis à un routeur ou passerelle (qui est forcement directement connecté).

Routage exemple

Par exemple, mon portable chez moi à 3 lignes dans la table de routage :

- le réseau wifi local 192.168.3.0/24 (interface wlp1s0)
- le réseau virtuel des VM hébergées 172.16.0.0/16 (interface virbr0)
- le réseau par défaut 0.0.0.0/0 (passerelle 192.168.0.1, interface wlp1s0)

Les premiers bits de chaque adresse sont comparés à ceux de chaque ligne et celle ayant la plus grande correspondance est choisie.

- 192.168.3.4 est envoyé directement par l'interface wlp1s0 ;
- 134.214.101.2 est envoyé par l'interface wlp1s0 à la passerelle 192.168.0.1.

Connexion réseau

- Un dialogue sur le réseau implique forcement 2 processus sur 2 appareils.
  - 2 adresses IPs servent à identifier les appararеils ;
  - 2 ports servent à identifier les processus.
- Le quadruplet IPsource, port source, IPdestination port destination est unique sur internet et permet à un processus d'envoyer quelque chose à un autre processus.
- Si on inverse la source et la destination, cela concerne les mêmes processus mais pour le sens inverse.
- Il y a 2 modes :
  - connecté (TCP), le quadruplet est utilisé pendant un certain temps ;
  - déconnecté (UDP), le quaruplet n'est utilisé que pour un allez retour.
- Les 2 modes sont au niveau 4 du modèle OSI, pour le moment rien ne dit ce qui est utilisé.

Client/Serveur

Au niveau 4, c'est toujours un client qui contacte un serveur et obtient une réponse.

- Ce qui identifie le serveur est l'adresse et un port connu de tous ou trouvé via d'autres services (google, qwant).
- Ce qui identifie le client est l'adresse de la machine et un port (en général tiré au hasard).
- Beaucoup de services ont des ports par défaut connus :
  - http : 80, https : 443 ;
  - DNS : 53.
- Mais il n'y a aucune obligation à les utiliser.
- La seule restriction est que la plupart des systèmes d'exploitation réservent les ports inférieurs à 1024 à l'administrateur.

Pare feux

Pour des raisons de sécurité, il est necessaire de filtrer certains paquets

- pour empêcher l'installation de serveur non officiel ;
- pour empêcher la sortie d'information ;
- pour empêcher certaines attaques (DoS) ;
- pour limiter certains virus ;
- ...

Pour cela, certains routeurs et les machines peuvent appliquer des filtres :

- en fonction des sources, destinations, ports ;
- avec une certaine mémoire (ouverture du filtre pour la réponse à un paquet sortant) ;
- en fonction de certains protocoles connus (ouverture du port pour la connexion data de FTP) ;
- avec des analyses de plus haut niveau (rare).

Utiliser un service malgrès un firewal revient la plupart du temps à détourner un flux c'est à dire à utiliser un autre port et une autre adresse.

Traduction d'adresse

Certains paquets ne peuvent pas être routés directement.

- à cause d'une adresse non routable ;
- parce que le chemin qu'il doit prendre n'est pas officiel (VPN).

Pour que cela fonctionne, le routeur doit changer l'adresse IP utilisée, c'est la traduction d'adresse (NAT).

- Soit le routeur remplace simplement l'adresse non utilisable du client par la sienne pour les paquets allé et fait le remplacement inverse au retour. C'est le NAT.
- Soit le routeur remplace aussi le numéro de port source par un autre non utilisé, c'est le PAT. Cela permet de donner accès à plusieurs machines.

Cas particulier de HTTP

Par définition, HTTP est un protocole applicatif (niveau 7 du modèle OSI). Mais son rôle a beaucoup évolué.

- Beaucoup de protocoles se sont développés au dessus de HTTP (DAV, websocket, REST, ...)
- Les proxy inverses permettent de contacter des services différents à partir de la même connexion HTTP. Même fonctionnement que le port dans TCP.
- Les proxys HTTP (inverse ou non) permettent aussi de contacter des services web en remplaçant l'adresse du vrai client, comme le NAT.

HTTP change de rôle pour celui d'un protocole bas niveau permettant de transporter les données des protocoles de niveau supérieur.
De nos jours, les services web proposent toute sorte de solutions :

- connexion à distance ;
- copie de fichiers ou partage ;
- visio conférence ;
- ...

Redirection de port

Un routeur peut changer la destination de paquet :

- Pour détourner le flux vers un serveur interne
  - pour les mapping de port de docker ;
  - pour les machines virtuelles ;
  - par les *BOX pour rendre certains ordinateur utilisable de l'extérieur (jeux, administration).
- Pour capturer le flux
  - portail captif d'authentification ;
  - controle parental.
- Sous linux c'est fait via les iptables dans la table nat.

Tunnel ssh (port forwarding)

Il est aussi possible de créer des programmes pour écouter sur un port et transmettre tout le flux à un autre endroit. Il est même possible de déplacer les données entre 2 machines avant la retransmission. C'est un *tunnel*.

Principe

ssh est capable de se mettre en écoute sur un port de la machine locale ou de la machine distante.

- toute communication sur le port en écoute est transférée à l'autre bout du tunel ssh
- puis retransmise à une nouvelle machine sur un autre port.

