Démarrage du Système

Présentation de Linux

Outils de cryptage

Niveau d'exécution (RunLevels) de linux

Détails sur les scripts dépendant du niveau d'exécution (RunLevel) :

- Init exécute le script « /etc/rc.d/rc » lorsqu'il entre/quitte un runlevel (Mandrake)
- Les scripts sont stockés dans « /etc/init.d/ »
- Ils doivent supporter les paramètres: start/stop/status/restart
- Le système crée des liens dans « /etc/rc.d/rcX.d » vers les scripts de « /etc/init.d »
- Ces scripts gèrent (arrêt/démarrage) TOUS les services

Niveau d'exécution (RunLevels) de linux

Détails sur les scripts dépendant du RunLevel sous Mandrake (méthode System V) :

- Les liens commençant par '**S**' (**start**) sont exécutés lors de l'entrée d'un runlevel
  - Les noms des liens commencent par '**S**' suivi d'un numéro de priorité suivi du nom du script
  - Les scripts de même priorités sont exécutés dans l'ordre alphabétique
    - ✓ Ex: S03iptables, S03shorewall, S10network, S11portmap, ...
- Les liens commençant par '**K**' (**kill**) sont exécutés lors de la sortie d'un runlevel
  - Les noms des liens commencent par '**K**' suivi d'un numéro de priorité suivi du nom du script
  - Les scripts de même priorités sont exécutés dans l'ordre alphabétique
    - ✓ Ex: K90network, K89portmap, ...

Outils de gestion d'espace: df

- Espace utilisé et libre sur les partitions
  - Commande: **`df [options] [FILE]`**
  - **`[FILE]`** peut être un fichier, un périphérique, un point de montage
  - Une option intéressante « -h » pour une lecture plus aisée
  - Exemples

```
[legond@hebe]> df /dev/hda5 /dev/sda1
Filesystem           1K-blocks      Used Available Use% Mounted on
/dev/hda5             38909396  32879472   6029924  85% /
/dev/sda1             27550256    267588  27282668   1% /data
```

```
[legond@hebe]> df  -h /dev/hda5 /dev/sda1
Filesystem            Size  Used Avail Use% Mounted on
/dev/hda5              38G   32G  5.8G  85% /
```

Outils de gestion d'espace: du/ls

- Occupation **<u>réelle</u>** d'un fichier, d'un répertoire ou d'un périphérique
- Commande: ***du***
  - Options utiles : -h (humain) -s (total/argument) -c (grand total)
  - Différent de ***ls -l*** ! (Fichiers creux / Sparse files)
  - Exemple de création d'un fichier creux
    - ✓**dd if=if/dev/zero of=/tmp/Essai seek=1000000 bs=1 count=1**
    - ✓Ou le programme C

Outils de gestion d'espace: du/ls

```
[legond@morphee]>ls -l /tmp/essai
---x-----x  1 bonnaire src 1000001 Oct 18 14:25 /tmp/essai
```

Taille apparente du fichier en octets

```
[legond@morphee]>du -B 4096 /tmp/essai
1       /tmp/essai

[legond@morphee]>du -h /tmp/essai
4,0K    /tmp/essai
```

Taille réelle: 4 Ko soit 1 bloc (Reiserfs block size = 4096 octets)

- L'option "-s" permet de faire la somme des répertoires

```
[legond@morphee]>du -s -h /usr
3,1G     /usr

[morphee 14:40]>du -s -h /usr/*
176M    /usr/bin
36M     /usr/include
690M    /usr/lib
...
```

Encore des outils de gestion de disque

- **dd** (disk dump): copie de données. Très souvent utilisé !
- **tar**, **un/zip**, **cpio**: création d'archives

```
tar cvBf - directory | (cd /backupdir; tar xpBf -)
```

- **gzip**, **bzip2**: compression de fichier
- **tee**: duplication d'un flux de données
- **mc**: midnight commander (manipulation de fichiers)
- **rsync**: copie/synchronisation de répertoire
- **dump**: sauvegarde de fs de type ext2/ext3
- **losetup**: permet de manipuler des images de disque (fichier .iso)
- **chown**, **chmod** : changer les propriétaires, les attributs des fichiers
- **touch, stats** : manipuler la date des fichiers
- **lsof**: liste des fichiers ouverts par un processus (UTILE!!)

Ajouter et modifier les utilisateurs

- Changer de mot de passe:
  - ✓ `passwd [-k] [-l] [-u [-f]] [-d] [-S] [nom_utilisateur]`
- Changer de shell:
  - ✓ `chsh [-s shell] [-l] [-u] [-v] [utilisateur]`
- Changer l'âge d'un mot de passe
  - ✓ `chage`
  - ✓ Expiration du mot de passe (0 ou 99999 ➔ pas d'expiration)
- Commande d'ajout d'utilisateur: **useradd**
  - ✓
```
useradd  [-u uid [-o]] [-g group] [-G group,...]
[-d home] [-s shell] [-c comment] [-m [-k template]]
[-f inactive] [-e expire ] [-p passwd] name
```
- Commande de modification d'utilisateur: **usermod**
  - ✓
```
usermod  [-u uid [-o]] [-g group] [-G group,...]
[-d home [-m]] [-s shell] [-c comment] [-l new_name][-f
inactive] [-e expire ] [-p passwd] [-L|-U] name
```

Les identités !

- Attention aux conflits entre les UID des utilisateurs locaux et externes.
  - Pose des problèmes de sécurité en NFS
  - Pose des problèmes lors de l'authentification
    - ✓Voir le fichier **/etc/nsswhich.conf**
- "id" permet d'obtenir des informations sur sa propre identité.

```
root@morphee> id
uid=0(root) gid=0(root)
groups=0(root),1(bin),2(daemon),3(sys),4(adm),6(disk),10(wheel)

root@framekit-dev> id -u
0

legond@framekit-dev> whoami
legond
```

- S'authentifier sous un autre utilisateur (changer son identité) :

```
legond@framekit-dev > su apache
Password:
```

Linux : un système multi-utilisateurs !

