Administration Système & Réseau

Philippe.Arnould@univ-pau.fr

Plan

- Introduction
- Administration système : Linux
  - Détail des taches
  - Distributions
  - Installation et démarrage
  - Fonctionnement et gestion des paquets
  - Modification de la configuration
  - Noyau et modules
- Administration de services réseau :
  - DHCP, NFS, NIS, Samba, DNS, Postfix

Introduction

- Qu'est-ce qu'un administrateur système ?
- À l'université c'est un métier spécifique :
  - Voir site Referens
  - http://cri.univ-pau.fr
- La journée des administrateurs système :
  - http://www.sysadminday.com/
- Ressources sur wikipédia

Détails des taches de l'administrateur

- Historiquement chaque administrateur avait un cahier ou il notait l'ensemble des informations concernant ses serveurs aujourd'hui on peut utiliser Wiki, base de données, ...
- Quelles sont les informations nécessaire d'enregistrer ?
- le matériel, les interfaces, la taille de la mémoire, les numéros de série, le téléphone de la maintenance,....
- Les logiciels...
- Les problèmes et les solutions...

Détails des taches de l'administrateur

- Pour installer la machine:
  - noter les informations sur les périphériques de la machine
  - les adresses matérielles des cartes et périphériques (@mac, IRQ, E/S, ...)
  - noter tout ce qui concerne la machine (mémoire taille des disques

Détails des taches de l'administrateur

- Préparation pour installer le Système d'exploitation ou un logiciel
  - calculer l'espace disque nécessaire pour chaque application à installer
  - lire la notice d'installation pour regarder si votre système d'exploitation est compatible.
  - préparer les médias d'installation
  - (re) lire le guide d'installation

Détails des taches de l'administrateur

- Installation du système d'exploitation
  - vérifier que l'installation par défaut est valable pour votre système
  - Fixer les interruptions et E/S sur les périphériques si nécessaire
  - suivre l'installation pas à pas et noter tous les choix et les erreurs

Détails des taches de l'administrateur

- Configuration du système d'exploitation et logiciels
  - vérifier les conditions de sécurité sont respectées (port réseau, droits,...)
  - installer le système et les applications
  - créer les utilisateurs, mot de passe
  - faire une sauvegarde du système
  - créer un journal du système

Détails des taches de l'administrateur

- Quotidiennement :
  - surveiller les ressources du systèmes (cpu, mémoire,...)
  - être rigoureux sur l'arborescence et l'emplacement des fichiers
  - mettre à jour les failles de sécurité et suivre le journal système (log)
  - sauvegarder et tester les sauvegardes
  - intervenir sur le réseau (si pas de spécialiste...)
  - former les utilisateurs de la machine :
    - mot de passe, sauvegarde,...

Détails des taches de l'administrateur

- Enregistrer
  - les modifications
  - les problèmes
  - utilisation du système
  - Description du matériel
  - Description des logiciels

Date	Modif système	Cause

Date	Problème	Solutions

Type	
Fabricant	
N° série	
DMA	
IRQ	
Adresse E/S	
Divers	

Détails des taches de l'administrateur

- Commandes Unix à connaitre :
  - gestion des fichiers
    - ls, mv, cp, rm, grep, file, find,...
  - visualisation :
    - cat, more, pg, less, head, tail,...
  - répertoire
    - cd, mkdir, rmdir, pwd...
  - le tout en utilisant :
    - <,>,>>, <<, |
  - éditeur de texte :
    - vi

Plan

- Introduction
- Administration système : Linux
  - Détail des taches
  - **Distributions**
  - Installation et démarrage
  - Fonctionnement et gestion des paquet
  - Modification de la configuration
  - Noyau et modules
- Administration de services réseau :
  - DHCP, NFS, NIS, Samba, DNS, Postfix

Distributions Linux

- Ensemble cohérent et stable de composants d'un système d'exploitation dont l'installation, l'utilisation et la maintenance sont facilitées.
- Intégration d'un logiciel d'installation et des outils de configuration.
- Il existe de nombreuses distributions, chacune ayant ses particularités : certaines sont dédiées à un usage spécifique (pare-feu, routeur, grappe de calcul...), d'autres à un matériel spécifique.
- On peut créer sa propre distribution :
  - Linux from scratch

Distributions Linux

- http://distrowatch.com/
  - Comparatifs, articles,...
- Choix Debian ...

La distribution Debian

Système d'exploitation libre

- noyau Linux : tâches élémentaires

Logiciels applicatifs

- 1500 paquets = logiciel pré-empaqueté dans un format pratique pour l'installation

Les auteurs de Debian :

- Développement basé sur le volontariat
  Les volontaires travaillent en respectant un ensemble de directives développées en coopération lors de discussions sur les listes de diffusion

Les archives Debian

Sites miroir Debian (ftp de UPPA)

/dist : distributions (certains vieux paquets)

/pool : nouvelle place des paquets des versions et pré-version de Debian

/tools : Utilitaires DOS : disquettes, partitionnement…

/doc : Documentation

/indices : Fichier Maintainers et fichiers override

/project : Matériel pour les développeurs (outils en développement…)

Les distributions Debian

Il y a 3 distributions dans le répertoire */dist/*

stable

testing

unstable = sid

Également ***frozen***

La distribution ***stable***

Distribution stable :

/stable/main : paquets de la version la plus récente du système Debian

/stable/non-free/ : paquets dont la distribution est restreinte (licence)

/stable/contrib/ : paquets libres mais dépend d'un paquet non-free

Les problèmes de ***stable***

http://ftp-master.debian.org/

La distribution ***testing***

Paquets enregistrés dans */testing/* après avoir subi des tests dans *unstable.*

- main, contrib et non-free également

Les nouveaux paquets sont rangés dans */pool/*

Paquets :

- avec moins de bogues critiques
- pas de dépendances qui les rendent instables

Distribution prête à être candidate à une sortie

La distribution ***unstable***

Paquets enregistrés dans */unstable/* après avoir été téléchargés

- restent jusqu'à leur passage dans *testing*

Cette distribution contient l'image la plus récente du système en déplacement

- les utilisateurs doivent utiliser et tester ces paquets
  ils sont prévenus de leur état

Avantage :

- toujours dernière version du projet

La distribution ***frozen***

Lorsque *testing* est mûre elle est gelée :

- on accepte plus de nouveau code (seulement des corrections)
- un nouvelle arbre *testing* est crée dans ***/dists/*** avec un nouveau nom de code

Quelques mois de tests : *cycles de test*

- on garde des traces des bugs
- lorsque leur nombre descend en dessous d'une limite fixée :
  - *frozen* devient *stable*
  - précédente *stable* devient obsolète

Les noms de code

Faciliter le travail des miroirs

Sept 2008 : stable est un lien symbolique vers Etch et testing vers Lenny

unstable est un lien symbolique vers *sid* qui est toujours la distribution instable

Noms de code (nom du film Toy Story)

- buzz (Buzz Lightyear) est le cosmonaute,
- rex est le tyranosaure,
- bo (Bo Peep) est la fille qui s'occupe du mouton,
- hamm est la tirelire en forme de cochon,
- slink (Slinky Dog) est le chien
- potato est, bien sûr, Mr. Potato
- woody est le cowboy,
- sarge est un chef des Hommes de l'Armée de Plastique Vert,
- etch (Etch-a-Sketch) est le tableau,
- sid est le garçon d'à côté qui détruit les jouets.
-

Plan

- Introduction
- Administration système : Linux
  - Détail des taches
  - Distributions
  - **Installation et démarrage**
  - Fonctionnement et gestion des paquets
  - Modification de la configuration
  - Noyau et modules
- Administration de services réseau :
  - DHCP, NFS, NIS, Samba, DNS, Postfix

Les types d'installation

CDROM avec disquettes de boot

CDROM bootable (pratiquement tous les ordinateurs)

Clé USB

Installation réseau :

NFS + disquette de boot réseau
Donner le nom de la machine serveur (réseau local)

WWW + disquette de boot réseau
Donner l'adresse web du serveur (éventuellement du proxy)

Installation Debian : http://www.debian.org/releases/stable/

Portable : http://www.linux-laptop.net/

Trouver le matériel

Cartes ISA, PCI (DD, carte vidéo, carte réseau…)

3 méthodes :

ouvrir le PC

trouver avec Debian :

```
lspci -v | less
more /proc/interrupts
more /proc/ioports
```

trouver avec d'autres OS (autre Linux : ex redhat meilleure) ou Windows

Installation

1. Installation de Windows (en premier)

   Partitionner le disque dur, laisser de la place libre pour Linux
   Installer Windows

2. Installation Linux

   Partitionner le disque dur
   - / : système
   - Swap
   - /home : répertoires utilisateurs
   - /usr : logiciels

3. GRUB pour le double boot

Installation Debian

Se reporter a :

http://www.debian.org/releases/testing/installmanual

Préparer les disquettes de boot et root (IDEPCI)
Booter
Partitionner le disque dur (Fdisk…)
Install OS
Installer puis configurer les pilotes
Installer puis configurer le système de base
Installer GRUB
Relancer le système
Install (download all)…

GRUB

- Outil pour installer le secteur de démarrage

  GRUB permet le double boot : Linux/Windows

  fichier /boot/grub/menu.lst installation avec grub-install

```
default         0
title           Ubuntu hardy (development branch), kernel 2.6.24-19-generic
root            (hd0,4)
kernel          /boot/vmlinuz-2.6.24-19-generic root=UUID=1f18e8ba-
        af5f-45b3-9562-6fb2898ec65e ro quiet splash lang=fr
initrd          /boot/initrd.img-2.6.24-19-generic
quiet

# on /dev/sda1
title         Windows Vista/Longhorn (loader)
root          (hd0,0)
savedefault
makeactive
chainloader     +1
```

Démarrage d'un ordinateur

Lancement du BIOS qui interroge les dispositifs déclarés bootables dans le setup

Le premier qui répond envoie son MBR (Master Boot Record) qui est exécuté :
Consultation de la table des partitions du disque
Recherche du premier secteur (Boot Record) de la partition active

Si partition active = DOS : BR=instructions pour lancer Windows
Si partition active = Linux : BR = menu qui permet de lancer le noyau Linux

Noyau = fichier compressé : il détecte le matériel puis monte l'arborescence des fichiers. Enfin il exécute le programme **/sbin/init = scripts de démarrage**

Runlevels

Le *runlevel* par défaut (donné dans /etc/inittab) spécifie les scripts a effectuer par le programme *init*

Linux a 7 *runlevels*

0 : stoppe le système
1 : mode *single-user*
2 à 5 : modes multi-utilisateurs (texte, graphique…)

2 - Multiuser, without NFS
3 - Full multiuser mode
4 - unused
5 - X11

6 : reboot le système

Démarrage : /etc/inittab

- Le noyau démarre le processus init qui lance les actions définies dans /etc/inittab

```
# The default runlevel.
id:2:initdefault:

# Boot-time system configuration/initialization script.
# This is run first except when booting in emergency (-b) mode.
si::sysinit:/etc/init.d/rcS

# What to do in single-user mode.
~~:S:wait:/sbin/sulogin

# /etc/init.d executes the S and K scripts upon change
# of runlevel.

l0:0:wait:/etc/init.d/rc 0
l1:1:wait:/etc/init.d/rc 1
l2:2:wait:/etc/init.d/rc 2 .....
```

/etc/inittab : runlevels

- Par défaut le runlevel est 2 (id:2)
  - Multi-utilisateur
  - Exécution des scripts contenus dans /etc/rc2.d dont le nom commencent par **S**
  - Fichiers de /etc/rc2.d = liens symboliques vers /etc/init.d
  - Les deux chiffres correspondent à l'ordre de lancement
  - init 3 puis init 2 ou reboot pour activer les chgts

ls /etc/rc2.d/

```
K03rhnsd        K45arpwatch     K72autofs    K95kudzu       S17keytable   S85gpm
K05atd          K46radvd        K73ypbind    S08ipchains    S20random     S90crond
K20nfs          K50snmpd        K74ntpd      S08iptables    S26apmd       S90xfs
K35smb          K50snmptrapd    K75netfs     S09isdn        S55sshd       S95anacron
K35vncserver    K50xinetd       K86nfslock   S10network     S60lpd        S98wine
K44rawdevices   K65identd       K87portmap   S12syslog      S80sendmail   S99local
```

Déconnexion

- Déconnexion et arrêt (*volontaire*)
  - Pour se déconnecter, entrer **exit** ou **logout.** Cela relance l'attente de login.
- Ne pas éteindre brutalement
  - Chaque processus actif doit recevoir du noyau la directive de s'arrêter et les systèmes de fichiers doivent être démontés.
  - Coupure brutale : réparations = fsck même avec ext3....
- Arrêter le système, l'administrateur **root** lance :
  - arrêt immédiat : halt (= shutdown -h now)
  - arrêt différé shutdown -h <nb mn>
    - il s'écoule <nb min> minutes entre l'avertissement et l'arrêt.
  - reboot : shutdown -r [<nb mn> | now] ou reboot ou ctrl-alt-del
  - On peut éteindre à l'invite du message : *The system is halted*

Plan

- Introduction
- Administration système : Linux
  - Détail des taches
  - Distributions
  - Installation et démarrage
  - **Fonctionnement et gestion des paquets**
  - Modification de la configuration
  - Noyau et modules
- Administration de services réseau :
  - DHCP, NFS, NIS, Samba, DNS, Postfix

Documentation

Documentation générale

- En tout premier lieu consulter les HOWTO dans /usr/doc/HOWTO
  S'ils n'ont pas été installés,
- Rechercher la doc sur les paquetages installés regroupés dans /usr/doc
- Consulter le "Man" (= manuel) : en ligne de commande, man *commande*

Les répertoires

/ : répertoire racine (contient tous les autres)

/root : répertoire personnel de l'utilisateur root

/home : répertoires utilisateurs

/etc : fichiers de configuration

/bin : exécutables indispensables au fonctionnement du système

/sbin : comme **/bin** mais pour les commandes d'administration

/tmp : fichiers temporaires (souvent créés par le système)

/dev : Périphériques et fichiers spéciaux

Le répertoire /usr

Important

/usr/bin et /usr/sbin : exécutables non vitaux

/usr/man, /usr/doc et /usr/info : manuel, documentation et infos

/usr/local : place pour installer des logiciels

/usr/include : pour le compilateur C

/usr/src/linux : sources du noyau

/usr/X11R6 : X-Window Système

Les répertoires

/var : fichiers variables changés par le système

/var/log : journaux divers (ex : syslog, auth.log à lire régulièrement pour voir les tentatives de piratages)

/var/spool : fichiers en file d'attente

/var/spool/lpd : imprimante
/var/spool/mail : courrier

```
ls /

bin   dev  home    lib         misc  opt   redhat6.0  sbin  usr  zelda0
boot  etc  initrd  lost+found  mnt   proc  root       tmp   var  zelda1
```

Nom des périphériques

Les périphériques sont contenus dans /dev :

/dev/fd0 : premier lecteur de disquette
/dev/fd1 : second lecteur de disquette
/dev/hda1 : première partition sur /dev/hda…

Les types de périphériques :

/dev/hda : disque IDE ou cd-rom sur la première nappe (maître)
/dev/hdb : disque IDE ou cd-rom sur la première nappe (esclave)

/dev/hdc : disque IDE sur la deuxième nappe (IDE maître)
/dev/hdd : disque IDE sur la deuxième nappe (IDE esclave)

/dev/sda : 1er disque SCSI (sdb, sdc…)

/dev/ttyS0 : 1er port série (COM 1) : ttyS1, ttyS2…

Système de fichiers

Organisation des fichiers propres à un OS

MS-DOS : 8 caractères maxi, pas de permission : ***msdos***
Linux = ***ext2*** ou ***ext3***

Un périphérique physique peut contenir plusieurs systèmes de fichiers

pour rendre accessible un système de fichier on lui donne un répertoire dans un autre système de fichier = point de montage
Monter = associer un système de fichier à un point de montage

Ex : montage d'un cdrom sur /dev/cdrom

Le répertoire auquel on accède se trouve sur le DD

Fdisk

Édition et modification de la table des partitions

fdisk /dev/hda1

```
Commande (m pour aide) : p
Disque /dev/hda : 255 têtes, 63 secteurs, 2432 cylindres
Unités = cylindres sur 16065 * 512 octets
Périphérique Amorce    Début       Fin    Blocs   Id  Système
/dev/hda1              1         893    7172991    b  Win95 FAT32
/dev/hda2            894        1913    8193150    f  Win95 Etdue (LBA)
/dev/hda3   *       1947        2432    3903795   83  Linux
/dev/hda4           1914        1946     265072+  82  Echange Linux
/dev/hda5            894        1913    8193118+   b  Win95 FAT32
```

Montages provisoires

Commande ***mount*** :

Exemple : si on a un CD dans le lecteur ***/dev/hdc***

```
mkdir /tmp/cederom
mount /dev/hdc /tmp/cederom
```

Après utilisation, il faudra libéré le disque

```
umount /tmp/cderom
```

On peut préciser le type de système de fichier

```
mount -t vfat /dev/fd0 /tmp/disquette
mount -t vfat /dev/fd0 /tmp/disquette -o ro (read only)
```

Montages automatiques

/etc/fstab indique les montages a effectuer au démarrage

noauto : mount /floppy suffit

```
# /etc/fstab
#
# <device> <mountpoint> <filesystemtype><options> <dump> <fsckorder>

/dev/hdb5         /                ext2     defaults          1        1
/dev/hdb2         /home            ext2      defaults         1        2
/dev/hdc          /mnt/cdrom       iso9660   noauto,ro,user   0        0
/dev/hda1         /mnt/dos/c       msdos     defaults         0        0
/dev/hdb1         /mnt/dos/d       msdos     defaults         0        0
/dev/fd0          /mnt/floppy      ext2      noauto,user      0        0
/dev/hdb4         none             ignore    defaults         0        0
none              /proc            proc      defaults
/dev/hdb3         none             swap      sw
```

Montages automatiques

Device : (/dev/hdc) périphérique physique

Mountpoint : (/mnt/cdrom) point de montage

File system type : (iso9660) type du système de fichier du périphérique

Option : (noauto,ro,user) liste d'options à utiliser lorsqu'on monte le système de fichier

auto, noauto : le SF doit ou non être monté automatiquement
ro : read-only
user : permet à tous les utilisateurs de monter : ex disquette

Dump : indique si un SF doit être pris en compte lors d'un dump

Fsckorder : indique si un SF doit être vérifié

Montages de partitions Windows

VFAT : Windows 98

mount –t vfat /dev/hda1 /windows

NTFS : Windows XP

mount –t ntfs /dev/hda1 /windows

Si ces options ne sont pas supportées : recompiler le noyau en les ajoutant en module ou en dur (voir partie compiler un noyau)

Pour un montage automatique ajouter dans /etc/fstab

```
/dev/hda1     /windows          ntfs     defaults          0        0
```

X Window

X = abstraction du matériel graphique (carte vidéo, souris…)

Les programmes (Clients X) s'adressent au serveur X qui traduit dans le langage du matériel

Le serveur inclut un pilote pour la carte vidéo

Les programmes (Clients X) s'adressent au serveur X qui traduit dans le langage du matériel

Client X spécial : le gestionnaire de fenêtre (fvwm, afterstep…)

Lancer l'environnement X : startx ou xdm, gdm (gnome) ou kdm (KDE)

Recherche de fichiers

whereis *commande*

pour rechercher les fichiers exécutables, les fichiers de configuration, les sources et les pages de manuel de *commande*.
La recherche s'effectue dans /bin, /usr/bin, /etc ...