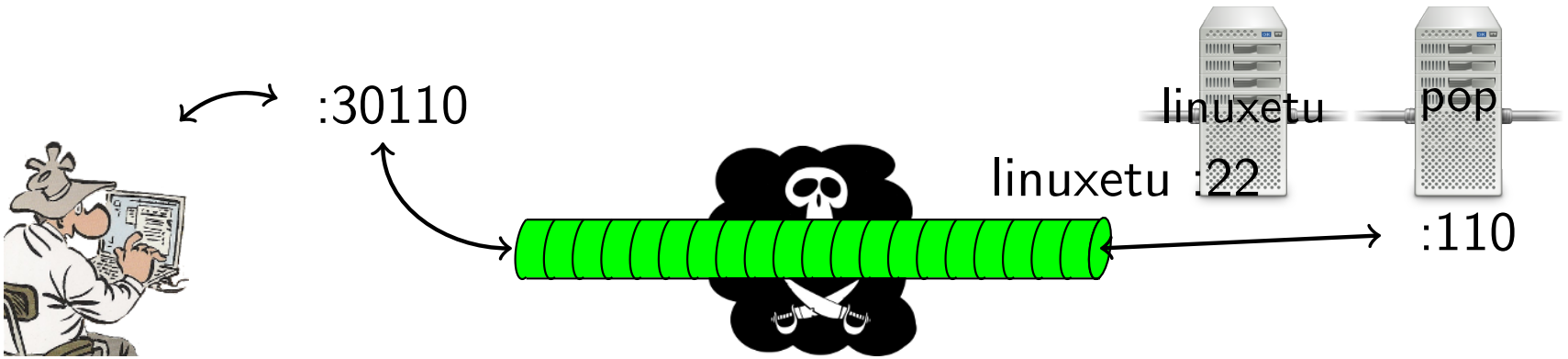
La communication à travers le tunnel est effectuée par ssh donc chiffrée.

Pour faire un tunnel, il faut utiliser les options :

- −L (local → distant)
- −R (distant → local).

```
ssh -L8080:serveurInterne:80 serveurExterne.univ-lyon1.fr
```

Exemple



Exemple (Lecture de mail)

Par exemple :

- telnet pop.univ−lyon1.fr 110 ⟶ lecture du mail non sécurisé !
- On peut faire passer n'importe quoi dans un tunnel ssh :
  ssh −L 30110:pop.univ−lyon1.fr:110 linuxetu .univ−lyon1.fr
  telnet localhost 30110
- Tout ce qu'on envoie sur localhost/30110 arrive sur pop/110 mais les données sont chiffrées entre la machine locale et le serveur ssh (encapsulées dans le tunel)

Limitations

Cela fonctionne uniquement si les connexions n'utilisent qu'un port connu à l'avance

Cela pose des problèmes similaires à ceux des firewall.

- les connexions de retour (ex : ftp) ;
- l'utilisation de port aléatoire (ex : jeux en réseaux).

Proxy dynamique, proxy socks

Une autre option de ssh permet de mettre en place un proxy socks :

- option −D <num de port d'entree> ;
- proxy dynamique qui permet de transférer un flux vers n'importe quel serveur ;
- necessite que le programme client sache utiliser le protocole socks (4 ou 5).

Pour utiliser le proxyfieur avec n'importe quel programme, il faut utiliser un proxifieur. C'est un programme qui « proxyfie » les demandes de connexion d'un autre programme.

- proxychains sous linux/mac ;
- proxifier , ProxyCap, CapSocks64 sous windows.

Exemple de proxyfieur

Exemple (Configuration de proxychains)

/etc/proxychains.conf

```
[ProxyList]
# entrée du tunnel ssh
socks5  127.0.0.1 2222
```

Exemple (utilisation du tunnel)

- Mise en place du tunnel :

  ssh −D 2222 p13434344@linuxetu.univ−lyon1.fr

- Lancement d'une commande :

```
$ proxychains rdesktop -k fr -g 90% tsedoua.univ-lyon1.fr
[proxychains] config file found: /etc/proxychains.conf
[proxychains] preloading /usr/lib64/proxychains-ng/libproxychains4.so
[proxychains] DLL init: proxychains-ng 4.10
[proxychains] Strict chain  ...  127.0.0.1:2222  ...  tsedoua.univ-lyon1.fr:3389  ...  OK
Connection established using SSL.
...
```

Mini VPN

Pour aller plus loin dans le domaine de transfert de port, on peut créer une interface réseau virtuelle de chaque coté de la connexion ssh. Les 2 interfaces sont reliées via un tunnel chiffré et permettent de mettre en place un VPN entre 2 réseaux (voir option -w de ssh).

- Il faut être administrateur sur le client et le serveur ssh pour avoir les droits de créer les interfaces.
- Il faut autoriser le routage sur les machines. (voir /proc/sys/net/ipv4/ip_forward).
- Il faut mettre en place *à la main* les règles de routage. (voir commande route)

OpenVPN

OpenVPN est un utilitaire qui permet de faire un VPN sous la pluspart des systèmes d'exploitation.

- Peut utiliser TCP ou UDP pour transporter les paquets.
- Propose 2 modes :
  - TAP : openvpn partage l'accès à sa carte via un bridge. Le Client a alors une pseudo interface dans le réseau distant.
  - TUN : openvpn route les paquets dans un tunnel. Il doit utiliser le NAT pour permettre aux clients de dialoguer avec d'autres machines du résau distant.
- Il permet des connexions entre machines pour faire un VPN entre sites.
- Il permet des connexions avec un utilisateurs pour permettre un accès à des travailleurs externes.