- “getent” permet d’obtenir des informations sur un utilisateur ou un groupe.

```
legond@hebe > getent passwd apache
apache:x:74:74:system user for apache2:/var/www:/bin/sh

legond@hebe > getent group src
src:*:300:busca,darche,jlm,cdu,bf,cg,vevar,root
```

- “testsaslauthd” permet de vérifier l’authentification sur un service (login, ftp, …)

```
legond@hebe > testsaslauthd -u legond -s su -p monpass
0: OK "Success."
```

Les fichiers « shadow »

- Le mots de passe sont stockés sur disque
  - Fonction a sens unique
  - Contient un ensemble d'infos sur les utilisateurs
- PB: Tout le monde peut lire /etc/passwd.
  - Problème : les clefs des mdp sont stockés en clair
  - **Dangereux !!! ➔ brute force attack**
- Les clefs des mdp ne sont plus stockés dans « /etc/passwd ».
- Elles sont stockés dans « /etc/shadow » que seul root peut lire.
- Pour convertir les fichiers « /etc/passwd » classiques vers le format shadow : *unshadow* de john The Ripper

Choisir un mot de passe

- Certains mots de passes peuvent être devinés:
  - On crypte un ensemble de mdp préexistants
  - Dictionnaires anglais, français, …
  - Noms de films/acteurs/produits/personnes célèbres
  - Génération de mdp par des règles de transformations (Librairies de crack)
- **Un bon mdp ne doit pas :**
  - Utiliser le dictionnaire
  - Indépendant de l'environnement de l'utilisateur (login name, nom réel)
    - ✓ Pas de nom, surnom, d'identifiant (même modifier), de mot connu, marque
    - ✓ Pas de date (textuelle ou numérique), de clavier
- **Un bon mdp doit :**
  - Etre composé de ponctuations, chiffres, lettres majuscules et minuscules
  - Avoir au moins 7 caractères (si possible 10 à 14)
  - Vous pouvez vous trouver un algo (mais il doit rester secret)
  - Changer son mdp régulièrement

Choisir un mot de passe

- Récupérer les fichiers fichiers
  - /etc/shadow
  - /etc/passwd
- Afficher le fichier passwd et constater qu'il est illisible
- Problème de l'attaque par brute force
- Attaquer les mots de passe faibles
  - Un système est aussi faible que le plus faible de ses maillons
  - Utiliser la commande unshadow
  - Utiliser la commande /usr/sbin/john (de john the ripper)
- Exemple de bon mot de passe
  - Exemples: zqhYtds/965 moNpass,67$ Hg#91__bb
  - On peut en générer de façon « pseudo-aléatoire » : « pwgen » et « mkpasswd »
  - Test des mots de passe : « cracklib_test » (compiler)

Effacer un utilisateur

- Commande de modification d'utilisateur: **`userdel`**
  - Effacer un utilisateur et, éventuellement, tous ses fichiers
    - ✓**`userdel [-r] name`**
  - L'option "-r" efface tous les fichiers associés
    - ✓Le répertoire de travail (homedir) et sa boîte mail, …
    - ✓**ATTENTION: ne peut être annulé !**
- Vérifier l'intégrité de « /etc/passwd »: **`pwck`**
- Encore et toujours: RTFM !

Outils de gestion des droits

- Changer les droits :
  `chmod [options] [augo] [+-=] [rwxstugo] fichier(s)`
- Signification :
  - ✓ « **a** » : S'applique à tout le monde (user, group et other)
  - ✓ « **u** » : S'applique au propriétaire du fichier (user)
  - ✓ « **g** » : S'applique au groupe (group)
  - ✓ « **o** » : S'applique aux autres (other)
  - ✓ « + » : Ajout de nouveaux droits
  - ✓ « - » : Suppression d'anciens droits
  - ✓ « = » : Redéfinition complète des droits sans tenir compte des anciens
  - ✓ Fixer les droits par leurs valeurs octales pour u/g/o : r(4), w(2), x(1)

```
legond@scylla > ls -al unfichier
-rw-------  1 legond src 0 oct  1 22:59 unfichier
legond@scylla > chmod go+w unfichier
legond@scylla > ls -al unfichier
-rw--w--w-  1 legond src 0 oct  1 22:59 unfichier
legond@scylla > chmod 644 unfichier
legond@scylla > ls -al unfichier
-rw-r--r--  1 legond src 0 oct  1 22:59 unfichier
```

Outils de gestion des droits

- Définir les droits par défaut des fichiers créés:
  - ***umask [-p] [-S] [mode]***
  - Fixer les droits par leurs valeurs octales pour u/g/o : r(4), w(2), x(1)
  - Ajouter tous les droits que vous voulez interdire !

```
legond@scylla 12:09 > umask 000
legond@scylla 12:09 > umask -s
u=rwx,g=rwx,o=rwx
legond@scylla 12:11 > umask 222
legond@scylla 12:11 > umask -s
u=rx,g=rx,o=rx
```

- Changer le propriétaire / le groupe d'un fichier :
  - ***chown [options] propriétaire[:groupe] fichier(s)***
  - ***chgrp [options] groupe fichier...***

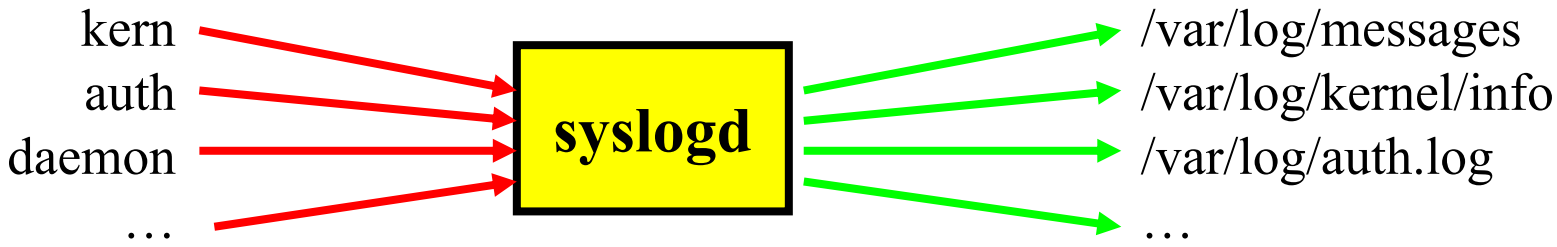
Gestion des traces (logs)