find *rep* -name *expression*

permet de rechercher les fichiers dans le rép (ou à défaut dans le rép. courant) avec une expression pour sélectionner.

```
whereis w ---> w: /usr/bin/w  /usr/man/man1/w.1
which ftp
find -name smb* recherche d'un fichier de configuration
find /usr -name pine localiser une application
find / -name grasp*
```

Paquets Debian

Ils contiennent les fichiers nécessaires pour implémenter un ensemble de commandes (logiciels)

Paquets binaires

Contiennent les exécutables, fichiers de conf, man pages, licence…
Format spécifique .deb

Paquets sources

Dépendances documentées dans le fichier *control* associé à chaque paquet

Outils : manipuler, gérer, découper, construire, installer les paquets

Priorité des paquets

Priorité assignée par les personnes maintenant la distrib

Required

Paquets indispensables au fonctionnement du système **(à ne pas désinstaller)**

Important

Paquets devant être présent sur tous les système Unix-like : infrastructure de base

Standard

Emacs, une partie de latex…

Optional

X11, latex complet, de multiples applications

Extra : conflit avec des paquets de priorité supérieure

Dépendance des paquets (1)

A *depends* on B

B doit absolument être installé pour que A fonctionne
A dépend d'une certaine version de B (limite inférieure)

A *recommends* B

On juge que la plupart des utilisateurs n'installerons pas A sans avoir les fonctionnalités fournies par B

A *suggests* B

B contient des fichiers en relation avec des fonctionnalités de A

Dépendance des paquets (2)

A *conflicts with* B

A ne fonctionnera pas si B est installé sur le système
Souvent à cause de fichiers de A qui sont des améliorations de fichiers de B

A *replaces* B

Des fichiers installés par B sont remplacés par des fichiers de A

A *provides* B

Tous les fichiers et fonctionnalités de B sont contenus dans A

Outils de gestion

Il existe différents outils de gestion des paquets Debian

dpkg : installation basée sur les fichiers
apt-get : installation centrée sur les archives APT
dselect : outils de gestion des paquets à l'aide de menus

Outils pratiques

apt-cache : recherche une archive de paquet dans le cache local
dpkg-reconfigure : reconfigure un paquet déjà installé
dpkg-source : gère les fichiers de paquets de sources

dpkg

Outil de base pour la manipulation de fichiers .deb

Utilisation

Ramener le fichier .deb (web)
Installer : ***dpkg –i nom_du_fichier.deb***

Autres options

dpkg -r : remove (laisse les fichiers de conf)
dpkg -P : purge (enlève également les fichiers de conf)

man dpkg pour les multiples autres options

Exemple d'installation avec dpkg

Installation d'un logiciel pour visualiser un fichier MPEG

Recherche du nom du paquet (google)

mpg123.deb

Téléchargement de l'archive (site miroir de la Debian)

ftp://debian…/mpg123.deb

Installation

dpkg –i mpg123.deb

APT (1)

APT : interface pour le système de packages de la Debian

- Commandes : apt-get, apt-cache et apt-cdrom

Installation : apt-get

- Recherche sur différents supports : miroir local ou réseau
- Apt-get install et apt-get remove
- Différentes options : paquets de unstable, testing…

Dépendances : récupère automatiquement les paquets avec le drapeau *recommends* mais ne s'occupoe pas des *suggests*.

APT (2)

Exemple

apt-get install telnetd
apt-get install –t unstable libc6
apt-get remove --purge useless-old-package

Système de mise à jour

apt-get update puis
apt-get –u upgrade : récupère les paquets recommandés
apt-get –u dist-upgrade : récupère et vérifie les dépendances

Utilisation d'APT

Recherche du nom du paquet

apt-cache search mpg
Affiche le nom de tous les paquets contenant dans leur nom ou dans leur description la chaîne « mpg »
BD local : /var/cache/apt/

Installation

apt-get install mpg123
Consultation de /etc/apt/sources.list (stable, unstable…)
Téléchargement en fonction dans /var/cache/apt/archives/
Installation : utilise dpkg

```
more /etc/apt/sources.list

deb http://http.us.debian.org/debian stable main contrib non-free
deb http://http.us.debian.org/debian unstable main contrib non-free
```

dselect

Utilitaire fonctionnant au dessus de dpkg et APT : haut niveau

Contrôle avancé sur le choix des paquets (dépendances)

Processus d'installation

- Choix de la méthode d'accès : multi_cd, multi_nfs, multi_mount, apt
- Mise à jour de la liste des paquets
- Sélection des paquets
- Installation et mise à jour des paquets désirés
- Configuration des paquets non configurés
- Suppression des paquets non désirés

Informations : fichiers et paquets

- Information sur les fichiers
  - dpkg –S nom : trouve le paquet à partir du nom de fichier installé
- Information sur les paquets
  - apt-get check : met à jour le cache et vérifie les dépendances
  - apt-cache search texte : cherche un paquet à partir de texte
    - Base de donnée locale
    - Tous les paquets ne sont pas présent en local
  - apt-cache policy paquet : information sur la priorité d'un paquet
  - dpkg –s paquet : état et description d'un paquet installé
  - dpkg –L paquet : liste les noms de fichiers installés par le paquet

Paquets RPM

Format de paquets pour Redhat/Fedora et Mandriva

- Fichier .rpm

- Download du fichier sur Internet
  rpm –i fichier.rpm équivalent de dpkg –i fichier.deb

Avantages du système APT

- Mise à jour automatique
  Recherche et installation automatisée des paquets

Debian, Redhat/Fedora ou Ubuntu

- Version stable de Debian en retard sur les autres (mais stable)
  Ubuntu, Redhat/Fedora, Mandriva plus accessibles aux débutants

Plan

- Introduction
- Administration système : Linux
  - Détail des taches
  - Distributions
  - Installation et démarrage
  - Fonctionnement et gestion des paquets
  - **Modification de la configuration**
  - Noyau et modules
- Administration de services réseau :
  - DHCP, NFS, NIS, Samba, DNS, Postfix

Gestion des utilisateurs

Le fichier /etc/passwd contient la liste des comptes locaux d'une machine

Exemple :

nom_du_compte : mot_de_passe : numero_utilisateur : numero_de_groupe :
commentaire : répertoire : programme_de_demarrage

```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/bin/bash
daemon:x:2:2:daemon:/sbin:/bin/bash
news:x:9:13:News system:/etc/news:/bin/bash
uucp:x:10:14::/var/lib/uucp/taylor_config:/bin/bash
```

/etc/passwd : mot de passe

Mots de passe classiques

- Le mot de passe crypté apparaît dans /etc/passwd
- /etc/passwd accessible en lecture à tous

Problème de sécurité

Options

- * = interdire l'accès
- Vide = autoriser à tout le monde

Les numéros d'utilisateur et de groupe (UID et GID) sont utilisés pour les droits d'accès aux fichiers

- Compte privilégié : UID = 0

Mots de passe shadow

Mots de passe shadow

- x apparaît dans /etc/passwd (pas le password crypté)

Le mot de passe crypté est stocké dans le fichier /etc/shadow

- /etc/shadow accessible en lecture uniquement au root
  Plus de sécurité

L'utilisation de shadow est optionnelle

- Choix à l'installation

Ajouter des utilisateurs

A la main (non shadow):

Ajouter une ligne dans /etc/passwd

```
dur:!:1234:100:Marcel Dur:/home/dur:/bin/bash
```

Créer le home directory

```
cp -a /etc/skel /home/dur
chown -R dur /home/dur
```

Donner un mot de passe

```
passwd dur
```

Commande *adduser* : procédure automatique (shadow)

Commande *useradd* : ligne de commande (options)

Outils graphiques selon les distributions de Linux

Autres commandes

Suppression d'un utilisateur *user*

userdel – r user

pwconv et pwunconv

Crée le fichier /etc/shadow à partir de /etc/passwd
Transforme /etc/passwd

groupconv et groupunconv

Crée le fichier /etc/gshadow à partir de /etc/group
Transforme /etc/group

groupadd et groupdel : ajout et suppression de groupes

Ajouter des utilisateurs : adduser

```
Enter a username to add: philippe
Adding user eric...
Adding new group philippe (1000).
Adding new user philippe (1000) with group philippe.
Creating home directory /home/philippe.
Copying files from /etc/skel
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully  (shadow non activé)
Changing the user information for philippe
Enter the new value, or press return for the default
        Full Name []: Philippe Arnould
        Room Number []: 7879
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [y/n] y
```

Shadow ou non

/etc/passwd (non shadow)

```
eric:U5jK4WnaGix5g:1000:1000:Eric G:/home/eric:/bin/bash
```

Pas d'entrée dans /etc/shadow

/etc/passwd (shadow)

```
eric:x:1000:1000:Eric G:/home/eric:/bin/bash
```

Mot de passe crypté dans /etc/shadow (U5jK4WnaGix5g)
Autre avantage : on peut désactiver le compte (*) sans effacer le mot de passe

Sécurité

Utiliser des mots de passes compliqués

Pour vérifier que vos administrés utilisent des passwords complexes (diminue les risques de piratages) :

Installer le paquet ***john*** : cracker de mot de passe
Lancer en root
Vérifier quels sont les utilisateurs utilisant des mots de passes trop simples.

```
john –incremental /etc/shadow (si shadow)
john –incremental /etc/passwd (sinon)
```

Ces commandes affichent la liste des utilisateurs dont le mot de passe a été craqué

Shell

Interpréteur de commande :

Permet de lancer les programmes
csh, ksh, zsh, bash…
Gestion et surveillance des processus
(pour le root)

Variables d'environnement

printenv

```
PWD=/root
USER=root
LANG=fr_FR
COLORTERM=
DISPLAY=:0.0
LOGNAME=root
SHELL=/bin/bash
HOME=/root
TERM=xterm
PATH=/sbin:/usr/sbin:/bin:/usr/bin:/
usr/X11R6/bin:/usr/local/bin
```

PATH

Indique où résident les commandes

Trop long de parcourir tout le disque et trop limité de se contenter des exécutables du répertoire dans lequel on se trouve

Variable définie dans /etc/profile : applicable aux utilisateurs

Personnifiable dans ~/.bashrc…

```
export PATH=$PATH:.:/zelda0/JDK/jdk1.2/bin/
```

Ordre des répertoire dans le fichier = ordre de recherche

Connexion

L'interpréteur de commande (shell) associé est lancé automatiquement dès la saisie du login utilisateur.

exécute des scripts globaux à tous les utilisateurs et des scripts liés au compte et qui permettent une personnalisation.

affiche le prompt et se met en attente d'une commande jusqu'à la commande **exit**,

Scripts de connexion

Les scripts de connexion

Script **/etc/profile** *communs à tous les users y compris* root
Puis exécution de **$HOME/.bash_profile**. Il s'agit ainsi d'un fichier de démarrage personnel et paramétrable.

Ou exécute $HOME/.bashrc dans lequel il est recommandé de placer toutes les fonctions ou alias personnels…

Puis le prompt utilisateur s'affiche et le shell attend une commande ...

Création de compte (adduser)

Scripts copier dans le nouveau home directory depuis /etc/skel
Scripts modèles modifiables par root

Personnaliser le shell

/etc/bashrc est le dernier script d'initialisation du shell bash. Il contient des alias redéfinissables ou à compléter par l'utilisateur root.

```
alias l=""ls --color=tty -F -b -T 0"
alias ll="l -l"
alias lp="ll | more"
alias la="ll -a"
alias x="startx"
alias m="mc -c"
```

Ou modifier .bashrc locaux (exemple : modifier le PATH)

source .bashrc

Exemple : enlever le bip

Readline : bibliothèque de saisie de caractères avec historique et complétion utilisée par de nombreux programmes (bash, ncftp, gnuplot...).

Fichier ~/.inputrc

```
# Permettre de rentrer & recevoir des caractères accentués
     set meta-flag on
     set convert-meta off
     set input-meta on
     set output-meta on

# Pas de bip audible mais visible
     set bell-style visible
```

Permissions

ls –l /etc/resolv.conf

Mode, nb de liens ou de fichier dans un répertoire, propriétaire et groupe propriétaire, taille, date.

```
rw-r--r--  1 root  root  47 2002-07-08 10:09 /etc/resolv.conf
```

/etc/group : fichier des groupes

Modes :

Lecture, écriture, exécution (rwx)
Pour le propriétaire, le groupe, les autres

Modifier les permissions

chmod u+x monfichier

Ajoute les permission d'exécution à l'utilisateur

chmod go-r monfichier

Retire les permissions de lecture au groupe et aux autres

chmod ugo=rx monfichier

rx activées pour l'utilisateur, le groupe et les autres

chmod 777 monfichier

Nombre de 12 bits écrit en octal : r = 0444 = 000 100 100 100

MC : Midnigth Commander

Installer : apt-get install mc

Plusieurs fonctionnalités

- Gestionnaire de fichier (permissions, emplacements…)
- Éditeur (copier-coller…)
- Visionneur (chercher des mots dans des fichiers)
- Démarrage automatique
  - exécution
  - man, html : trouve le bon visualiseur
  - tar, gz : affiche le contenu comme sous répertoires
- Système de fichier FTP virtuel

Sauvegardes (1)

Copier un répertoire entier

```
cp –a /source/directory /dest/directory
```

Commande tar

```
tar cvf sauvegarde.tar /home/eric
tar xvf sauvegarde.tar
```

c = crée une nouvelle archive
v = mode verbeux (explications des actions pendant la création)
f = chaîne suivante = nom du fichier à créer ou du périphérique à utiliser (tar cvf /dev/fd0 /home/eric)
z=compresse ave gzip
j=compressse avec bzip2

Sauvegardes (2)

Pax : nouvel utilitaire d'archivage muti-plateforme

```
apt-get install pax
man pax
```

Compression : commande *gzip* et *gunzip (bz2, zip…)*

```
gzip monfichier
gunzip monfichier.gz
```

Avec tar (pas de compression) : transformation d'un répertoire en une archive

```
tar zcvf monfichier.tgz /home/eric
mcopy monfichier.tgz a:
tar zxvf monfichier.tgz
```

Planification de tâches

Lancement de commandes périodiques consommant des ressources à des heures creuses (sauvegardes…)

crontab –e

Crée ou édite un fichier crontab pour programmer des événements régulièrement

echo tar cvf /dev/st0 | *at 3:40 monday*

Diffère une sauvegarde sur bande lundi à 3h40

Démon cron

Le démon ***cron*** se réveille chaque minute et vérifie les tâches à effectuer dans :

/etc/crontab
/etc/cron.d
/var/spool/cron/crontabs/*user*

crontab ma_crontab

ma_crontab sera la crontab par défaut pour l'utilisateur ayant fait la commande

crontab

more /etc/crontab

```
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
HOME=/
# minute heure jourdumois mois jourdelasemaine commande

01 * * * * root run-parts /etc/cron.hourly
02 4 * * * root run-parts /etc/cron.daily
22 4 * * 0 root run-parts /etc/cron.weekly
42 4 1 * * root run-parts /etc/cron.monthly
```

ls /etc/cron.daily/

```
00-logwatch  logrotate         rpm            tetex.cron
0anacron     makewhatis.cron   slocate.cron   tmpwatch
```

Fichier /etc/crontab

crontab du root

Minute : 0-59
Heure : 0-23
Jour du mois : 1-31
Mois : 1-12 ou jan, feb…
Jour de la semaine : 0-7 ou sun, mon…

run-parts /repertoire

Lance tous les fichiers exécutables du répertoire donné en argument

Démons

Démons lancés en mode multi-utilisateur : /etc/init.d

```
calif:/etc/init.d# ls
apache          atd             networking      rmnologin
bind            ifupdown        nfs-common      samba
inetd           sendmail        nis             sendsigs
pcmcia          cron            kerneld         portmap
ppp             squid           dhcp            klogd
ssh             dns-clean       logoutd         quota
lpd             quotarpc        umountfs        halt
```

Démarrer (start), redémarrer (restart), stopper (stop)

```
/etc/init.d/samba stop
```

Démons : exemple

gpm : gestion de la souris en mode texte

`/etc/init.d/gpm stop` = plus de souris en mode texte
`/etc/init.d/gpm start` = pour relancer

networking : gestion des communications réseau

`/etc/init.d/networking stop` = plus de communication (ex : ping)
Attention : commande refusée si des partitions nfs sont montées.
`/etc/init.d/networking restart` = si on veut une prise en compte d'une modification du n°IP dans le fichiers /etc/network/interfaces

inetd

Un démon important : services réseau (telnet, ftp…)

Fichier de configuration : /etc/inetd.conf

```
# <service_name> <sock_type> <proto> <flags> <user> <server_path> <args>

#:STANDARD: These are standard services.
#<off># telnet   stream  tcp     nowait  telnetd.telnetd /usr/sbin/tcpd
/usr/sbin/in.telnetd

#<off># smtp    stream  tcp     nowait  root    /usr/sbin/sendmail sendmail -Am
-bs
```

inetd : exemple avec telnetd

- On recherche le nom du package du serveur telnet
  - apt-cache search telnet = telnetd
  - apt-get install telnetd
  - more /etc/inetd.conf

```
#:STANDARD: These are standard services.
telnet    stream  tcp     nowait  telnetd.telnetd /usr/sbin/tcpd
/usr/sbin/in.telnetd
```

- On relance le démon inetd (prise en compte des modifs)
  - /etc/init.d/inetd restart

Les fichiers de log

- Répertoire ***/var/log***

```
/var/log# ls
    apache       exim       ksymoops  mail.err   news      user.log
    auth.log     faillog    lastlog   mail.info  samba     uucp.log
    daemon.log   installer.log  lp-acct   mail.log   scrollkeeper.log  wtmp
    debug        kdm.log lp-errs   mail.warn  squid             XFree86.0.log
    dmesg        kern.log       lpr.log   messages   syslog

    /var/log/samba# ls
    log.nmbd  log.smbd

    more /var/log/syslog

    #dernière ligne après une tentative de connexion en telnet
Aug 28 09:55:52 calif in.telnetd[402]: connect from calif.univ-fcomte.fr
```

/var/log/dmesg : contient le compte-rendu de l'initialisation.

Vies des processus

Noyau Linux : lance, gère les processus et contrôle leur échanges avec les périphériques.

Il tient à jour une table des processus en exécution
Le premier processus, ancêtre de tous les autres est **init.**
Tous les processus sont créés par un parent et appartiennent à un utilisateur.
Chacun est identifié par un numéro, son **PID**

Il peut être important de connaître le PID d'un processus, ne serait-ce pour pouvoir le "tuer", s'il ne répond plus et bloque une console

ps : liste des processus
ps aux : donne tous les processus, avec leur numéro PID
ps aux | less : pour contrôler le défilement
ps aux | grep X11 : pour n'afficher que certaines lignes
kill PID : met fin normalement à la tâche
kill -9 PID : action si nécessaire encore plus radicale !

Gestion des processus (1)

ps aux

"USER" à quel utilisateur appartient le processus.
"PID" est le numéro qui identifie le processus
"%CPU" en % les ressources du microprocesseur utilisées
"%MEM" en % les ressources en mémoire vive utilisées
"RSS" mémoire réellement utilisée en ko par le processus.
"START" l'heure à laquelle le processus a été lancé.

pstree | less permet de visualiser la filiation des processus sous forme arborescente.

pidof httpd, pour connaitre la liste des PID des processus d'un programme

Le numéro PID d'un service est souvent stocké dans un fichier qui porte son nom, dans le répertoire **/var/run**

Gestion des processus (2)

Envoi de signaux par l'intermédiaire des commandes kill et killall

Principales actions que l'administrateur peut être amené à utiliser

kill -15 PID : demande normale d'arret au processus, il peut refuser (-15 peut etre remplacé par SIGTERM)

killall -9 httpd : suppression plus radicale, en cas de processus récalcitrant ! (Le signal -9 par exemple s'appelle SIGKILL

kill $(pidof ypserv) : supprime le processus serveur NIS, dont le pid est obtenu le résultat de la commande pidof

killall -HUP httpd : ordonne au processus de relire son fichier de configuration, ce qui évite de le relancer.

Configuration réseau : IP

Configuration de l'interface réseau eth0 : *ifconfig*

```
ifconfig
eth0      Lien encap:Ethernet  HWaddr 00:00:86:4F:CE:8D
          inet adr:194.57.88.150  Bcast:194.57.88.255  Masque:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:31163 errors:0 dropped:0 overruns:0 frame:0
          TX packets:4581 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 lg file transmission:100
          RX bytes:2507508 (2.3 MiB)  TX bytes:299222 (292.2 KiB)
          Interruption:11 Adresse de base:0xd400
```

Changer la configuration

```
ifconfig eth0 194.57.88.151 netmask 255.255.255.0 up
```

Fichier de configuration

Fichier de configuration : /etc/network/interfaces

/etc/init.d/networking restart : si modifications

```
more /etc/network/interfaces
auto eth0
iface eth0 inet static
        address 194.57.88.150
        netmask 255.255.255.0
        network 194.57.88.0
        broadcast 194.57.88.255
        gateway 194.57.88.254
```

⇨ Modif : route add default gw 194.57.88.254

Masque de sous-réseau

- Permet de déterminer à quel réseau ou sous-réseau appartient une machine

IP A : 194.57.88.150
Masque : 255.255.255.0
Network A : 194.57.88.0

IP B : 194.57.88.156
Masque : 255.255.255.128
Network B : 194.57.88.128

194 .57 .88 .10001110
255 .255.255.10000000
194 .57 .88 .10000000 = 194.57.88.128

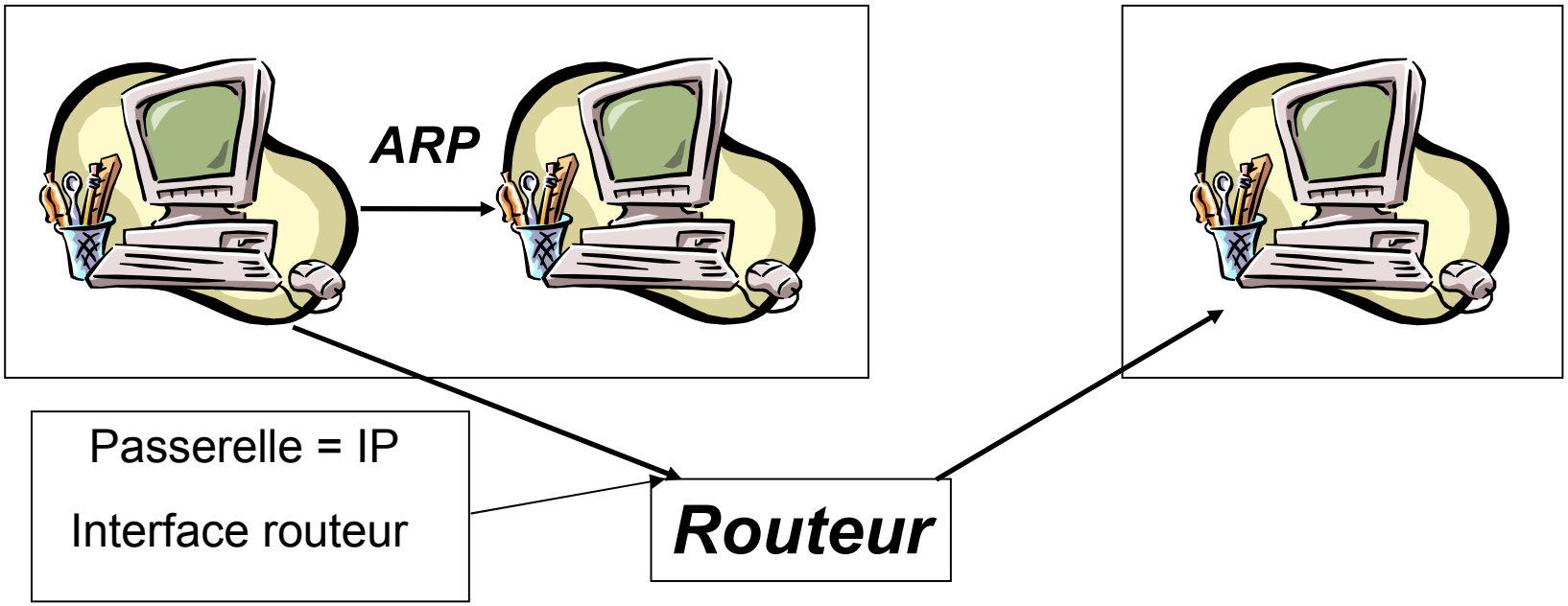
Passerelle

Communication entre deux machines

Et logique entre les adresses et le masque pour déterminer si elles appartiennent au même réseau

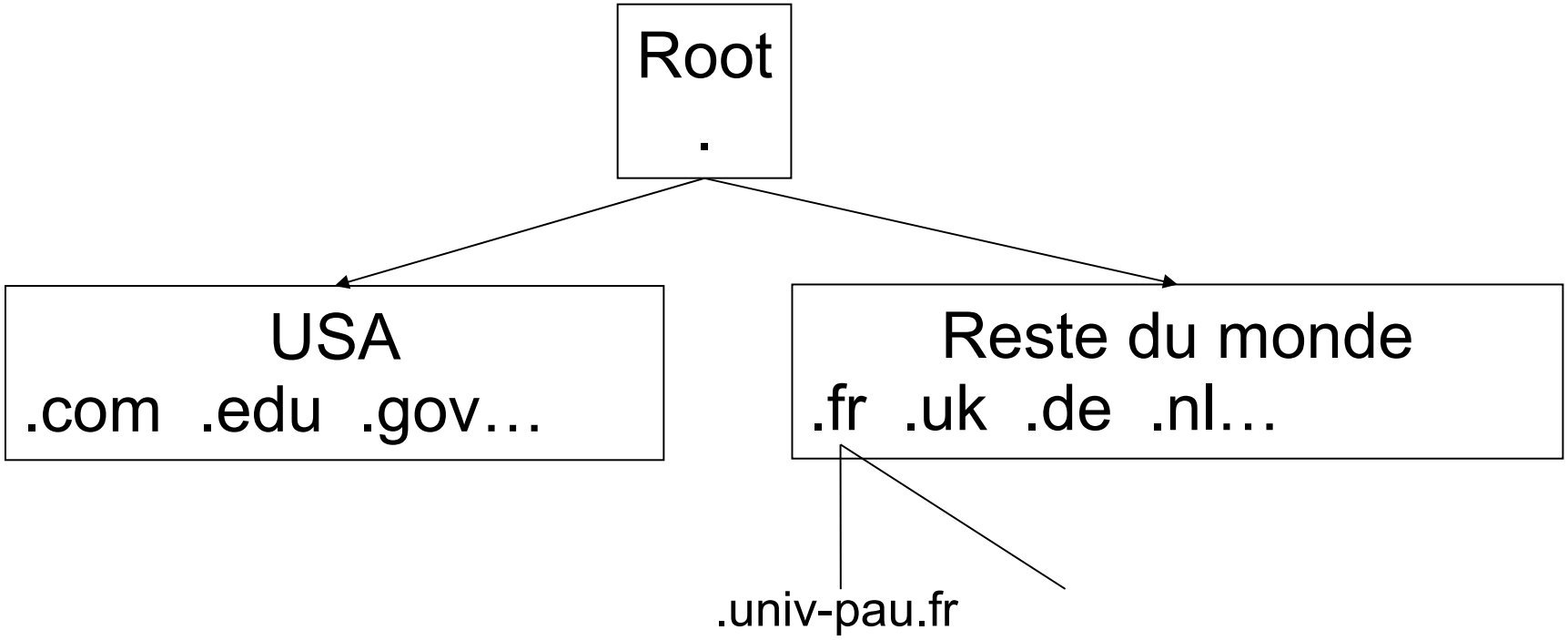
Même réseau : ARP sur le réseau local pour avoir l'adresse MAC de la machine destination

Réseau différent : envoie du message vers la passerelle



DNS

- Association nom de machine : No IP
  - www.univ-pau.fr = 194.167.156.240



Configuration réseau : DNS

Fichier de configuration : client DNS

```
more /etc resolv.conf
*** /etc : répertoire ***
::::::::::::::
resolv.conf
::::::::::::::
search univ-pau.fr
nameserver 194.167.156.13
```

search indique le domaine par défaut

DHCP

Fichier de configuration : /etc/network/interfaces

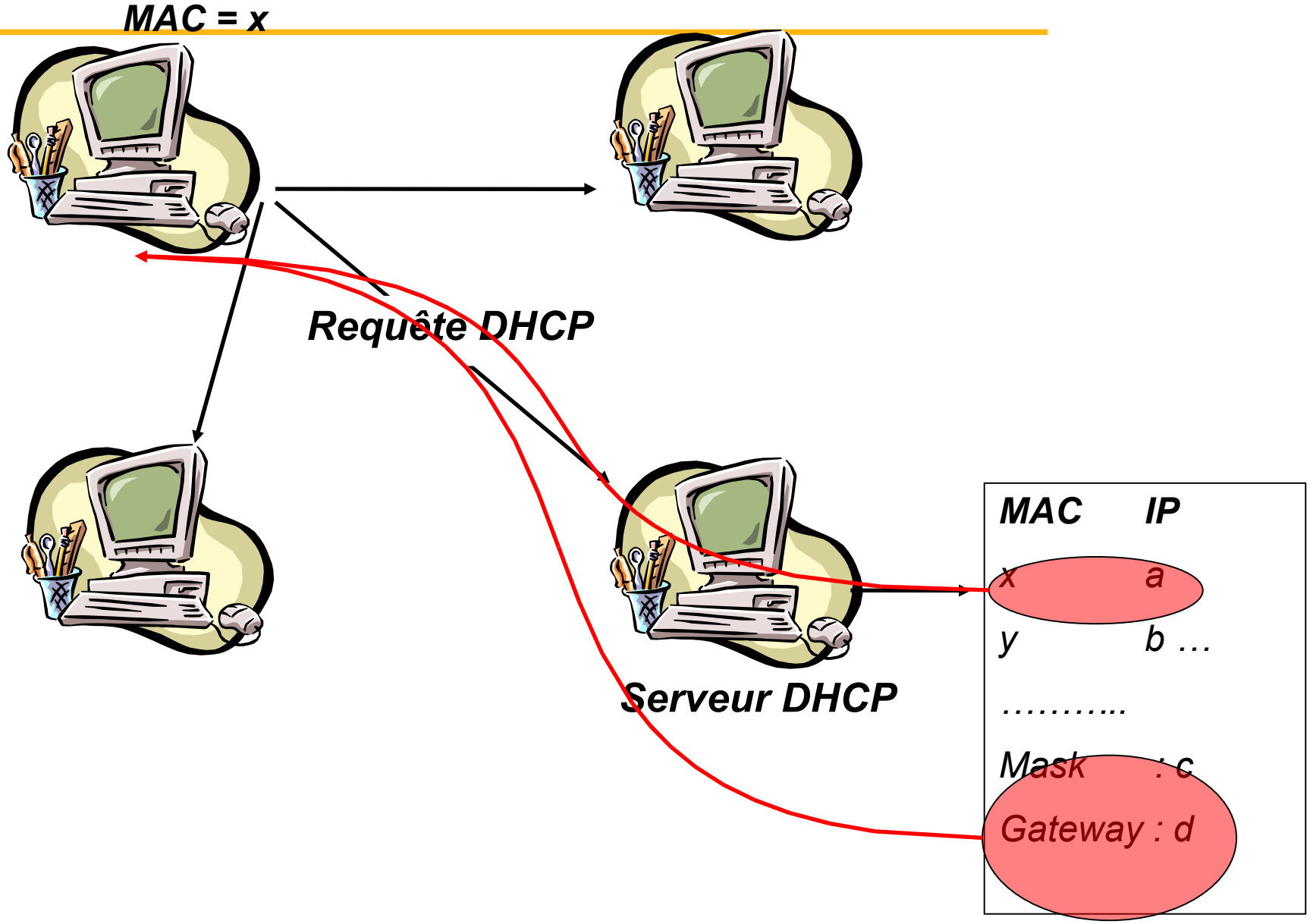
```
more /etc/network/interfaces
# /etc/network/interfaces -- configuration file for ifup(8), ifdown(8)
auto eth0
iface eth0 inet dhcp
```

Configuration automatique de l'adresse ip, de la passerelle… à chaque démarrage de la machine (ou restart de networking)

Système utilisé à l'IUT

- La machine boot
- Lancement du démon networking = lecture de /etc/network/interfaces
- Configuration par DHCP

DHCP : principe



Configuration réseau : NIS (1)

Client NIS : commandes */etc/init.d/ypbind* ou */etc/init.d/nis* suivant les distributions et les versions

Établir le nom de domaine

```
nisdomainname mdm
```

Serveur NIS

Trouvé par un broadcast
Ou dans le fichier de configuration client nis : /etc/yp.conf

Relancer le démon

```
/etc/init.d/nis restart
```

Configuration réseau : NIS (2)

- Fichier /etc/yp.conf

```
# yp.conf  Configuration file for the ypbind process. You can define
#          NIS servers manually here if they can't be found by
#          broadcasting on the local net (which is the default).
#
#         See the manual page of ypbind for the syntax of this file.
#
# IMPORTANT: For the "ypserver", use IP addresses, or make sure that
#           the host is in /etc/hosts. This file is only interpreted
#           once, and if DNS isn't reachable yet the ypserver cannot
#           be resolved and ypbind won't ever bind to the server.
ypserver goldorak.univ-pau.fr
```

- **ypwhich** = goldorak.univ-pau.fr
- **ypcat auto.home** =

```
-rw,soft          goldorak:/home/goldorak1
-rw,soft          goldorak:/home/goldorak0
```

Commandes yp

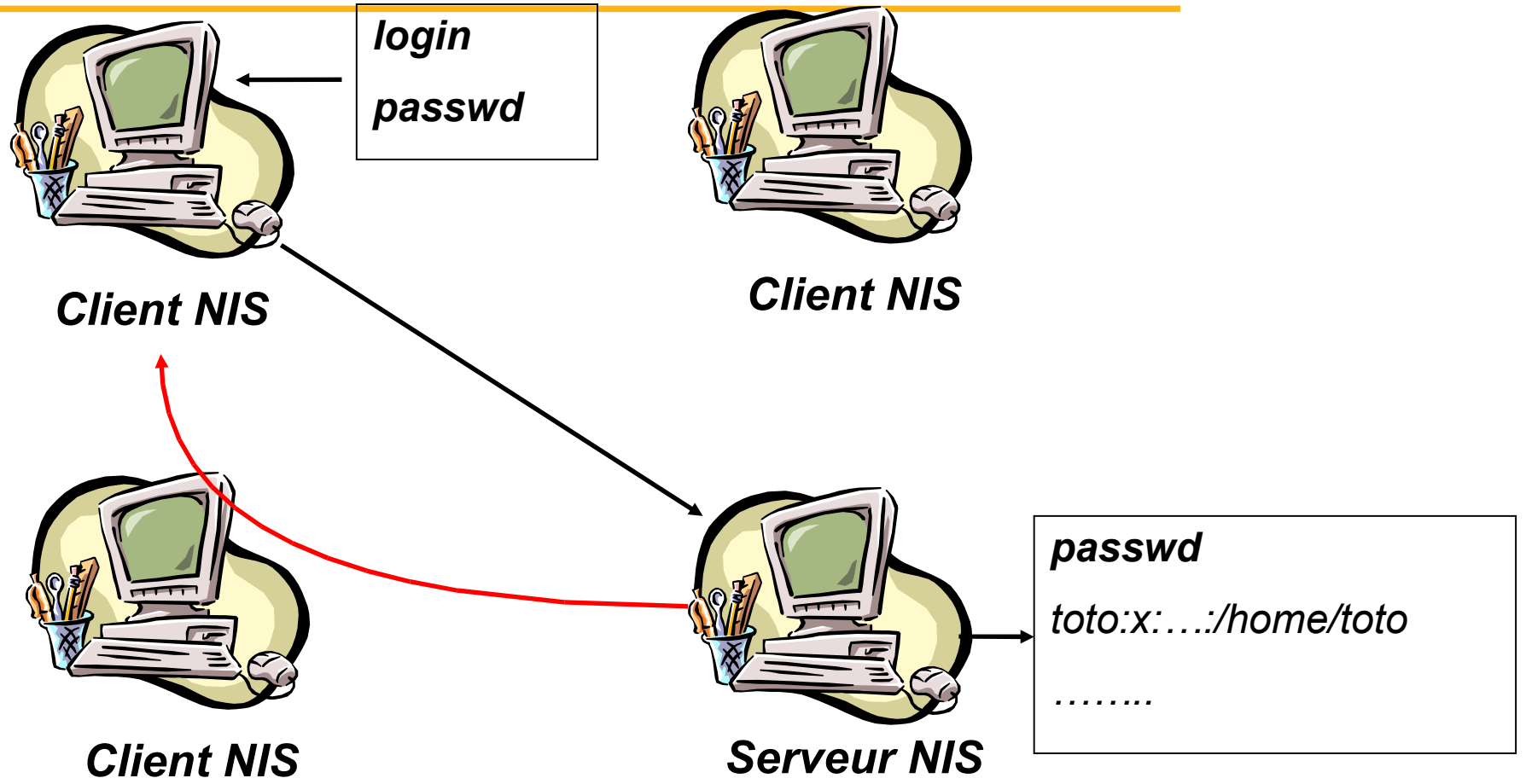
Toutes les commandes commençant par ***yp*** sont envoyées au serveur NIS qui les traite

`ypwhich` : questionnement du serveur sur son nom

`yppasswd` : envoi au serveur d'une demande de changement de mot de passe
ATTENTION : la commande `passwd` ne fonctionne pas avec des comptes NIS

`ypcat` : questionnement du serveur sur ses tables…

NIS : principe côté client



ypbind pour lier un client à un serveur NIS

Imprimantes

Système de gestion des imprimantes : LPR

- Reçoit les requêtes d'impression
- Stocke les travaux dans la file
- Mise en forme, conversion pour le type d'imprimante utilisé
- Les travaux peuvent être transmis à une autre machine ou à une imprimante réseau

Fichier de configuration : /etc/printcap

Commandes

- lpr : imprime
- lpq : travaux dans la file
- lprm : supprime des travaux

lpq, lprm

```
>lpq
Printer: laserjet4050@zelda (dest laserjet4050@laserjet4050)
 Queue: no printable jobs in queue
 Server: no server active
 Status: job 'jc@zelda+313' removed at 11:53:38.733
 Rank   Owner/ID                  Class Job Files                 Size
    Time
done   garcia@zelda+516             A   516 (STDIN)             1105969
    15:17:05

>lprm 516
Printer laserjet4050@zelda:
  checking perms 'garcia@zelda+516'
  dequeued 'garcia@zelda+516'

>lpq
Printer: laserjet4050@zelda (dest laserjet4050@laserjet4050)
 Queue: no printable jobs in queue
 Status: job 'jc@zelda+313' removed at 11:53:38.733
```