- Principes
  - Notifier toutes les actions du système dans un fichier qui est en perpétuel extension
  - Les journaux peuvent être gérés par les daemons/services « ***syslogd*** » ou « ***syslog-ng*** »
  - Type des journalisation possible: console, mail, fichier, machine
- Niveaux possibles de sévérité/criticité des erreurs (« level ») sont dans l'ordre :
  - **emerg(ency):** Erreurs diffusées à tout le monde. Le système ne sera plus en état de fonctionner
  - **alert:** Erreurs qui doivent être corrigées immédiatement.
  - **crit(ical):** Erreurs critiques à corriger le plus tôt possibles (erreurs de périphériques)
  - **err(or):** Erreurs de niveau « standard »
  - **warning:** Erreurs légères
  - **notice, info, debug:** Informations avec plus ou moins de détails ou d'importance
- Fonctions C pour « logger » des évènements: openlog(), syslog(message), closelog()
- Voir les derniers logs du noyau : « ***`dmesg`*** »
- Voir les dernières connexions : « ***`last`*** »

Les « facilities »

- Les évènements générés par une application/un service peuvent être envoyés à différentes destinations appelées usines de traces (« facilities »).
- Usines de traces (« facilities »):
  - user : messages des processus utilisateurs
  - kern : messages du noyau (et du firewall!)
  - daemon : messages des services
  - mail : messages du services de mail (pop/smtp)
  - news : messages du services de news
  - auth : messages d'authentification et de login
  - cron : messages des planificateurs de tâches
  - local0-7 : des facilities en libre services
  - lpr / uucp / mark (timestamp)

Le démon / service « syslog »



- C'est un commutateur (« dispatcher ») de lignes de traces
- Lit les données envoyées, par les applications, aux facilities
- Les données peuvent être envoyées à « syslog » localement ou par le réseau (UDP port 514, non crypté)
- Ecrit les traces dans des endroits désignés par l'administrateur.
- Capable de séparer les informations
- Configuration par le fichier « /etc/syslog.conf »

« /etc/syslog.conf »

- Lignes de configuration de la forme

  "facility_source_filter destination"

- ***Facility_source_filter*** : permet de sélectionner et d'extraire des données provenant des facilities.
  - Une forme simple est: « facility.level »
  - Le niveau (« level ») peut être crit, emerg, *, …
  - L'usine (« facility ») peut être auth, kern, *, …
  - * désigne tous les éléments ou tous les niveaux
  - Possibilité de séparer, d'agréger et/ou de dupliquer les logs
- ***Destination*** : vers quoi envoyer les données reçues par syslog
  - Un fichier simple. Ex: /var/log/messages
  - Un fichier particulier. Ex: /dev/tty5 (la 5^{e} console)
  - Vers le port UDP 514 d'une machine. Ex: @diane (envoie à la machine diane)

Fichiers de logs

- Exemple :

```
*.*;auth,authpriv.none                    -/var/log/syslog
kern.=debug;kern.=info;kern.=notice       -/var/log/kernel/info
kern.=warn                                -/var/log/kernel/warnings
kern.err                                  /var/log/kernel/errors
```

- Les Fichiers de logs importants sous linux sont :
  - /var/messages ➔ messages du systèmes
  - /var/syslog ➔ fichier de log important !
  - /var/auth.log ➔ tout ce qui est authentification
  - /var/boot.log ➔ log du démarrage du système
  - /var/kernel/* ➔ tous les messages du noyau
  - /var/daemons/* ➔ tous les messages des services
  - ….
- **ATTENTION : La structure du /var/log varie suivant les systèmes et les administrateurs !!!**
- Voir aussi acct ➔ lastcomm, ac, sa, accton, dump-acct, dump-utmp,

Outils de configuration réseaux

- **Scripts de démarrage / arrêt** du réseau :
  - Sur Mandrake : fichier « ***/etc/init.d/network*** »
  - Sur Gentoo : fichier « ***/etc/init.d/net.eth0*** »
  - Il configure automatiquement les routes
- La configuration de la carte se fait par « ***ifconfig*** » ou « ***ethtool*** »
- L'affichage et la manipulation de la table de routage IP se fait par la commande « ***route*** »
- L'ensemble du contrôle TCP/IP peut se faire par la commande « ***ip*** »

Diagnostiques réseaux : ping, netstat, telnet

- La commande « ***ping nom*** » permet de savoir si une machine est vivante. Une version parallèle nommée « *fping* » existe.
- La commande « ***netstat*** » permet de savoir l'ensemble des ports ouverts

```
root@scylla >  netstat -ln
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:32768           0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:994           0.0.0.0:*               LISTEN
......
```

- « ***netstat –r*** » affiche la table de routage
- « ***host nom*** » permet d'obtenir l'IP ou le nom de la machine
- « ***resolveip ip*** » permet d'obtenir l'ip d'une machine (ou d'une IP)
- « ***telnet machine port*** » permet d'ouvrir sur une connexion sur un service d'une machine distante

Diagnostiques réseaux : lsof

- Une source d'information importante !!!!
- Obtenir la liste des fichiers ouverts par le processus 1200
  - « ***lsof -p 1200*** »
- Obtenir la liste des ports ouverts par le processus 1200
  - « ***lsof -p 1200 -i 4 –a*** »
- Savoir quel(s) processus sont en contact avec les ports 1 à 1024 de ares.lip6.fr
  - « ***lsof –i @ares.lip6.fr:1-1024*** »
- Savoir quel(s) processus ont ouvert le fichier « ~/foobar »
  - « ***lsof ~/foobar*** »
- Savoir quels sont les fichiers ouverts par l'utilisateur « apache »
  - « ***lsof –u apache*** »

Informations sur le réseau : dig/nslookup

- Interroger un dns pour obtenir un nom de machine ou une ip : « ***nslookup*** »
- « ***dig*** » est identique à nslookup mais il offre plus d'options

```
legond:@eos > nslookup www.lemonde.Fr
Server:         132.227.64.13
Address:        132.227.64.13#53

Non-authoritative answer:
www.lemonde.Fr  canonical name = www.lemonde.fr.d4p.net.
www.lemonde.fr.d4p.net  canonical name = a245.g.akamai.net.
Name:   a245.g.akamai.net
Address: 193.50.203.46
Name:   a245.g.akamai.net
Address: 193.50.203.53
```

Informations sur le réseau

- « ***traceroute*** », « ***traceroute6*** », « ***tracepath*** » et « ***tracepath6*** » permettent de voir le chemin jusqu'à une machine

```
legond@scylla > tracepath  www.lemonde.fr
 1:  scylla (132.227.64.30)                                  0.258ms pmtu 1500
 1:  castor (132.227.64.15)                                  1.519ms
 2:  r-jusren.reseau.jussieu.fr (134.157.254.126)            1.557ms
 3:  gw-rap.rap.prd.fr (195.221.127.181)                   asymm  4   2.292ms
 4:  jussieu-g0-1-165.cssi.renater.fr (193.51.181.102)       2.541ms
 5:  nri-c-pos2-0.cssi.renater.fr (193.51.180.158)           2.364ms
 6:  193.50.203.53 (193.50.203.53)                           3.787ms reached
     Resume: pmtu 1500 hops 6 back 6

legond@scylla > traceroute  www.lemonde.fr
traceroute to a245.g.akamai.net (193.50.203.53), 30 hops max, 38 byte packets
 1  castor (132.227.64.15)  1.334 ms  1.211 ms  1.387 ms
 2  r-jusren.reseau.jussieu.fr (134.157.254.126)  0.821 ms  1.305 ms  0.614 ms
 3  gw-rap.rap.prd.fr (195.221.127.181)  1.760 ms  1.672 ms  1.545 ms
 4  jussieu-g0-1-165.cssi.renater.fr (193.51.181.102) 1.343 ms 0.790 ms 1.184 ms
 5  nri-c-pos2-0.cssi.renater.fr (193.51.180.158)  1.617 ms  1.605 ms  1.479 ms
 6  193.50.203.53 (193.50.203.53)  1.911 ms  1.921 ms  0.893 ms
```

Analyser le réseau

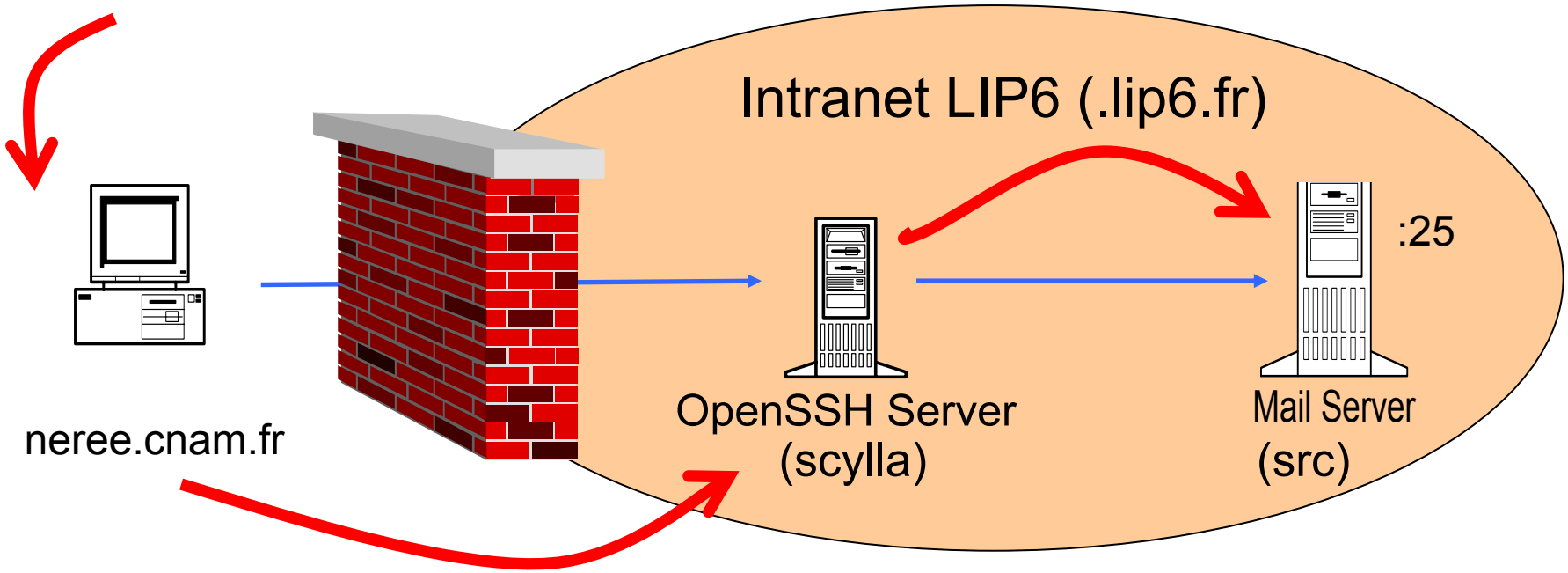
- Commandes SunOS : ***etherfind***, ***snoop***
- « ***tcpdump*** » permet la capture et le filtre des communications entre les machines et/ou les services
  - Capturer toutes les communications provenant de zeus
    - ✓ « ***tcpdump src host zeus*** »
  - Capturer tous les paquets provenant du serveur DHCP
    - ✓ « ***tcpdump udp and port 67*** »
  - Capturer les paquets de gestion du réseau (ICMP)
    - ✓ « ***tcpdump icmp*** »
- « ***ethereal*** » permet la capture et l'analyse du trafic réseau.
  - Des exemples ? RTFM !!
  - Très complexe, Très puissant
- « ***dsniff*** » permet la capture et l'analyse du trafic réseau.
  - Faites ensuite un « telnet pop.free.fr »
- Utilisation « ***nmap*** »