/etc/printcap

```
# nom court de l'imprimante locale
    lp:\
            # nom du répertoire de la file d'attente (sd=spool directory)
           :sd=/var/spool/lpd/lp:\
            # la taille maximum du fichier est illimitée (car 0)
            :mx#0:\
            # pas de page de séparation
           :sh:\
           # nom du fichier spécial pour printer locale
           :lp=/dev/lp0:\
           # nom du fichier de traitement du fichier
           :if=/var/spool/lpd/lp/filter:

# nom court de l'imprimante REMOTE
    lp:\
           # nom du répertoire de la file d'attente
           :sd=/var/spool/lpd/lp:\
            ..............................
           # nom du serveur d'impression  distant (rm=remote machine)
           :rm=pc1.cfipen.fr:\
           # nom de l'imprimante distante (rp=remote printer)
       &nbModifications, ajout d'une imprimantesp;
             :rp=lp:\
```

Impression : CUPS

The Common UNIX Printing SystemTM

http://localhost::631/

Permet de faciliter la gestion des imprimantes sous Unix

http://cups.org

Administration graphique

Dans le temps : tout faire à la main, en mode ligne de commande

Aujourd'hui des outils facilitent les tâches d'administration

interfaces conviviales et en automatisation de procédures.
démocratisation de Linux

Ces aides n'excluent pas un minimum de connaissances

Il y a plusieurs acteurs majeurs dans le monde des outils d'administration :

Webmin
YaST (Suse)
…

Webmin

Interface web pour l'administration générale

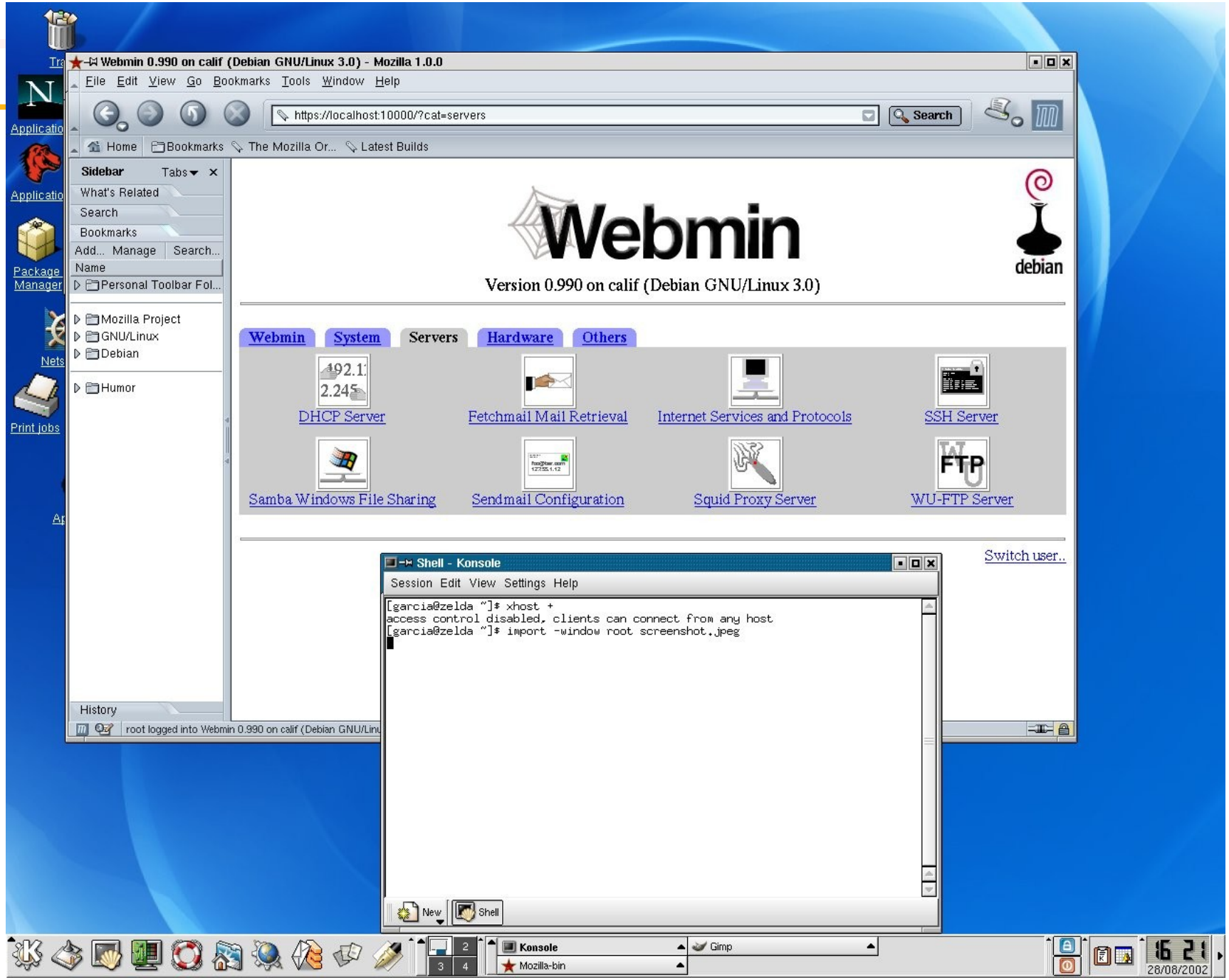
apt-cache search webmin

```
webmin - Web-based administration toolkit

webmin-squid - squid control module for webmin
webmin-sshd - SSH server control module for webmin
webmin-status - server and system status control module for webmin
webmin-stunnel - stunnel control module for webmin
webmin-usermin - usermin control module for webmin
webmin-wuftpd - wu-ftpd control module for webmin
webmin-xinetd - xinetd control module for webmin
webmin-filemanager - file manager module for webmin
webmin-sentry - portsentry module for webmin
webmin-telnet - telnet module for webmin
```

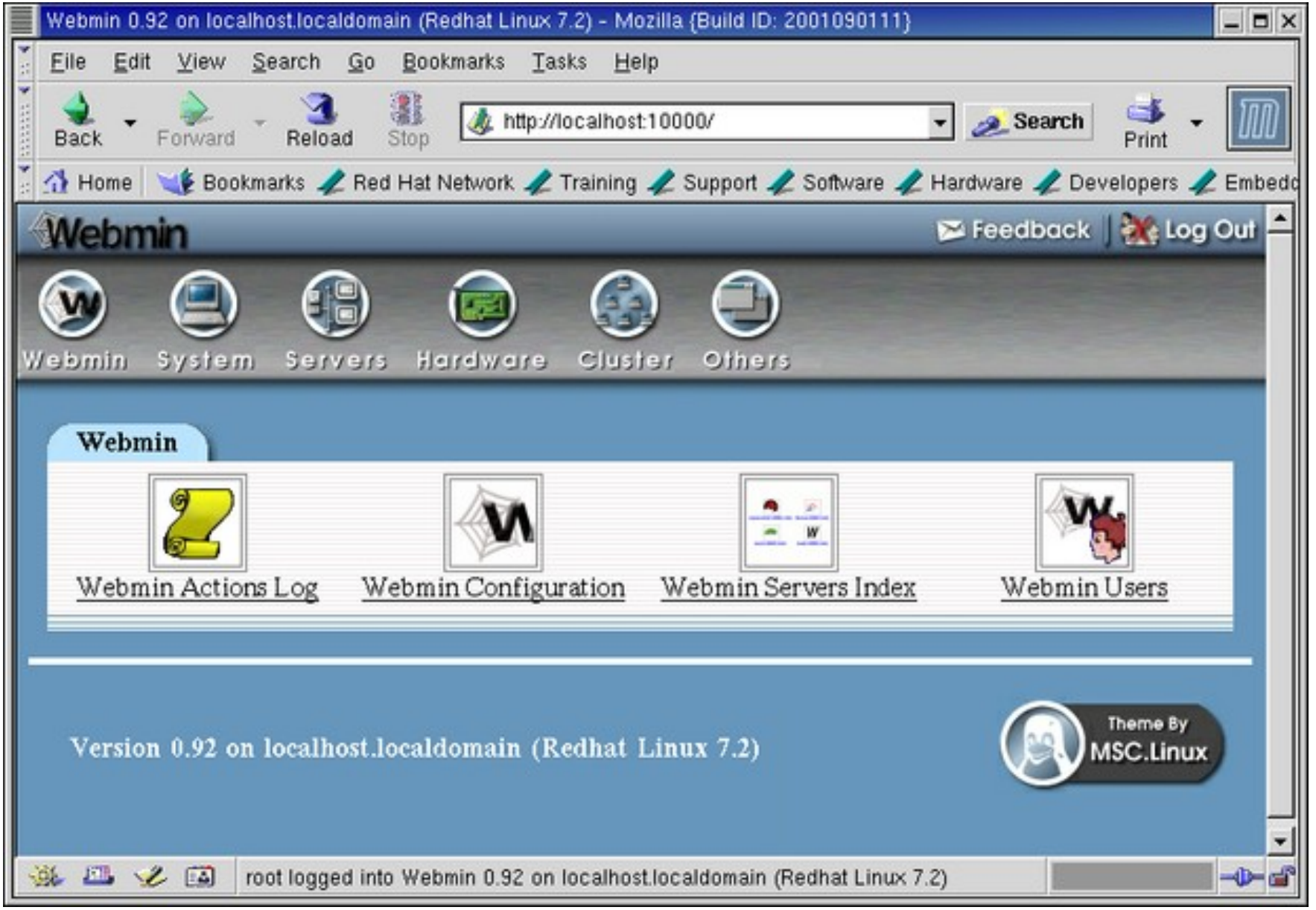
https://localhost:10000 pour lancer l'outil

Webmin : copie d'écran



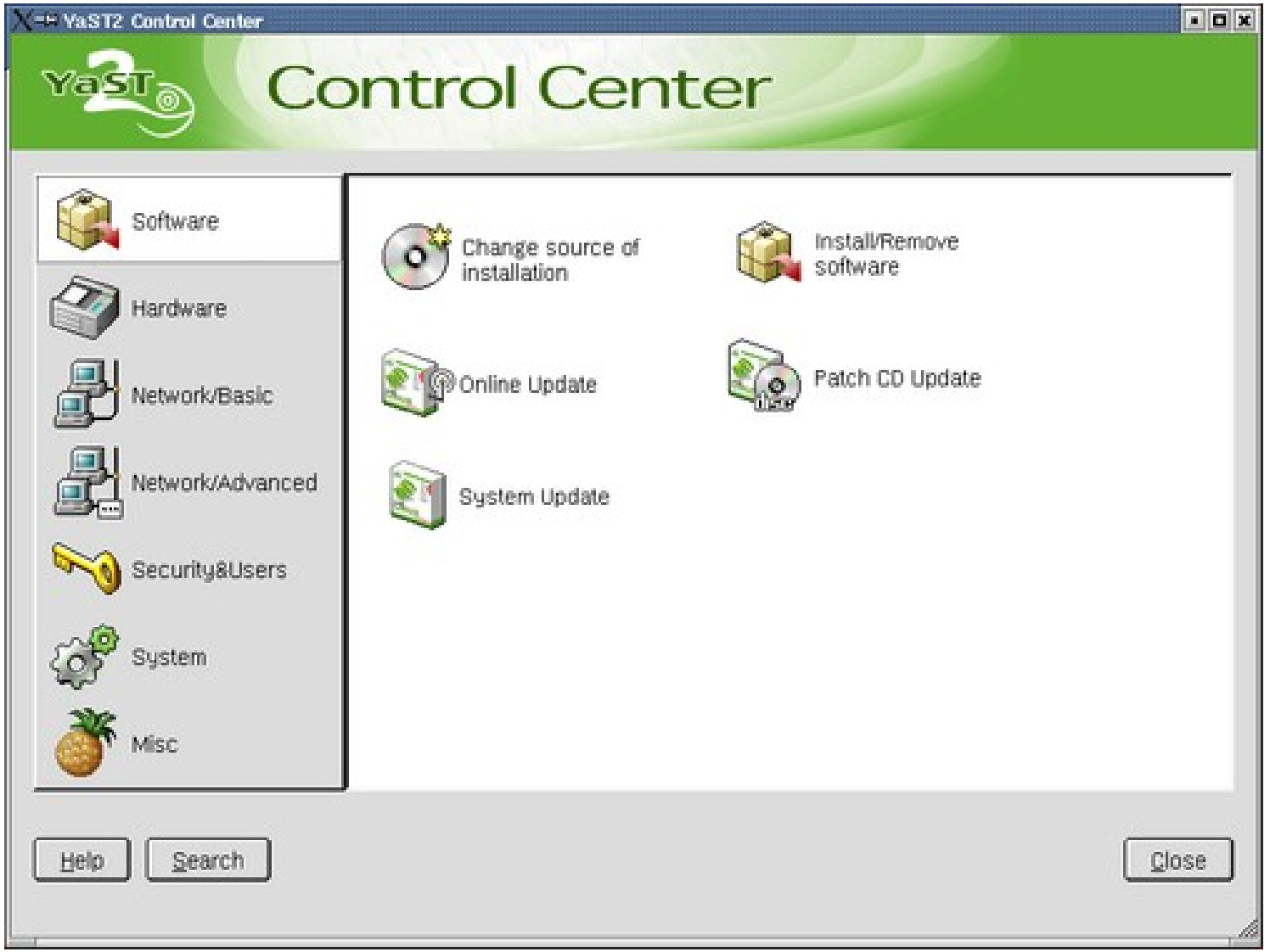
Outils graphiques d'administration

Webmin pour Gnome



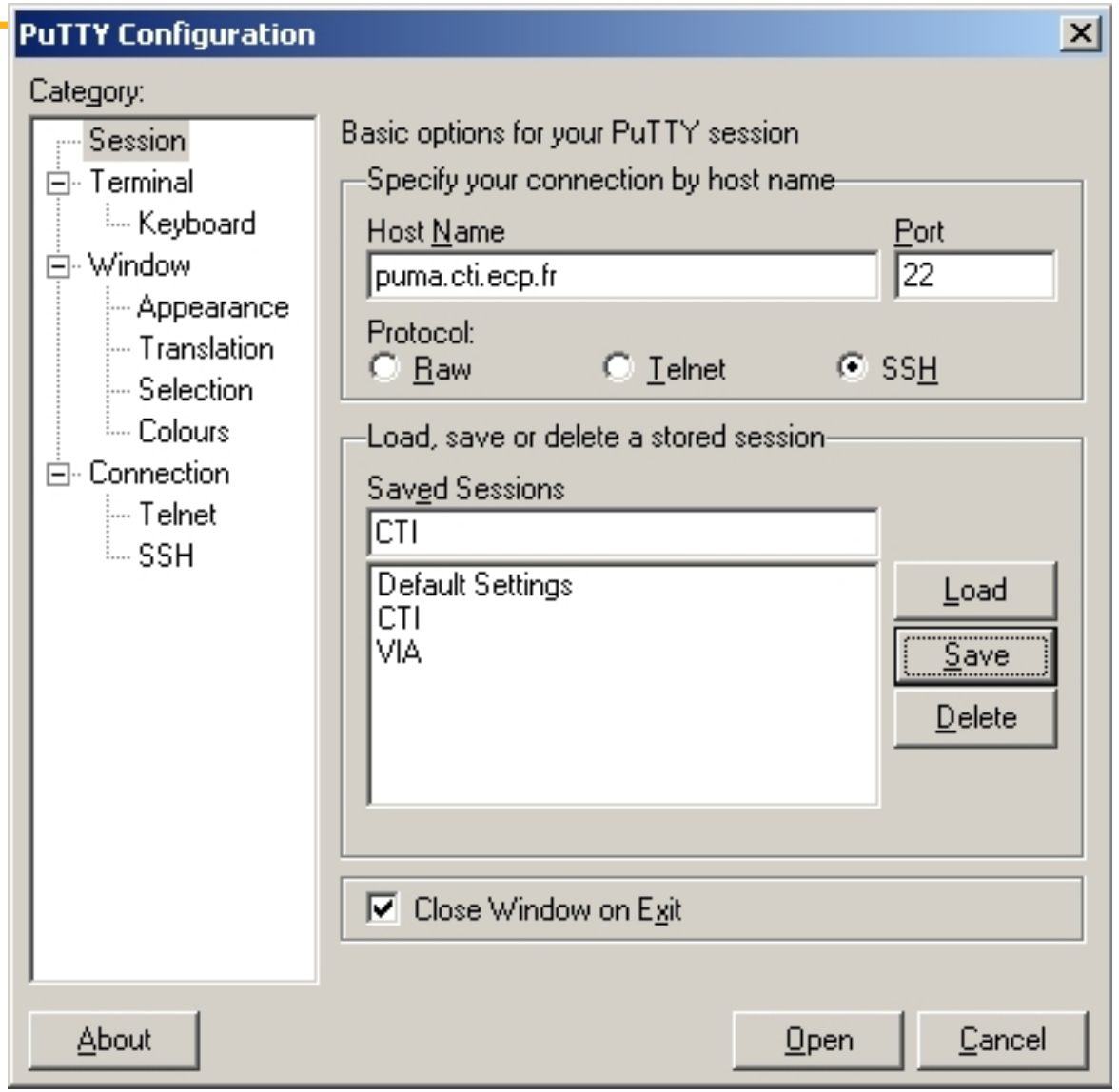
Outils graphiques d'administration

Yast pour KDE



Outils Windows : PuTTy

- Un client Telnet et SSH opensource pour Windows.
- SSH = cryptage des mots de passe