SSH

- Ajout de clefs
  - Création d'une clef RSA ou DSA
  - « ssh-keygen –t rsa –b 2048 –C "comment" »
- En ssh, il existe des noms de clefs prédéfinis
  - identity, id_dsa ou ids_rsa
  - Les fichiers .pub sont les parties publiques (distribuables)
  - Mettre des droits corrects
- Changer le mdp sur la partie privée
  - « ssh-keygen –p monfic »
- Vérifier qu'il n'y a aucune information sur la personne
  - Ce n'est pas un certificat

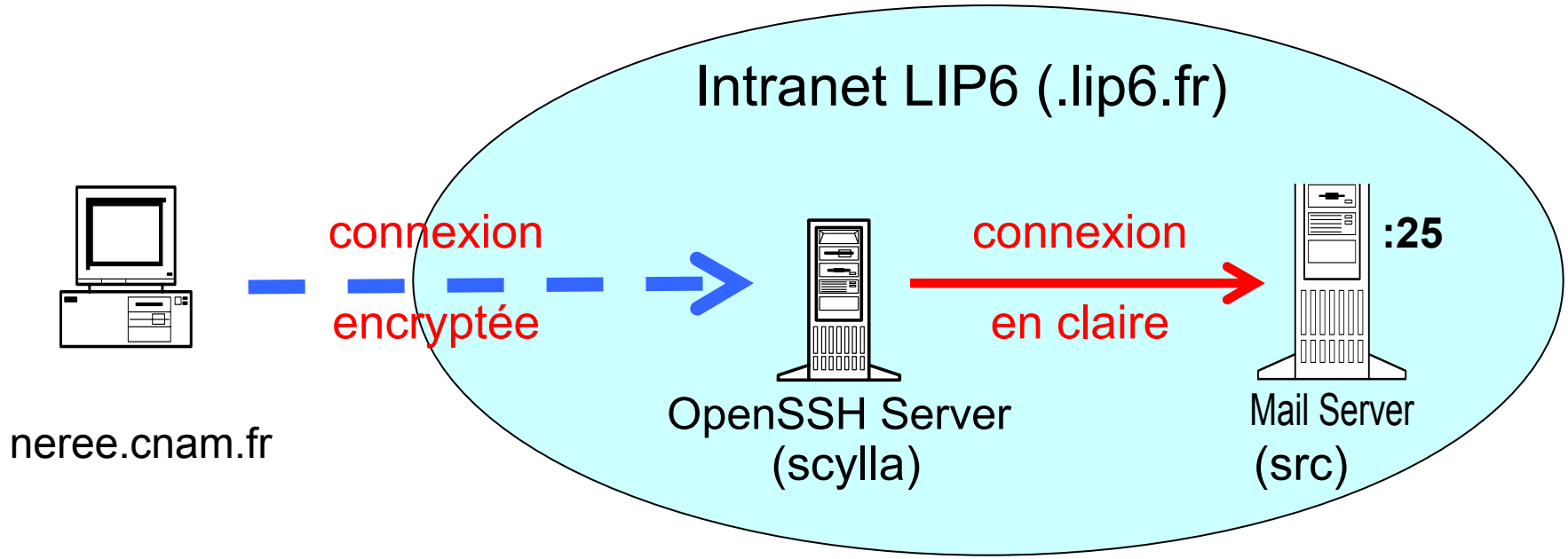
Tunnels sortant (Port forwarding)

- Etablissement d'un tunnel sortant TCP via SSH
  - « -L port_local:hote_distante:port_distant »
  - Toute connexion sur le port « port_local » de la machine local sera transférée sur le port « port_distant » de la machine « hote_distante » à partir de la machine où l'on est connecté
- ***ssh -L 1999:src:25 scylla.lip6.fr***



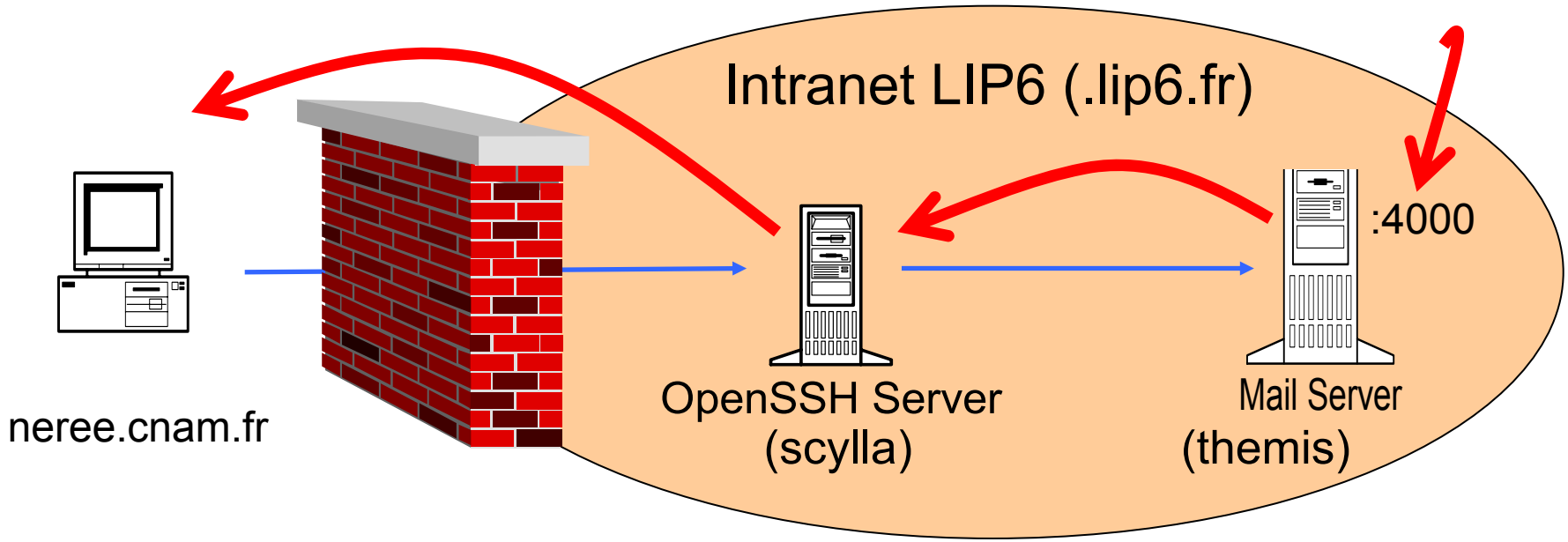
Tunnels sortant (Port forwarding)