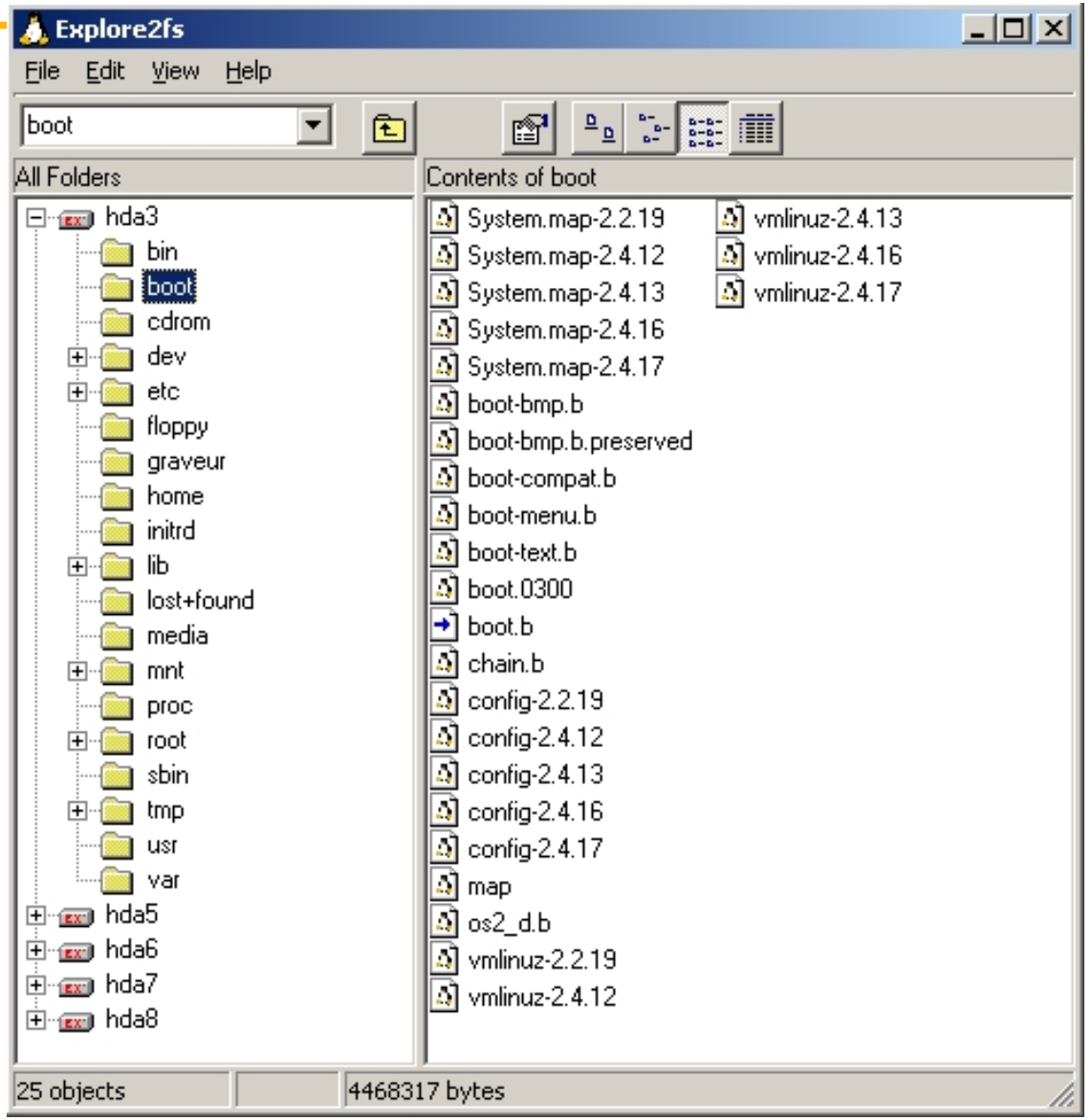


Outils Windows : Explore2fs

http://www.chrysocome.net/explore2fs

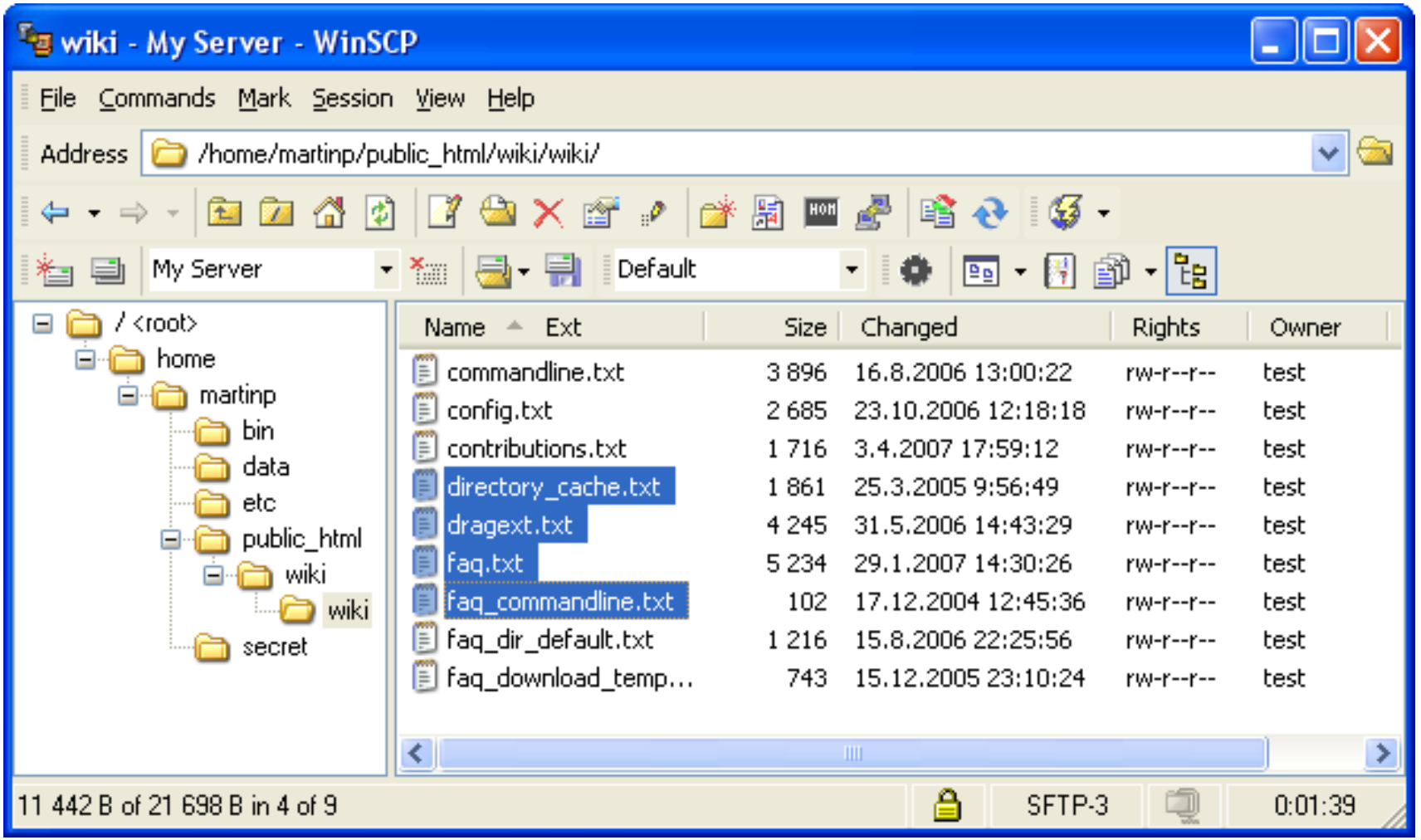
Un programme opensource qui permet de lire ses partitions Linux depuis Windows.

Attention, pour se servir de ce programme sous Windows XP, il faut avoir les privilèges d'administrateur



Outils Windows : WinSCP

Programme permettant de transférer des fichiers en utilisant un serveur ssh



Plan

- Introduction
- Administration système : Linux
  - Détail des taches
  - Distributions
  - Installation et démarrage
  - Fonctionnement et gestion des paquets
  - Modification de la configuration
  - **Noyau et modules**
- Administration de services réseau :
  - DHCP, NFS, NIS, Samba, DNS, Postfix

Noyau

Cœur de l'OS = interface entre programmes et matériel

- Gestion des processus, de la mémoire des droits…
- Noyau sous licence GPL (General Public Licence) = on le retrouve dans toutes les distributions Linux

Pourquoi recompiler ?

- But : OS le plus rapide possible
- Personnaliser le noyau pour qu'il ne supporte que nos périphériques (il sera plus petit)
- Ajouter des modules pour supporter certaines fonctionnalités spéciales (USB…)

Connaître la version courante : `uname -rs`

Noyau

Noyau = programme exécutable lancé au démarrage et gérant les processus, la mémoire…

Comme tous les programmes il provient de la compilation de fichiers .c et .h

Étapes pour créer un nouveau noyau

- Ramener les sources : fichier .h, .c et Makefiles
- Configurer la compilation : choisir les éléments devant être supportés par le noyau
- Compiler
- Installer le nouveau noyau (programme issu de la compilation) afin de pouvoir démarrer dessus (on peut également garder l'ancien)

Ramener les sources

Choisir la dernière version

- http://www.kernel.org/

  En septembre 2008, 2.6.26.3

Pour ramener les sources : www.kernel.org

Pour appliquer un patch au noyau avant de le recomplier, taper :

```
cd /usr/scr
gunzip -c fichier.patch | patch -p0
```

Il est préférable d'appliquer les patchs un à un (patch1, patch2...), car après patch1 vient patch10 et non patch2 si on introduit une commande du type patch*.

Pour vérifier que tout s'est bien passé, entrer :

```
find /usr/src/linux -follow -name ``*.rej'' -print
find /usr/src/linux -follow -name ``*#'' -print
```

pour trouver ceux qui ont été rejetés.

Noyau -> Méthode classique

```
# cd /usr/src
# tar xfvz linux-whatever.tar.gz
# rm -rf linux
# ln -s linux-whatever linux
# cd linux
# make menuconfig
... configurez ...
# make dep
# make bzImage
... editions des fichiers de configuration pour / grub ...
... déplacez /usr/src/linux/arch/i386/boot/bzImage vers boot ...
# make modules; make modules_install
   ... ajoutez les noms des modules dont vous avez besoin dans
  /etc/modules
# shutdown -r now
... redémarrez avec le nouveau noyau ...
```

Noyau -> Méthode classique

Outils nécessaires :

gcc, ld et make on vérifie via la commande:

- dpkg -l | egrep -e "gcc|make|binutils"

Pour télécharger le noyau :

- wget -c http://www.kernel.org/pub/linux/kernel/v2.6/linux-2.6.26.tar.bz2

Noyau -> Méthode Debian

Outils nécessaires :

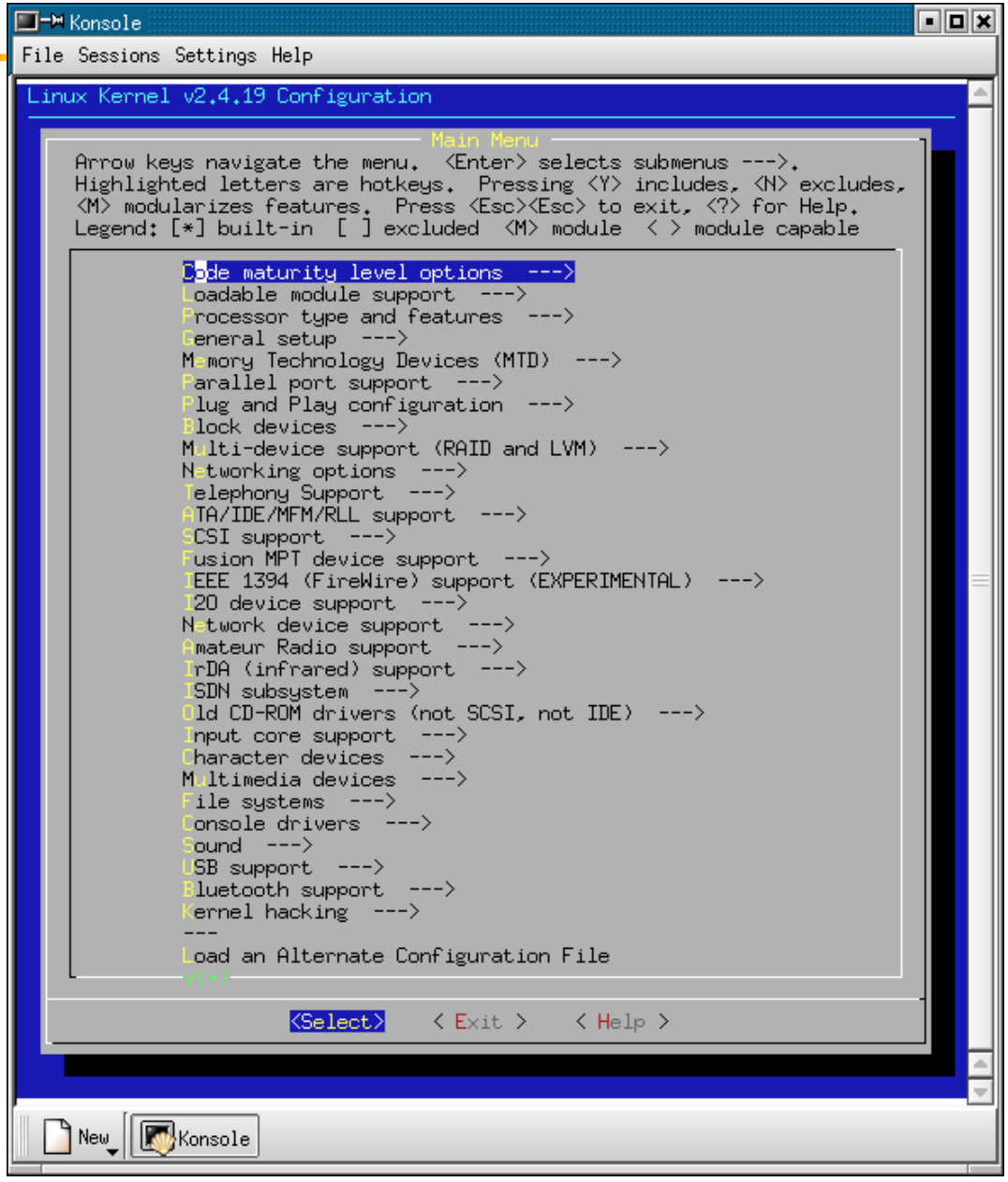
- Paquet kernel-package :
  - Apt-get install kernel-package
- On décompresse le noyau dans le répertoire /usr/src et on fait le lien symbolique
- On peut récupérer l'ancienne configuration :
  - Cp /boot/config.2.24-1-686 /usr/src/linux
  - On configure le noyau :
  - Make menuconfig, make xconfig
  -
- On lance la compilation :
  - make-kpkg clean
  - make-kpkg kernel_image
- On installe comme un paquet :
  - dpkg -i kernel-image-2.x.x.xxx_i386.deb

Configuration du noyau

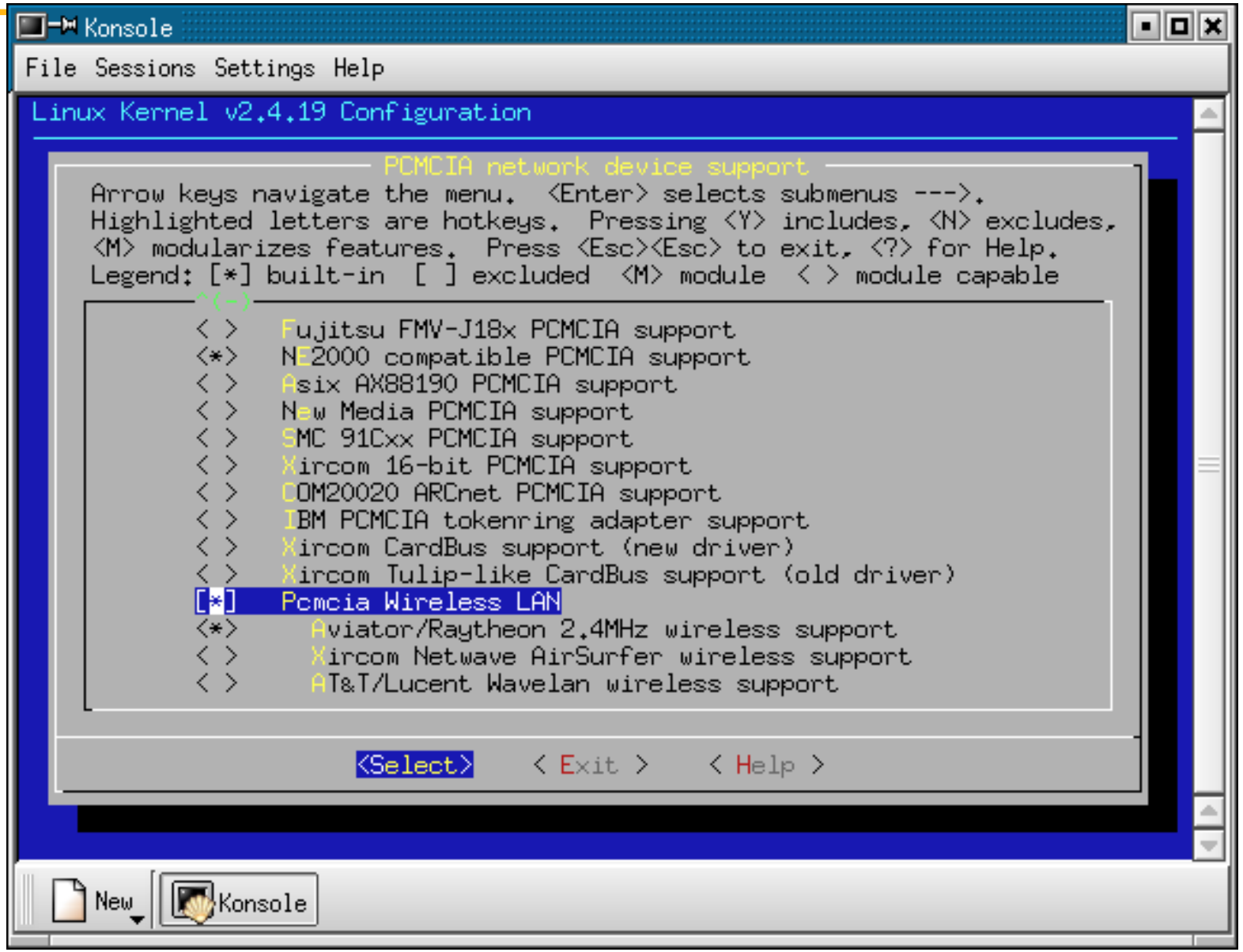
make menuconfig

Choix

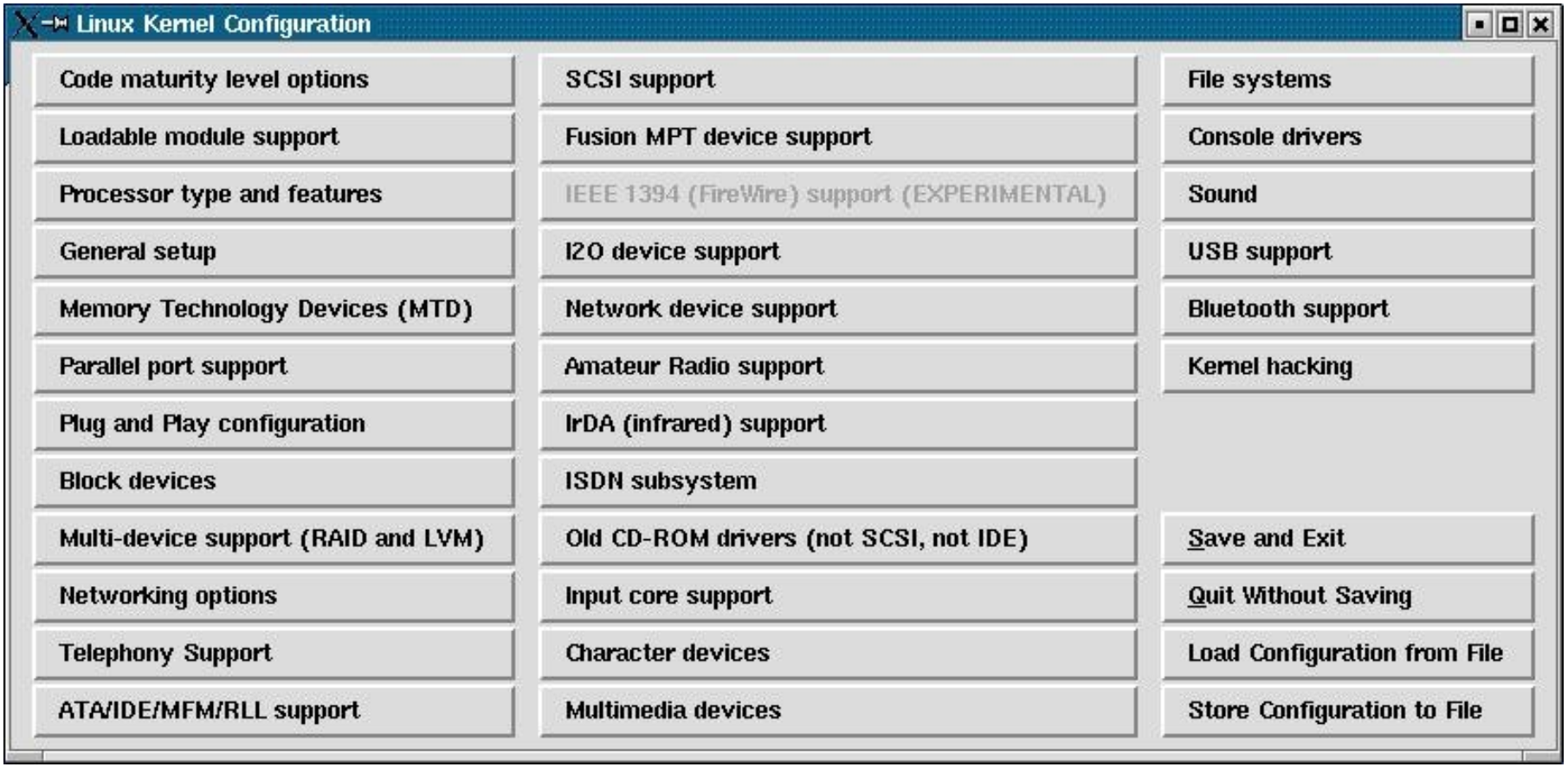
- Option en dur [*]
- Option en modules pouvant être ajoutés ou enlevés sans devoir rebooter [M]
- Attention à la taille du noyau
- Attention certaines fonctionnalités doivent impérativement être en dur



Configuration du noyau :exemple



Make xconfig



Configuration du noyau : menu (1)

Code maturity level options ---> Ce menu ne contient qu'une seule option qui, si elle est cochée, fait apparaître les options considérées comme "instables". Ces options apparaissent alors avec un flag [EXPERIMENTAL] ou même [DANGEROUS].

Loadable module support ---> C'est là que tu lui dis que ton noyau doit être capable d'insérer et d'enlever des modules à la volée sans rebooter. Il faut donc activer au minimum les options "Enable loadable module support" et "Kernel module loader".

Processor type and features ---> Dans ce menu, tu commences par définir ton type de processeur. Ensuite, il faut activer "MTRR (Memory Type Range Register) support" et, si tu as la chance d'avoir une machine multi-processeur, il faut aussi activer "Symmetric multi-processing support"….

Configuration du noyau : menu (2)

General setup ---> Dans tous les cas, tu auras besoin des options "System V IPC", "BSD Process Accounting", "Sysctl support" et "Kernel support for ELF binaries". Mets toutes ces options en dur. En plus, il faut activer, si tu en as besoin :

- le support du bus PCI : coche "PCI support" et "PCI device name database". Pour "PCI access mode" mets "Any".
- le support du bus ISA : coche "EISA support".
- le support du bus PCMCIA : coche "Support for hot-pluggable devices" et va dans le sous-menu "PCMCIA/CardBus support" qui apparaît. Dans ce sous-menu, mets en dur "PCMCIA/CardBus support", "CardBus support", "i82092 compatible bridge support" et "i82365 compatible bridge support".
- le support du Power Management : mets en dur "Power Management support", "Advanced Power Management BIOS support" et "Enable PM at boot time".

Configuration du noyau : menu (3)

Memory Technology Devices (MTD)
Parallel port support
Plug and Play configuration
Block devices
Multi-device support (RAID and LVM)
Networking options
Telephony Support
ATA/IDE/MFM/RLL support
SCSI support
Fusion MPT device support
I2O device support

Network device support
Amateur Radio support
IrDA (infrared) support
ISDN subsystem
Old CD-ROM drivers
Input core support
Character devices
Multimedia devices
File systems
Console drivers
Sound
USB support
Bluetooth support

Compiler le noyau

Commandes pour la compilation du noyau à partir de la nouvelle configuration

`make dep` : pour les dépendances

`make clean`

`make bzImage` : compiler le noyau en lui même

`make modules` : compiler les modules

`make modules_install` : installer les modules (/lib/modules)

Les makes (1)

`make bzImage`

Compile le noyau (fichier bzImage) et le place dans `arch/i386/boot`

`make bzdisk`

Même chose et place aussi l'image sur la disquette

Le préfixe bz signifie que le noyau est compressé : il sera décompressé à son exécution

`make zImage`

Attention avec les noyaux actuels l'image du noyau risque d'être trop importante (moins de compression)

Les makes (2)

```
make mrproper
```

Nettoyage extensif. Il est conseillé de le faire au début à chaque patch.
Attention : sauvegarder votre ancienne configuration (.config)

```
make oldconfig
```

Configure à partir d'un ancien fichier de configuration

Installer le noyau : grub (1)

- On déplace le nouveau noyau dans le répertoire /boot :

```
cp /usr/src/linux-2.6.25/arch/i386/boot/bzImage
/boot/vmlinuz-2.6.26
```

- Il faut aussi bouger la map du system :

```
cp /usr/src/linux-2.6.26/System.map /boot/System.map-2.6.26
rm /boot/System.map
ln -s /boot/System.map-2.6.26 /boot/System.map
```

- On va aussi sauvegarder dans /boot le fichier de configuration du noyau

```
cp /usr/src/linux-2.6.26/.config /boot/config-2.6.26
```

Installer le noyau : grub (2)

- Editer /boot/grub/menu.lst et rajouter le nouveau noyau
-

Les modules

- /lib/modules
  - Commandes `modprobe` et `insmod` pour installer
  - /lib/modules/net = drivers des cartes réseau
  - make modules ; make modules_install

Les modules commandes

`lsmod :`

liste les modules présents en mémoire.

`insmod /lib/modules/2.6.25/net/3c509.o`

Permet de charger manuellement un module.

`rmmod 3c509 :`

Permet de décharger manuellement un module .

`modprobe`

Permet de charger un module et toutes les dépendances qui lui sont associées.

Les modules : boot

Pour qu'un module soit lancé au boot

Le placer dans /lib/modules/…
Ajouter son nom dans /etc/modules

```
# /etc/modules: kernel modules to load at boot time.
#
# This file should contain the names of kernel modules that are
# to be loaded at boot time, one per line.  Comments begin with
# a #, and everything on the line after them are ignored.

3c59x
```

Le module : NTFS Support

- Pour pouvoir monter une partition NTFS
  - make menuconfig : vérifier que l'option est activée soit en module soit en dur (généralement en module)
  - File Systems / NTFS file system support (read only)
  - File Systems / NTFS write support DANGEROUS
- En module ou en dur
  - En module :
    - ajouter ntfs à /etc/modules
    - ou insmod /lib/modules/fs/ntfs/ntfs.o
  - En dur : ça fonctionne mais attention à la taille du noyau
- Enfin : mount –t ntfs /dev/hda1 /windows

Ramener un noyau précompilé

- Image précompilée du noyau

- Mise à jour de la liste

```
apt-get update
```

- Obtenir le nom de la dernière version

```
apt-cache search kernel-image
```

- Installer le noyau et configurer le boot (exécution de lilo)

```
apt-get install kernel-image-2.6.24
```

Plan

- Introduction
- Administration système : Linux
  - Détail des taches
  - Distributions
  - Installation et démarrage
  - Fonctionnement et gestion des paquets
  - Modification de la configuration
  - Noyau et modules
- **Administration de services réseau** :
  - DHCP, NFS, NIS, Samba, DNS, Postfix

Client – Serveur

- Programme client-serveur :
  - Serveur : programme en attente de connexions et fournisseur de services
  - Client : Connexion puis demande de service au serveur
- Programmation à l'aide de sockets :
  - On crée du côté serveur une socket attendant les demandes de connexion sur un port donné
  - Du côté client une socket demandant une connexion sur un port d'un machine donnée
  - Sockets respectant des normes : communications possibles entre Windows et Linux

Serveur

- Programme tournant sur une machine et attendant les connexion sur un port particulier
  - Port spécifié lors de la création de la socket serveur (ex : 4444)
  - Certains ports sont réservés à des protocoles standards (22 : SSH, 23 : telnet…)
- Protocole de communication :
  - Le serveur envoie des données sous un format précis (byte, caractères…)
  - Le client doit parler le même langage

Programme serveur

```
public class Server1{

  public static void main(String[] args) {
      ObjectInputStream in = null;
      ObjectOutputStream out = null;
      Socket clientSocket;

      try {
      ServerSocket server = new ServerSocket(4444);
      clientSocket =  server.accept();
      out = new ObjectOutputStream(clientSocket.getOutputStream());
      in = new ObjectInputStream(clientSocket.getInputStream());

      for(;;){
      out.writeObject("bonjour du serveur");
      System.out.println(in.readObject());
       }
      }
```

Client

- Programme demandant la connexion
  - Il doit connaître le port du serveur correspondant
  - Il doit parler le même langage que le serveur (format des données émises)
- Conditions
  - Si un programme serveur ne tourne pas sur le port spécifié le client essuie un refus
  - Le serveur peut être paramétré pour accepter des connexions de certaines IP
  - Les proxy, routeurs… peuvent filtrer certain ports

Programme Client

```
public class ExempleSocket1 {

public static void main(String args[]) {
    if (args.length != 2)
            System.out.println("usage : ExempleSocket1 hote port");
    else {
        Socket sk = null;
        try {
            sk = new
    Socket(args[0],Integer.valueOf(args[1]).intValue());
            DataInputStream is = new
    DataInputStream(sk.getInputStream());
             String ligne;
             while ((ligne = is.readLine()) !=  null)
                 System.out.println(ligne);
            }
```

Utilisation du client

- Essai de lecture sur les ports 22 et 23
- Aucun serveur telnet (port 23) ne tourne sur la machine lifc
- Le serveur SSH de lifc envoie la chaîne de caractère « SSH-1.99-OpenSSH_3.4p1 » et attend les commandes du client

```
C:\FormationJava\FormationJava>java ExempleSocket1 lifc 23
erreur entree/sortie : Connection refused: connect

C:\FormationJava\FormationJava>java ExempleSocket1 lifc 22
SSH-1.99-OpenSSH_3.4p1
```

Utilisation des ports

- Fichier /etc/services

```
# /etc/services:
tcpmux          1/tcp                   # TCP port service multiplexer
echo            7/tcp
echo            7/udp
. . .
ftp-data        20/tcp
ftp             21/tcp
fsp             21/udp          fspd
ssh             22/tcp                  # SSH Remote Login Protocol
ssh             22/udp                  # SSH Remote Login Protocol
telnet          23/tcp
# 24 - private
smtp            25/tcp          mail
```

Surveillance des ports

- Commande ***nmap***
  - `nmap localhost` : permet de voir la liste des ports ouverts = scannage des ports
  - Un scannage n'est possible que sur la machine dont on est root
  - Introduction de pirates sur un système possible uniquement par des ports ouverts
- Commande ***netstat -p***
  - Liste des connexions ouvertes à un instant donné sur une machine

Sécurité Debian

- Dernières informations de sécurité Debian
  - abonnez-vous à la liste de diffusion **debian-security-announce**
- Apt pour récupérer les mises à jour relatives à la sécurité.
  - Ajouter la ligne suivante à /etc/apt/sources.list

```
deb http://security.debian.org stable/updates main contrib non-free
```

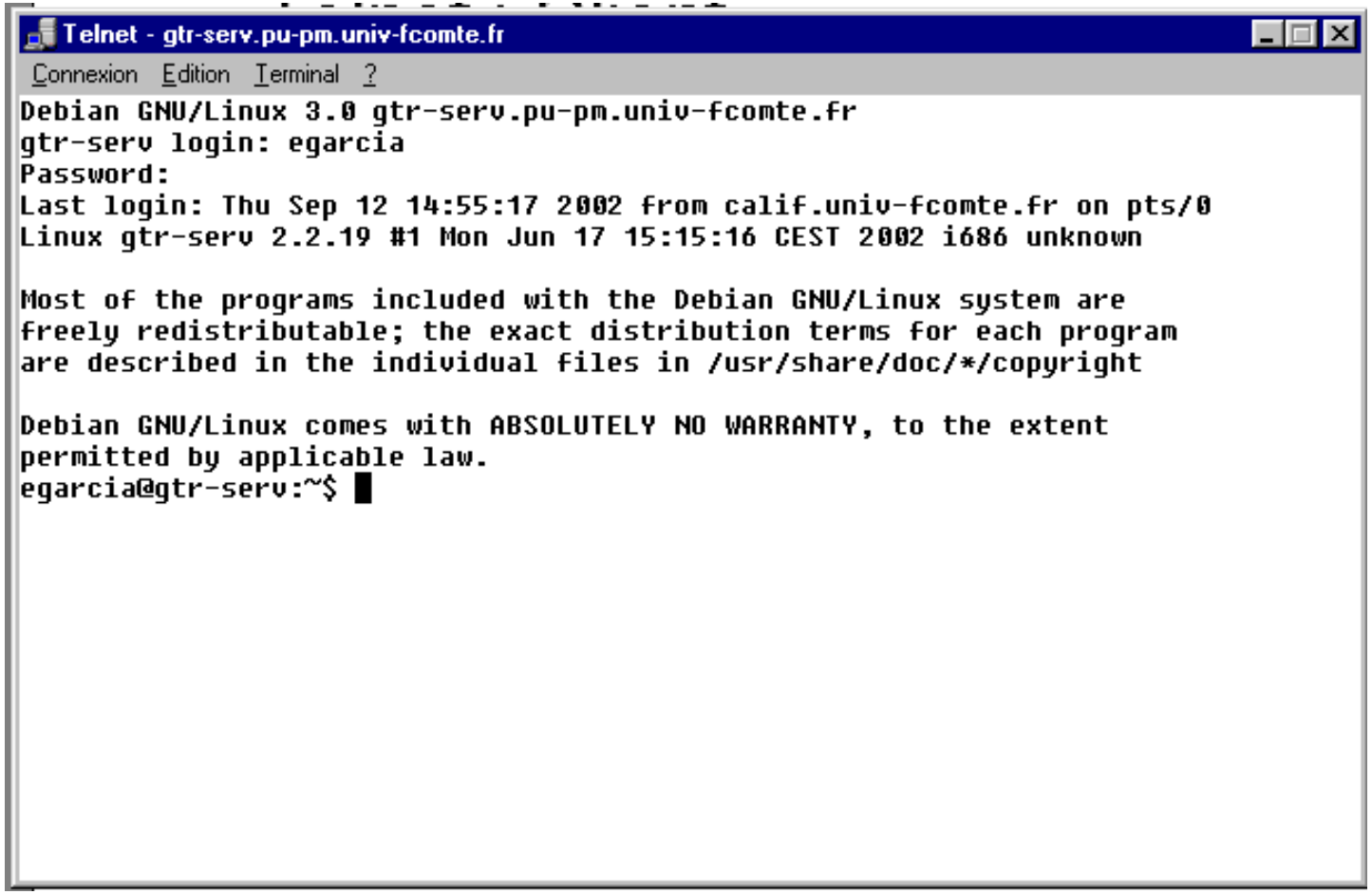
  - Puis `apt-get update` et `apt-get upgrade`

Sécurité Debian

- Exemple de trou de sécurité : le « . » Dans le PATH
- Scénario :
  - Un pirate peut copier un script nommé « ls » dans votre répertoire /tmp
  - « ls » peut être un script malveillant
  - En tant que root vous vous placez dans /tmp et tapez « ls » pour lister les fichiers
  - C'est le script « ls » qui est exécuté avec les droits administrateur
  - Si le « . » n'est pas dans le PATH : ./ls pour exécuter ce srcipt

Telnet : Client

- Connexion sur un serveur telnet
  - Login + password = commandes



Telnet : Serveur

- Une connexion telnet vers une machine n'est possible que si cette machine possède un serveur telnet
  - Programme écoutant sur le port 23 et utilisant un format de communication = client telnet
  - Exemple : programme utilisant des sockets (serveur écoute sur un port) et client se connecte et envoie un message d'un format particulier (ajoute = ajouter un user dans un annuaire…)
- Fichier de configuration
  - /etc/inetd.conf

SSH : version sécurisée

- Secure Shell : encryption du mot de passe
  - telnet envoie les mots de passe en clair sur le réseau
- Fichiers de configurations
  - /etc/ssh/ssh_config pour le client
  - /etc/ssh/sshd_config pour le serveur
  - Port, PermitRootLogin, X11forwarding…
- Démon : /etc/init.d/ssh
- Commandes :
  - ssh eric@gtr-serv.pu-pm.univ-fcomte.fr : connexion
  - scp test.txt eric@gtr-serv.pu-pm.univ-fcomte.fr : transfert

SSH : X11Forwarding

- Installation
  - apt-get install ssh
- Edition de /etc/ssh/sshd_config côté serveur
  - X11Forwarding yes
- Edition de /etc/ssh/ssh_config côté client (condition A)
  - ForwardX11 yes (pour certains hôtes ou pour tous ***host ****)
- Lancement
  - /etc/init.d/ssh restart
  - Puis depuis un client : ssh toto@serveur si condition A ou ssh –X toto@serveur sinon

Transfert de fichiers

- File Transfer Protocol (port 21)
- Commandes :
  - cd, ls : distants
  - lcd : local
  - put, get : prendre ou déposer un fichier
- Exemple de serveur : ProFTPd
  - apt-get install ProFTPd
  - Fichier de configuration : /etc/proftpd.conf
- Option : serveur anonyme = tout le monde peut se connecter

ProFTPd : fichier de configuration

- Fichier proftpd.conf
  - Port 21
  - DisplayLogin welcome.msg
  - TimeoutNoTransfer 600
- Option : serveur anonyme = tout le monde peut se connecter dans la limite d'un certain répertoire
  - Décommenter ***Anonymous ftp***

Serveur FTP : anonymous

- Décommenter la fin du fichier /etc/proftpd.conf
- Il faut qu'un compte et un groupe ***ftp*** existent
  - adduser ftp
  - /etc/passwd : * pour le mot de passes pour invalider le compte afin qu'il ne soit pas utilisable en telnet
- Parametrage pour limiter les accès en lecture ou en lecture écriture
  - ???
- Pour se connecter : login `anonymous`, passwd `anonymous`