- Note :
  - La connexion entre neree et scylla est cryptée par le protocole SSH.
  - La connexion entre scylla.lip6.fr et src.lip6.fr n'est pas cryptée.
  - La connexion entre scylla et src reste "interne" au réseau LIP6 (ce n'est pas une obligation mais un conseil)
  - C'est une « téléportation » de connexion



Tunnels entrant

- Etablissement d'un tunnel entrant TCP via SSH
  - « -R port_local:machine_distante:port_distant »
  - Toute connexion sur le port « port_distant » de la machine distante « hote_distante » sera transférée sur le port local « port_local »
- ***ssh -R 2000:themis:4000 scylla.lip6.fr***



Tunnels SSH : Attention !

- Attention aux risques du “port forwarding” !
  - Permet de contourner les règles de sécurités
  - Contourner les filtres sortants en se connectant à une machine externe au réseau administré
  - Contourner les filtres entrants en se connectant, de l’extérieur, à une machine interne au réseau administré
- Ne pas faire d’expérimentation. Le “port forwarding” doit être penser !
- Seul root peut transférer des ports privilégiés

Session distante

OpenSSH: options du client

- -X/-x : interdire/autoriser le forward x11
- -i : spécifier un clef numérique de connexion
- -g : autoriser la connexion distante sur les ports transférés
- -I : utiliser une carte à puce pour lire la clef numérique
- -N : n'exécute aucune commande, seulement pour le transfert de ports
- -f : passer en arrière plan (utile avec –N)
- -t/-T : (non) allocation d'un pseudo-terminal
- -C : compression
- -o : passer des options du fichier « ssh_config »

OPENSSL

- Il existe aussi d'autres outils : pgp, gpg, ssh-keygen, java, …
- Informations OpenSSL
  - Infos divers: « openssl version –a »
  - Aide « openssl –h » ou « openssl cmd –h »
    - ✓ version, ciphers, speed, errstr ➔ infos sur openssl
    - ✓ dsa, dsaparam, rsa, gendsa, genrsa ➔ manipulation des clefs asym
    - ✓ x509, ca, pkcs7, pkcs8, pkcs12, verify ➔ gestion des certificats
    - ✓ s_client, s_server ➔ de faux clients/serveurs pour le test des certificats
    - ✓ dgst ➔ Création d'un hache sur des données
    - ✓ enc ➔ Encodage/Décodage de données
    - ✓ crl, ocsp ➔ Certificate Revocation Lists
    - ✓ smime ➔ génération de mails signés et/ou cryptés
  - Liste des outils a disposition « openssl ciphers –v » ou « openssl ciphers –v 'HIGH' » ou 'LOW'
  - Test de performances : « openssl speed »

Création d'une clef ASYMETRIQUE

- Création d'une clef RSA
  - Utilisation de ssh-keygen (outils de ssh)
    - ✓ « **ssh-keygen –t dsa –b 2048 –f root_dsakey.pem** »
    - ✓ « -c » pour le commentaire, « -p » encoder la clef
    - ✓ Utiliser la clef ssh ➔ répertoire .ssh
  - Utilisation de openssl pour une clef RSA
    - ✓ « **openssl genrsa 2048 –out root_rsakey.pem** »
    - ✓ « **openssl genrsa 2048 > root_rsakey.pem** » (ancienne version)
    - ✓ « -des, -aes256 » pour encoder la clef
    - ✓ « **openssl rsa -in root_rsakey.pem -pubout -out root_rsakey.pub.pem** »
- Utilisation de openssl pour une clef DSA
    - ✓ « **openssl dsaparam –genkey –out root_dsaparam.pem 2048** »
    - ✓ « **openssl gendsa –out root_dsakey.pem root_dsaparam.pem** »
    - ✓ Chaque clef générée sera différente

Création, obtention d'un certificat

- Création d'un certificat
  - Certificat auto-signé (option –x509)
    « openssl req –new –x509 –key root_rsakey.pem –out root_cert.pem »
  - Demande de certificat
    « openssl req –new –key root_rsakey.pem –out root_certRequest.pem »
  - Autres options (-days, -newkey, -subj)
- Obtenir un certificat de quelqu'un
  - Exemple: www.microsoft.com (https)
  - « openssl s_client -state -ssl3 -connect www.microsoft.com:443 - cipher 'HIGH' »
  - Autres options « –debug, -showcerts, -verify »
  - Essayez avec www.cru.Fr:443 et « –verify 6 »
  - Essayez avec src.lip6.fr:995 et « -verify 6 » (spop)

Vérification/Importation/Exportation de Certificat

- Lancer un serveur de test avec son certificat
  - « openssl s_server -cert root_cert.pem -www -key root_rsakey.pem –accept xxxx»
- Vérifier un certificat (date, auto-signature)
  - « openssl verify certificat.pem »
- Exporter un certificat (format pkcs#12)
  - « openssl pkcs12 –export –in certificat.pem –out certificat.pxs -name ''un nom'' –password »
  - Option « -chain »
- Importer un certificat (format pkcs#12)
  - « openssl pkcs12 –in certificat.pxs –out certificat.pem »