Fichier /etc/ftpusers

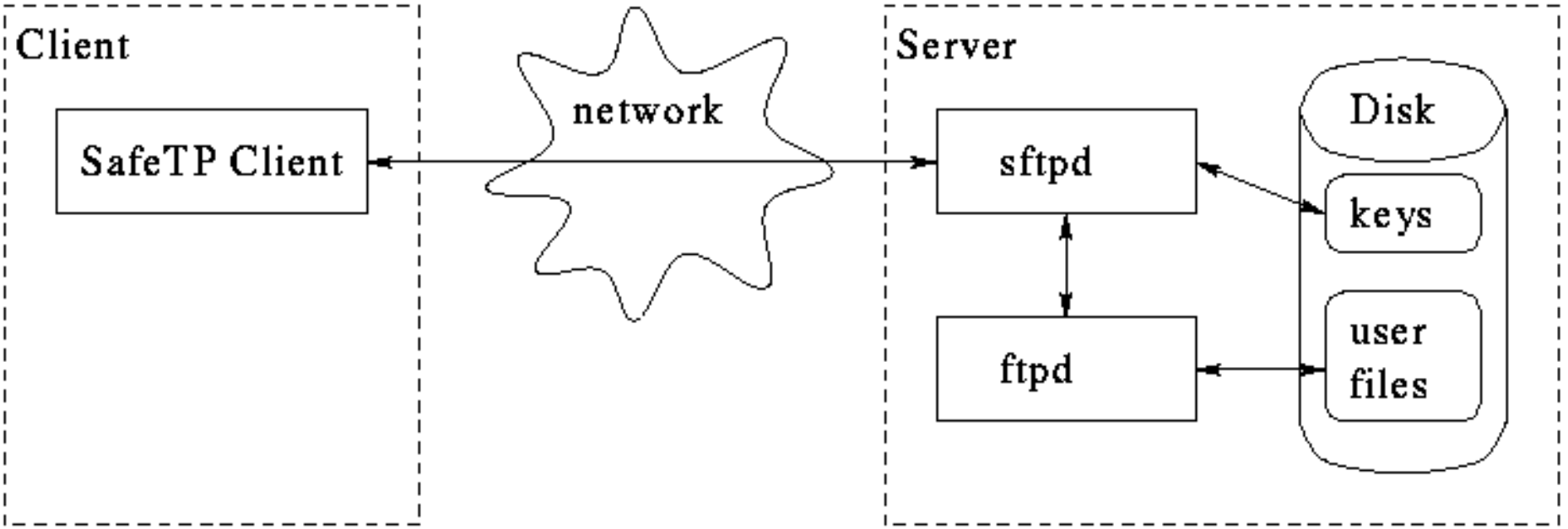
- Le fichier /etc/ftpusers permet d'interdire l'accès en FTP à la machine à certains utilisateurs
  - Ajouter un ID et relancer le serveur
  - Tous les utilisateurs présents dans ce fichier sont refusés au login ftp

```
# /etc/ftpusers: list of users disallowed FTP access. See
ftpusers(5).

root
daemon
bin
sys
sync
games
man
lp
mail
news
uucp
nobody
```

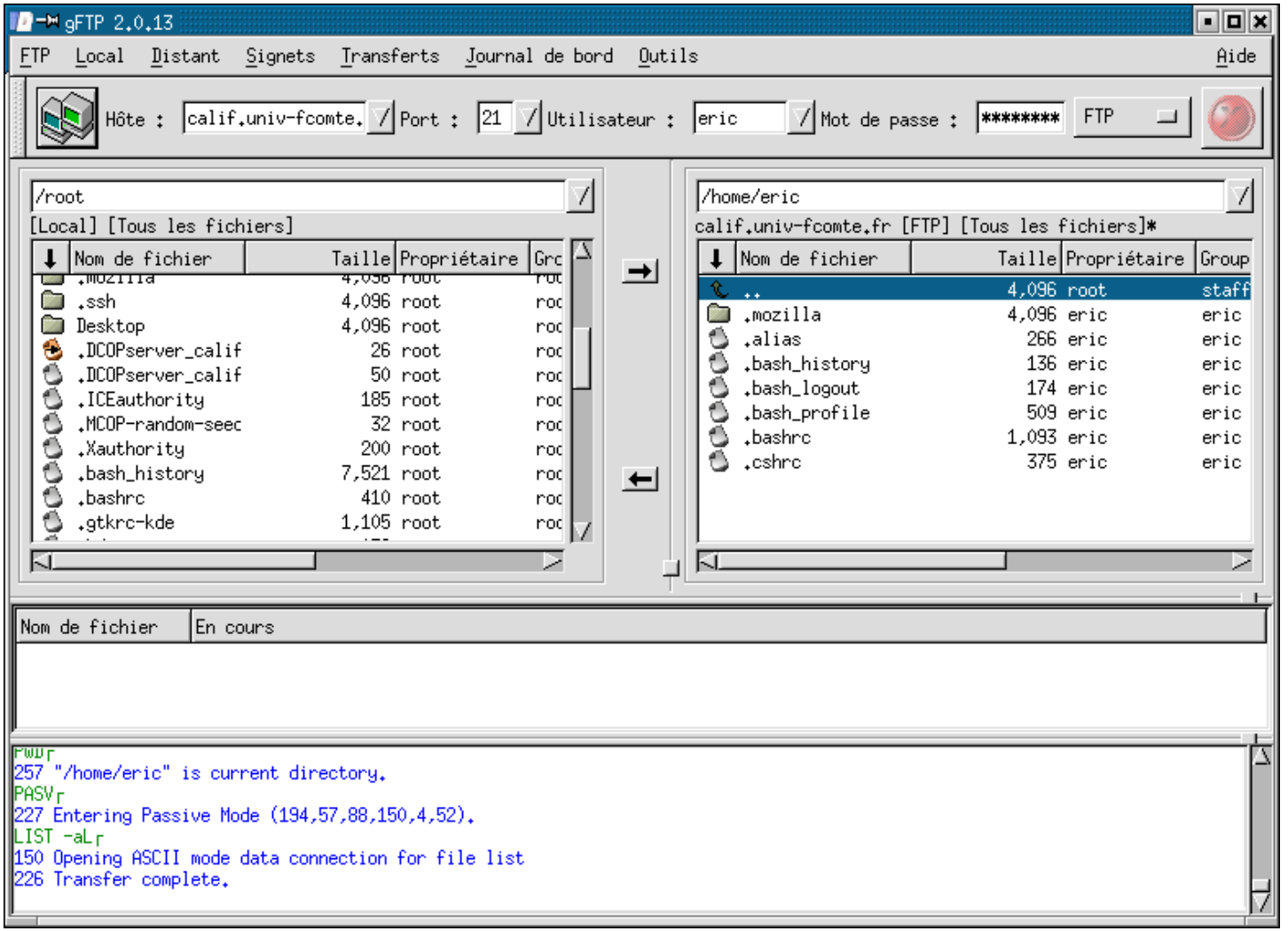
Versions

- Différentes versions
  - ftp, lftp
  - sftp utilise SSH
  - Serveurs : sftpd, ftpd, proftpd, wu-ftpd…
  - Activation ou désactivation dans /etc/inetd.conf



Interface graphique

- Versions graphiques
  - gftp : X gtk



Plan

- Introduction
- Administration système : Linux
  - Détail des taches
  - Distributions
  - Installation et démarrage
  - Fonctionnement et gestion des paquets
  - Modification de la configuration
  - Noyau et modules
- Administration de services réseau :
  - **DHCP**, NFS, NIS, Samba, DNS, Postfix

Installation

- DHCP (Dynamic Host Configuration Protocol)
  - Démon client : configuration automatique des hôtes d'un réseau*
  - Démons serveur : administration de la distribution des adresses
- Installation du serveur DHCP : dhcpd
- Installation du client : dhclient
- Debian paquet :
  - `dhcp3-server` : serveur
  - `dhcp3-client` : client

Configuration : serveur ouvert

- Fichier de configuration du serveur : `/etc/dhcpd.conf`
- Relancer le serveur DHCP : `/etc/initd/dhcp restart`

Fichier /etc/dhcpd.conf

```
option domain-name "univ-pau.fr";
option subnet-mask 255.255.255.0;
option domain-name-servers 194.57.91.200;
default-lease-time 86400;
max-lease-time 604800;

subnet 194.57.88.0 netmask 255.255.255.0 {
        option broadcast-address 194.57.88.255;
        option routers 194.57.88.254;
        range 194.57.88.50 194.57.88.150
}
```

Configuration : serveur ouvert

- Le serveur envoi à tous les clients :
  - le nom de domaine
  - le masque de sous réseau
  - le ou les numéros de DNS
  - Et leur attribue un n° IP pour 86400 s (24 h) et maxi 604800 s
- Le serveur distribue aléatoirement aux client du réseau 194.57.88.0 les adresses IP
  - Entre 194.57.88.50 et 194.57.88.150
  - Avec l'adresse de broadcast 194.57.88.255
  - Et l'adresse de passerelle 194.57.88.254

Connexions des clients

- Lorsqu'un client demande une adresse IP au serveur
  - Un bloc est ajouté au fichier `/etc/dhcp.leases` ou `/var/lib/dhcp/dhcpd.leases` ou `/var/dhcp/dhcpd.leases` selon les versions de Linux

```
lease 194.57.88.50
        starts 2 2002/09/21 06:28:48;
        ends 3 2002/08/22 06:28:48;
        hardware ethernet 08:00:46:26:29:E7;
        uid 01:08:00:46:26:29:E7;
        client-hostname « calif »
        }
```

Configuration : sécurisation

- Ne donner des IP qu'à des machines connues
  - Utilisation du fichier `/etc/dhcp.leases`
  - Demande des adresses MAC des carte des machines voulant une IP

Fichier /etc/dhcpd.conf

```
option domain-name "univ-pau.fr";
...
subnet 194.57.88.0 netmask 255.255.255.0 {
        option broadcast-address 194.57.88.255;
        option routers 194.57.88.254;
        }
host zorro {
  hardware ethernet 08:00:46:26:29:E7;
  fixed-address 194.57.88.65;
}...
```

Configuration des clients

- Les clients doivent avoir un client DHCP qui tourne
  - dhclient
- Configuration de leur carte réseau :
  - Fichier `/etc/network/interfaces`

```
more /etc/network/interfaces
# /etc/network/interfaces -- configuration file for ifup(8), ifdown(8)
auto eth0
iface eth0 inet dhcp
```

Test de fonctionnement

- Commande /usr/sbin/dhcpd –d –f sur le serveur
- Tentative de connexion par un client :

```
> dhclient
Internet Systems Consortium DHCP Client V3.0.6
Copyright 2004-2007 Internet Systems Consortium.
All rights reserved.
For info, please visit http://www.isc.org/sw/dhcp/

Listening on LPF/eth0/00:1e:0b:57:36:8a
Sending on   LPF/eth0/00:1e:0b:57:36:8a
Listening on LPF/vnet0/b2:52:04:aa:26:5a
Sending on   LPF/vnet0/b2:52:04:aa:26:5a
Sending on   Socket/fallback
DHCPDISCOVER on eth0 to 255.255.255.255 port 67 interval 6
DHCPOFFER of 10.0.50.33 from 10.0.50.1
DHCPREQUEST of 10.0.50.33 on eth0 to 255.255.255.255 port 67
DHCPACK of 10.0.50.33 from 10.0.50.1
 * Reloading /etc/samba/smb.conf smbd only
```

Client Windows

- Pour spécifier un serveur WINS
  - `option netbios-name-server 194.57.88.1`
- Problèmes avec les clients Windows
  - Le serveur doit pouvoir envoyer des paquets vers 255.255.255.255 (pas tjs possible avec différentes versions de Linux)

SI

> route add -host 255.255.255.255 dev eth0 **donne**

> 255.255.255.255: Unknown host

Faire

Ajouter 255.255.255.255 tout-le-monde à /etc/hosts

Puis essayer route add 255.255.255.255 dev eth0

Plan

- Introduction
- Administration système : Linux
  - Détail des taches
  - Distributions
  - Installation et démarrage
  - Fonctionnement et gestion des paquets
  - Modification de la configuration
  - Noyau et modules
- Administration de services réseau :
  - DHCP, **NFS**, NIS, Samba, DNS, Postfix

Introduction

- Network File System :
  - Partage de données entre plusieurs machines
  - exemple : /home de vos comptes
- Protocole peu performant
  - Bon pour des réseaux locaux,
  - Très inconfortable pour les liaisons modem
- Attention aux problèmes de sécurité
  - Mesures indispensables

NFS et les RPC

- NFS repose sur les RPC (Remote Procedure Calls)
  - Utilisation du portmapper (programme `portmap` de Linux)
  - Portmapper = conversion des n° de prog RPC en n° de ports
- Déroulement d'une RPC
  - Serveur RPC :
    - Indique à `portmap` le port qu'il utilise et les n° de prog RPC qu'il gère
  - Envoi d'une requête RPC par un client :
    - Il contacte `portmap` du serveur pour connaître le numéro de port du programme souhaité
    - Il envoie les données au port correspondant

Description du protocole NFS

- NFS est composé de 4 protocoles utilisant les RPC
- nfs = programme `nfsd`
  - Authentification + Création, recherche, lecture et écriture de fichiers
- mountd
  - Montage des systèmes exportés (mount et umount)
- nsm (Network Status Monitor) = programme `statd`
  - Surveillance des nœuds du réseau (redémarrages…)
- nlm (Network Lock Manager) = programme `lockd`
  - Section critique (lock les fichiers utilisés)

Installation

- Installation des paquets
  - apt-cache search nfs : nfs-kernel-server, nfs-common
- Lancement des démons
  - Vérifier `portmap` : rpcinfo –p
  - Lancer `mountd` et `nfs` s'ils ne le sont pas

```
rpcinfo -p
   program no_version protocole  no_port
    100000    2   tcp    111  portmapper
    100000    2   udp    111  portmapper
    100005    2   udp    745  mountd
    100005    1   tcp    747  mountd
    100003    2   udp    2049  nfs
    100003    1   tcp    2049  nfs
```

Configuration du serveur

- Fichier `/etc/exports`
  - Chaque ligne contient le répertoire à exporter et la liste des machines autorisées à y accéder
  - Ex : `/home gtrnet01(rw) gtrnet02(ro)…`
- Formulation des noms de clients :
  - Nom de machine : attention à la résolution des noms
  - Wildcards : exemple gtrnet*.pu-pm.univ-fcomte.fr
  - Un netgroup si on utilise NIS : @gtrlinux
  - Une adresse IP
- Les options :
  - rw et ro : le client peut lire et écrire ou lire uniquement
  - Man exports pour avoir la liste complète des options

Redémarrage du serveur

- Pour que les modifications soient prises en compte :
  - `exportfs` : transmet les modifications au serveur
  - Ou `/etc/init.d/nfs-kernel-server restart` qui fait appel à `exportfs`
- Fichiers important dans /var/lib/nfs/
  - rmtab, etab, xtab : utilisés par le noyau pour savoir si un client est autorisé à monter un répertoire NFS

Client

- Pour monter un SF NFS 2 solutions :
- `mount nom_server:/nom_rep /nom_point_de_montage`
- /etc/fstab : ajouter une ligne

```
/gtr-serv:/home        /home  nfs     defaults        0  0
```

- Options de montage (`man nfs(5)`):
  - rsize, wsize : taille des blocs en lecture ou en écriture
  - Soft, hard, hard,intr : type de gestion des pannes du serveur
  - noexec, nosuid : gestion de l'exécution de programme sur le SF NFS

Exemple complet

- gtr-serv veut partager le répertoire /gtr-serv0 avec gtrnet01 pour un espace de sauvegarde.

- Sur gtr-serv

  - Ajouter la ligne suivante à /etc/exports

    ```
    /gtr-serv0    gtrnet01(rw)
    ```

  - Relancer nfs :

    ```
    /etc/init.d/nfs-kernel-server restart
    ```
    ou
    ```
    exportfs
    ```

- Sur gtrnet01 : Monter le répertoire /gtr-serv0

  - `mkdir /gtr-serv0` : on crée le point de montage
  - `mount gtr-serv:/gtr-serv0  /gtr-serv0`

Lenteur de NFS

- NFS = protocole lent
  - Sur-coût en bande passante
  - Plus lent que ftp, http, ssh…
- Utilise les RPC et 4 protocoles
- Selon les options de montage, les problème de contact avec le serveur peuvent bloquer la machine un certain temps
- NFS n'est utilisable que sur un réseau local à débit élevé
  - N'essayez pas avec un modem.

Optimisation de NFS

- Options de montage rsize, wsize = écriture et lecture par blocs de 4096 ou 8192 octets par défaut
- Ce chiffre n'est pas toujours optimal = expérimentation

```
time dd if=/dev/zero of=/mnt/testfile bs=16k count=4096
```

  - Mesure le temps de création d'un fichier de 64 Mo

```
time dd if=/mnt/testfile of=/dev/null bs=16k
```

  - Mesure le temps de lecture du fichier

- Puis umount et mount avec d'autres valeurs pour rsize et wsize (multiples de 1024 et < à 16384)
- Puis on recommence pour trouver les valeurs optimales

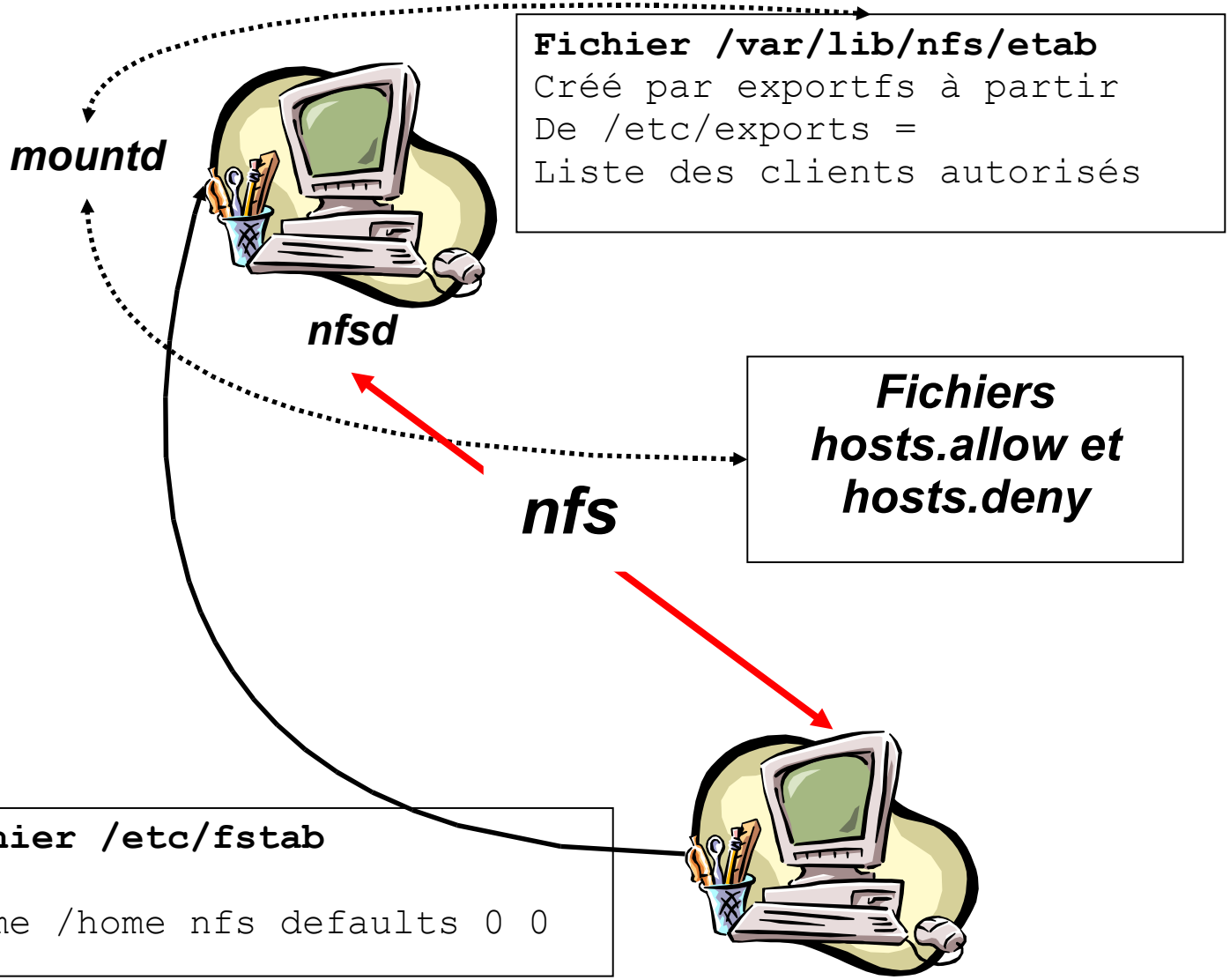
Sécurité de NFS

- Sécurité côté client : ne pas faire confiance au root du serveur
  - option *nosuid* de mount : pas de démarrage de programmes suid depuis le système nfs
- Sécurité côté serveur
  - option *root_squash* dans /etc/exports : transforme l'UID 0 en UID de l'utilisateur *nobody* (active par défaut : vérifier simplement que no_root_squash n'est pas présente)
- Firewalls
  - Routeurs ou firewall : couper les ports 2049 (nfsd), 749 (portmapper), 745 et 747 (mountd)

Sécurité de NFS : portmapper

- Portmapper utilisé par nfsd, mountd, ypbind/ypserv, pcnfsd, commandes « r » comme rusers…
- Fichiers /etc/hosts.allows et /etc/hosts.deny
  - Editer /etc/hosts.deny : `portmap: ALL` (on refuse l'accès à quiconque)
  - Puis /etc/hosts.allow pour rouvrir à quelques machines connues
- Exemple : ajouter toutes les machines d'un réseau
  - /etc/hosts.allow : `portmap: 194.57.88.0/255.255.255.0`
  - ATTENTION : mettre les IP non les noms de machines

Récapitulatif



Erreurs fréquentes

- Messages du type `RPC connection refused`
  - pas de `portmap`
  - ou problèmes avec `hosts.allow` et `hosts.deny`
  - `mountd` et `nfsd` ne tournent pas sur le serveur
  - Attention aux noms dans exports, mettre les n° IP pour éviter les problème de résolutions
  - Ne pas monter un SF avant d'avoir relancer `nfsd` ou `exportfs` sur le serveur après une modification de `/etc/exports`...