Création d'une entité de certification (CA)

- Création d'un certificat autosigné ou obtention d'un certificat
  - Cf. slides précédents
- Création du fichier de configuration openssl.cnf
  - Exécuter une fois « openssl ca »
  - Repérer le fichier de conf utilisé (extension .cnf)
    « *Using configuration from /usr/lib/ssl/openssl.cnf* »
  - Copier le fichier sous le nom « monopenssl.cnf »
  - Modifier le fichier (private_key, certificat, …)
- Création des répertoires que vous souhaitez utiliser
  - private, certs, crl

Création d'un certificat qui dépend d'un CA

- Création des fichier serial et index.txt
  - « **echo '01' > serial** » et « **touch index.txt** »
- Serial
  - contient le dernier « Serial » utilisé.
  - est incrémenté à chaque émission d'un certificat par le CA
  - sert de numéro de série tatoué sur chaque certificat émis
- Création du certificat à signer (demande de certificat)
  - « **openssl req –new –nodes –newkey rsa:1024 –keyout newclient_rsakey.pem –out newclient_certRequest.pem** »
- Certification :
  - « **openssl ca –config monopenssl.cnf –in newclient_certRequest.pem** -out newclient_cert.pem»
  - Obtention du certificat signé par le CA dans « new_certs_dir » du fichier de conf
- Vérification
  - « **openssl verify –CAfile root_cert.pem newclient_cert.pem** »
  - « **openssl x509 –in newclient_cert.pem -noout –text** »
  - Utilisation dans un service SSL

Exemple : Serveur apache2 TLS

- Mettre dans un SVC web apache (sous Debian)
  - Allez dans /etc/apache2/ssl
  - Récupéré le certificat auto-signé ou signé
  - Allez dans /etc/apache2/ports.conf
    - ✓ Ajouter « Listen 443 » (pour faire écouter apache sur le port HTTPs)
  - Allez dans /etc/apache2/sites-available
    - ✓ Copiez la configuration par defaut (« default ») en default-ssl
    - ✓ Ajoutez les lignes :
      SSLEngine on
      SSLCertificateFile /etc/apache/ssl/nomfichiercertificat.pem
      SSLCertificateKeyFile /etc/apache/ssl/nomfichierclef.pem
      SSLProtocol –all +SSLv2
  - Exécutez a2ensite default-ssl
  - /etc/init.d/apache start
- A la première connexion que se passe-t-il ?
  - Créer un certificat corrigeant le problème

Création d'un résumé pour un fichier

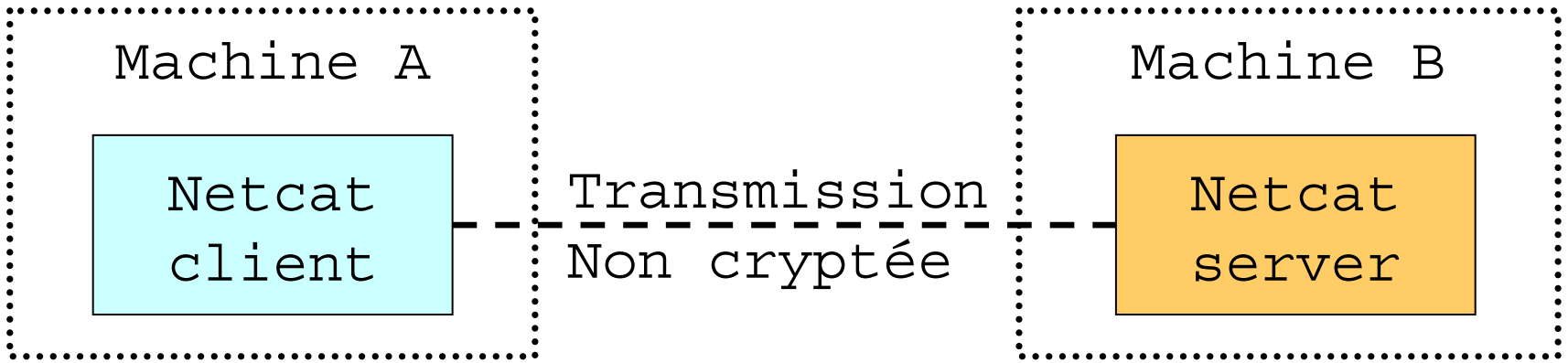
- Rappel :
  - Résumé = « digest »
  - Signature = « signed digest »
  - Résumé n'est pas une signature
  - Signature = Résumé crypté avec sa clef privée
- Résumé : « openssl dgst –md5 fichier »
  - Vous pouvez utiliser aussi md5sum, sha1sum
- « openssl dgst –sign maclef.pem –md5 -out sign.md5 fichier »
- « openssl dgst –md5 –verify maclefpublique.pem –signature sign.md5 fichier »
- Vous pouvez utiliser gpg …

Encodage/décodage de fichier

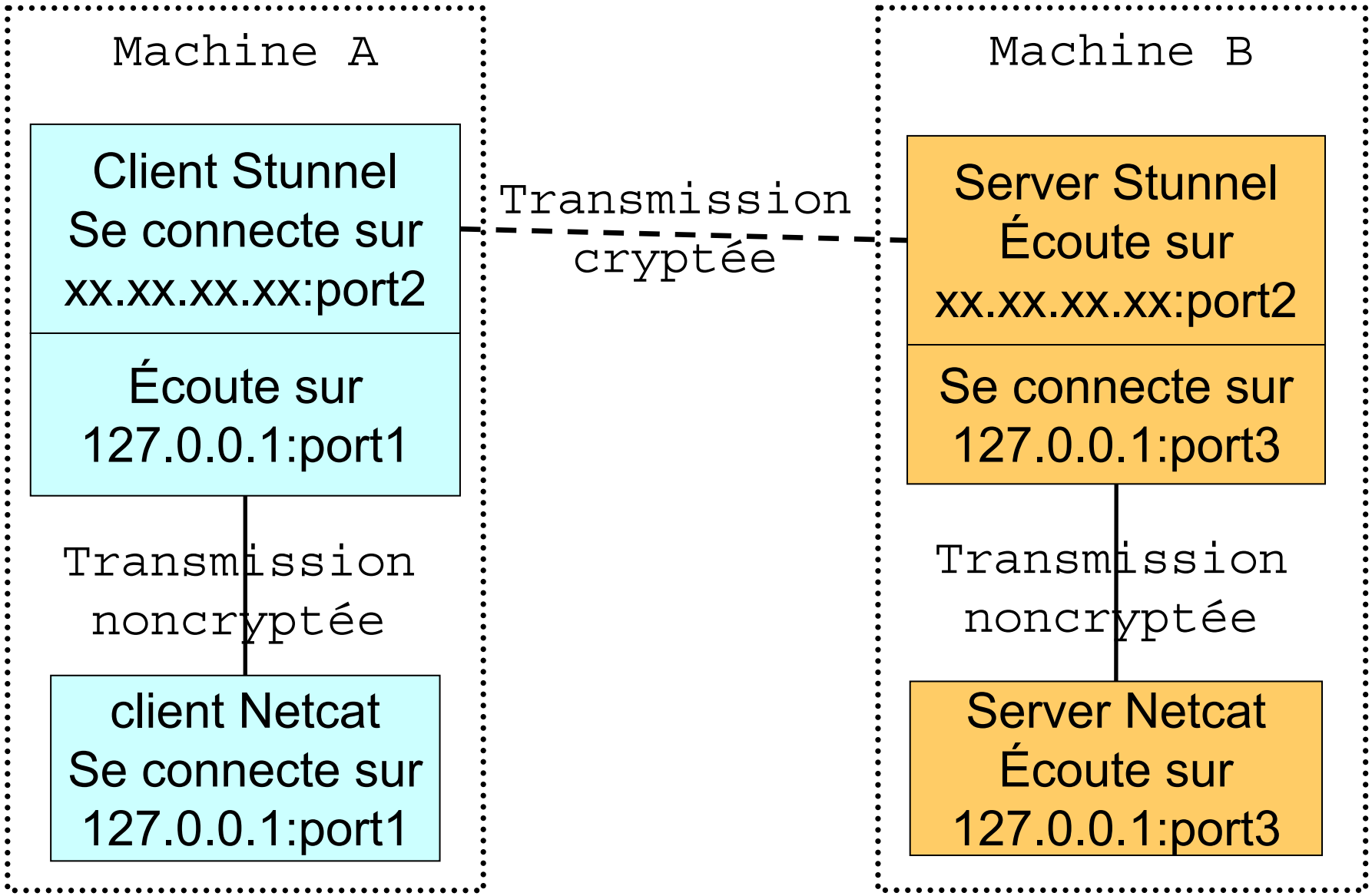
- Constats:
  - On n'utilise pas de clef asymétrique pour le code
  - On les utilise pour signer les fichiers
  - Codage à clef symétrique avec une passe-phrase
  - Mode d'encodage: « openssl enc –h » pour la liste complète
- Encodage par une clef symétrique :
  - « openssl enc –e –in monfichier –out monfichier.enc –salt –mode »
  - « -a » pour encoder en base64 (envoie par email)
  - « -salt » pour introduire de l'entropie
  - « -mode » pour le mode d'encodage/decodage
  - « -pass » pour donner une clef symétrique (cela peut être un fichier)
- Décodage par une clef symétrique :
  - « openssl enc –d –in monfichier.enc –out monfichier –mode »

Création d'un tunnel SSL

- Création d'un client/serveur (netcat, socat, cryptcat)
- netcat
  - A: « netcat –l –p 60000 – e /bin/bash »
  - B: « netcat machineb 60000 »
- Socat
  - A: « socat TCP4-LISTEN:60000,reuseaddr,fork EXEC:'/bin/bash' »
  - B: « socat – TCP4:machineb:80 »
  - Proxy : « socat –d –v TCP4-LISTEN:port1 TCP4:machine:port2 »



Création d'un tunnel SSL



Création d'un client tunnel SSL

- Création du « client » stunnel
  - Génération d'une clef privée et d'un certificat avec openssl
  - Génération d'un fichier de configuration

```
client = yes
CAfile = /chemin/vers/certs/serveurs/autorises/certs.pem
cert = /chemin/vers/certificat/client.cert.pem
key = /chemin/vers/clefprivee/client.privatekey.pem
verify = 3
socket = r:TCP_NODELAY=1
debug = daemon.7
foreground = yes
pid = /chemin/vers/fichierpid/client/stunnel.pid
#output = /chemin/vers/fichierlog/client/stunnel.log
[tunnel60k]
    accept = 127.0.0.1:60001
    connect = x.x.x.x:60002
```

  - Exécution par « stunnel monfichier_client.conf »

Création d'un client tunnel SSL

- Création du « serveur » stunnel
  - Génération d'une clef privée et d'un certificat avec openssl
  - Génération d'un fichier de configuration

```
client = no
CAfile = /chemin/vers/certs/clients/autorises/certs.pem
cert = /chemin/vers/certificat/serveur.cert.pem
key = /chemin/vers/clefprivee/serveur.privatekey.pem
verify = 3
socket = r:TCP_NODELAY=1
debug = daemon.7
foreground = yes
pid = /chemin/vers/fichierpid/client/stunnel.pid
#output = /chemin/vers/fichierlog/client/stunnel.log
[tunnel60k]
    accept = 0.0.0.0:60002
    connect = 127.0.0.1:60003
```

  - Exécution par « stunnel monfichier_serveur.conf »

Création d'un client tunnel SSL

- Test de lancement du client et du serveur stunnel
- Test de connexion entre le client et le serveur stunnel
- Test d'un client stunnel inconnu sur le serveur
  - Prendre un client d'un autre élève
- Ajouter un client au serveur