Plan

- Introduction
- Administration système : Linux
  - Détail des taches
  - Distributions
  - Installation et démarrage
  - Fonctionnement et gestion des paquets
  - Modification de la configuration
  - Noyau et modules
- Administration de services réseau :
  - DHCP, NFS, **NIS**, Samba, DNS, Postfix

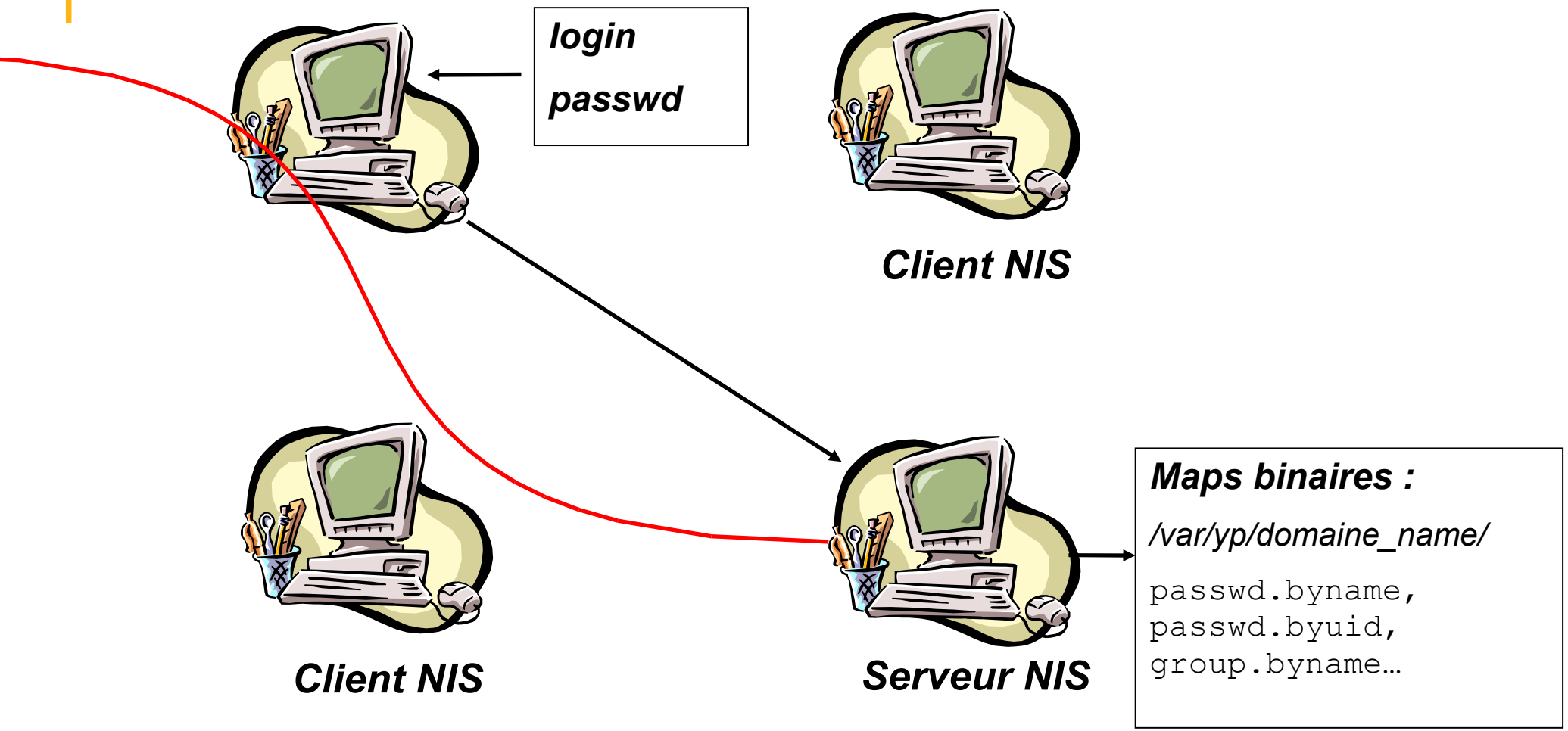
But

- Centraliser les connexions sur un réseau local
  - Se connecter à un serveur de fichier sous un compte centralisé
  - Ne pas définir de compte machine par machine
- Réseau homogène Linux
  - Connexion et authentification grâce au service NIS
  - Accès aux répertoires partagés grâce à NFS
  - Pour utiliser des stations Windows : serveur SAMBA
- Serveur NIS
  - Au moins un par réseau
  - Plusieurs : soit un par domaine NIS soit serveurs coopératifs (un maître et des esclaves)

Fonctionnement

- NIS maintient une base de donnée au format DBM sur un domaine NIS
  - Les informations sont contenue dans le répertoire : `/var/yp/nom_de_domaine` = maps
- Les clients cherchent des infos sur :
  - Résolution d'adresses, login, passwd, groupes…
  - Info habituellement situées dans `/etc/host`, `/etc/passwd`, `/etc/group`, `/etc/shadow`…
- NIS utilise également les RPC : le portmapper doit tourner

Récapitulatif



Les programmes

- Différents programmes :
  - ypbind : côté client, lie le client au serveur NIS
  - ypserv : côté serveur
  - yp-tools : outils NIS, yppasswd, ypcat, ypwhich…
- Debian :
  - `apt-get install nis` : tous les programmes
  - Script `/etc/init.d/nis` : teste si la machine est client NIS ou serveur et lance les programmes correspondants correspondants

```
if [ "$NISSERVER" != "false" ] then
                echo -n "ypserv "
                start-stop-daemon --start --quiet \
                        --pidfile /var/run/ypserv.pid --exec $
{NET}/ypserv
fi
```

Configurer un client

- Besoin des programmes ypbind et yp-tools
- Nom de domaine
  - Commandes `domainname, nisdomainname.`
- Fichier de configuration pour `ypbind` : `/etc/yp.conf`
  - Ajouter les lignes :

```
4. domain nom_de_domaine
5. ypserver toto.univ-fcomte.fr
```

  - Ces deux lignes sont optionnelles : le serveur NIS peut être trouvé par broadcast

nsswitch.conf

- Le fichier `/etc/nsswitch.conf`
  - Détermine l'ordre des recherches

```
hosts:          files dns nis
networks:             nis files ...
```

  - Pour les options passwd, group et shadow : 2 solutions
  - `passwsd :                files nis` ou
  - `Passwd :          compat`

Client NIS : mode ***compat***

- Pour pouvoir gérer les accès à NIS
- Exemple : Ajouter +:::::: à la fin de /etc/passwd chez les clients
  - \+ pour inclure des utilisateurs NIS sur une machine
  - \- pour exclure
- Lancer ypbind
  - /etc/init.d/nis start pour Debian
  - /etc/init.d/ypbind start pour Redhat

yptools

- Toutes les commandes yp : questionnement du serveur NIS sur ses tables ou maps
  - `ypwhich` : questionnement du serveur sur son nom
  - `yppasswd` : envoi au serveur d'une demande de changement de mot de passe
  - `ypcat` : questionnement du serveur sur ses tables…

```
calif:~# ypcat passwd.byname (liste des users NIS)
eric:x:1000:1000:Eric Garcia,421,,:/home/eric:/bin/bash
moi:x:1003:1003:,,,:/home/moi:/bin/bash
toto:x:1001:1000:,,,:/home/toto:/bin/bash
nobody:x:65534:65534:nobody:/home:/bin/sh
eric2:x:1002:1002:,,,:/home/eric2:/bin/bash
```

Configurer un serveur

- Préciser que la machine est un serveur NIS et son type
  - Fichier /etc/default/nis

```
# /etc/defaults/nis Configuration settings for the NIS daemons

# Are we a NIS server and if so what kind (values: false, slave,
master)

NISSERVER=master

# Location of the master NIS password file (for yppasswdd).
# If you change this make sure it matches with /var/yp/Makefile.
YPPWDDIR=/etc
```

Configurer un serveur : sécurité

- Donner le nom de domaine
  - Commandes `domainname, nisdomainname.`
  - Fichier `/etc/defaultdomain`
- Sécurité
  - RPC donc `/etc/hosts.allow` et `/etc/hosts.deny`
  - `/etc/ypserv.securnets` : préciser les machines autorisées aux services NIS

```
# Always allow access for localhost
255.0.0.0       127.0.0.0
# This line gives access to everybody. PLEASE ADJUST!
0.0.0.0                 0.0.0.0
```

Configurer un serveur : maps

- Les maps sont des fichiers binaires indiquant aux clients NIS tous les paramètres du serveur.
  - Placées dans `/var/yp/nom_de_domaine`
- Edition du Makefile pour configurer la fabrication des maps
- Générer les maps : `make` ou `/usr/lib/yp/ypinit -m`

```
YPPWDDIR = /etc

PASSWD = $(YPPWDDIR)/passwd

ALL =   passwd group hosts rpc services netid protocols netgrp
networks

#ALL += amd.home auto.master auto.home auto.local
```

Synchronisation des esclaves

- Génération des maps sur le serveur maître.
  - `/usr/lib/yp/ypinit -m`
- Sur un serveur esclave
  - Pour contourner les pannes ou les attentes longues les esclaves ont une copie des maps du serveur maître mais doivent vérifier régulièrement que le serveur ne les a pas modifiées
  - `/usr/lib/yp/ypinit -s masterhost`
  - Pour être sûr que les maps NIS sont synchronisées, ajouter dans leur crontab les lignes

```
20 *     ***   /usr/lib/yp/ypxfr_1perhour
20 6     ***   /usr/lib/yp/ypxfr_1perday
20 6,18  ***   /usr/lib/yp/ypxfr_2perday
```

Fichier /etc/ypserv.conf

- Fichier de configuration du serveur
  - Si on ne veut pas que les client puissent accéder aux maps des passwd shadow par exemple

```
# and you can deny or restrict access to certain maps based
#on the originating host.
# Host          : Map             : Security    : Passwd_mangle
 *              : passwd.byname   : port        : yes
 *              : passwd.byuid    : port        : yes
 *              : *                 : none
# Default - restrict access to the shadow password file,
# allow access to all others.
*                : shadow.byname     : port
...
```

Configurer un serveur : résumé

- Préciser le type de serveur de la machine (master, slave)
- Etablir le nom de domaine : `nisdomainname`
- Donner ou retirer les droits d'accès aux clients : `/etc/ypserv.securnets` ou `/etc/hosts.deny`
- Configurer la compilation (`Makefile`) et créer les maps (`make` ou `/usr/lib/yp/ypinit`) et `crontab` pour configurer les serveurs NIS esclaves
- Configurer les accès aux maps NIS : `/etc/ypserv.conf`
- Relancer le démon NIS : `/etc/init.d/nis restart`

NIS, NFS et autofs

- But : permettre à un groupe de client de monter des partitions NFS dont la gestion est centralisée sur le serveur
  - Côté client on ne veut pas ajouter d'entrée dans /etc/fstab
  - Avantage : changement d'architecture de partage de fichier sans modification de toutes les fstab des clients
- automount : un SF n'est pas monté au démarrage mais quand un utilisateur y accède
- Le démon automount lit le fichier /etc/auto.master pour se configurer

NIS, NFS, autofs : regroupement

- Fichier /etc/netgroup
  - Regrouper des machines
  - Map NIS
  - Sert dans /etc/exports pour ne pas avoir à lister toutes les machines pouvant accéder à un SF

```
#/etc/netgroup (de la machine serveur NIS calif)
mongroup (zelda,,) (jigoro,,) (darwin)

#/etc/exports (sans netgroup)
/home zelda(rw) jigoro(rw) darwin(rw)

#/etc/exports (avec netgroup)
/home @mongroup(rw)
```

NIS, NFS, autofs : auto.master

- Création d'une carte sur le serveur NIS

```
#/etc/auto.home
calif   calif:/home
```

- Intégration de ces informations dans les maps NIS
  - Modifier le Makefile de yp pour prendre en compte les netgroups et le auto.home
  - /var/yp/make ou /usr/lib/yp/ypinit –m
- Côté client : il faut préciser à autofs qu'il y a une autre carte

```
#/etc/auto.master
/home   yp auto.home  --intr,nosuid,nodev
```

Plan

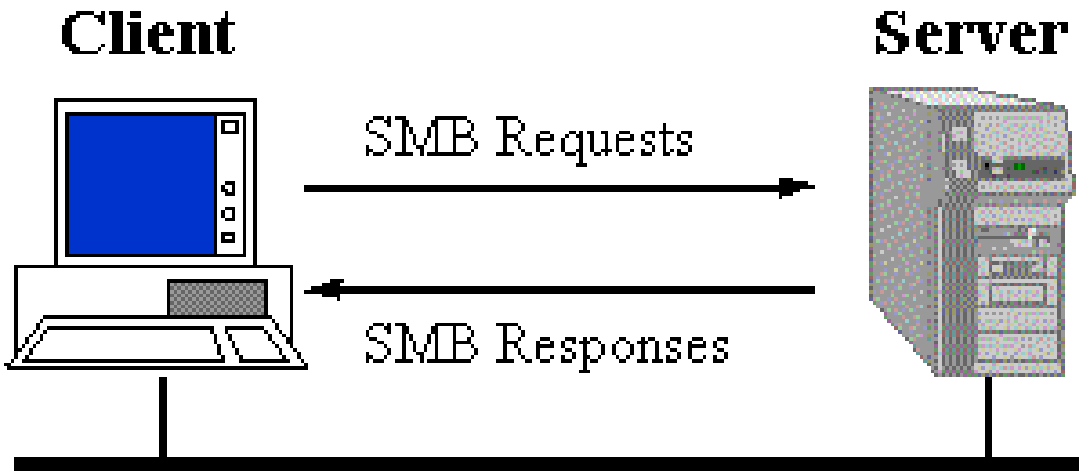
- Communications et transferts
- DHCP
- NFS
- NIS
- **Samba**
- DNS
- Sendmail

Présentation de Samba

- Ensemble de programmes : connecter à un serveur LINUX, des stations sous
  - Windows 3.11, Windows 9x, Windows Nt, OS/2, Mac....
- Le serveur Linux est en mesure de se conduire comme un serveur de fichiers capables d'offrir les services habituels sur un réseau :
  - partage de fichiers et de répertoires,
  - partage d'imprimantes,
  - respect des comptes utilisateurs
  - gestion des permissions d'accès
  - exécution de scripts de connexion personnalisés

Présentation de Samba

- Protocole de communication sous-jacent = communication Linux-Dos/Win
  - s'appuie sur NetBios et s'appelle smb = Server Message Block.
  - implémentation sur Unix d'une émulation d'un serveur LanManager, développé par Microsoft vers 1987
- Le projet Samba : initié dès 1991



Protocoles et démons

- Serveur offre ses ressources aux clients Windows
  - connexion sous un compte créé par root, après authentification
- Travail partagé par 2 démons:
  - smbd pour le service serveur
  - nmbd pour le service de résolution des noms Netbios.
- Côté client, le protocole SMB fonctionne au-dessus de plusieurs protocoles.
  - NetBIOS au dessus de TCP/IP

Protocoles et démons

- Demande de connexion d'une station au serveur Linux,
  - trace stockée dans un fichier log.%m, situé dans le répertoire /var/log/samba (%m désigne le nom de la station).
    identification précise puis examen sur une ligne de ce fichier

nom d'utilisateur, nom de la machine, date, heure de début, heure de fin, services utilisés...

- Sécurité au niveau de l'utilisateur,
  - non au niveau des ressources comme c'est le cas dans les réseaux de type WorkGroup.

Configuration : clients Windows

- Samba ne permet d'accéder à à la station Windows qu'à travers le protocole TCP/IP + besoin du protocole **NETBIOS**

- Sur chaque client :

  - ajouter TCP/IP et NETBIOS
  - vérifier que NetBios est activé avec TCP/IP (Voisinage réseau/Propriétés TCP/IP, onglet NetBios).
  - affecter une adresse IP à chaque station dans le même sous-réseau que le serveur Samba-Linux

- Exemple :

  l'adresse de sous-réseau **10.177.200.10x, x=1,2,3 ..**
  noms Microsoft des stations : **PCx** *par ex : PC1 .. PC8*
  nom de groupe de travail qui sera donné dans smb.conf : **fctice77**

Configuration : imprimante

- Imprimante déjà installée sur le serveur Linux et déclarée sur le serveur Samba (voir configuration dans /etc/smb.conf).
  - Lancer l'Assistant d'ajout d'imprimante (Paramètres /imprimantes)
  - Choisir imprimante réseau
  - Parcourir le *voisinage réseau* pour détecter l'imprimante : par exemple, choix de lp sur p00, le serveur Linux
  - Le nom de la file d'attente serait alors \\p00\lp
  - Choisir le modèle d'imprimante et le nom sous lequel elle apparaîtra sur la station cliente (par ex Olivetti sur PC1-Linux).
  - Pour vérifier faire imprimer une page de test.
- Désormais, l'imprimante partagée sera visible dans le voisinage réseau

Installation du serveur

1. Installer le package samba :

- ⇨ rpm -ivh samba-* (redhat)
- ⇨ Ou apt-get install samba (debian)

⇨ Tous les paramétrages dans un seul fichier :

- ⇨ /etc/smb.conf
- ⇨ ou /etc/samba/smb.conf

2. Vérifier et activer les changements

```
# conseillé plutôt que la commande smb restart
/etc/rc.d/init.d/smb stop
/etc/rc.d/init.d/smb start
```

Fichier smb.conf

- Organisé en sections (à la manière des fichiers ini de Win 3.x).
  - Les 2 principales sont **[global]** et **[homes]**.
  - L'administrateur root peut éditer, modifier et ajouter des sections = nouvelles ressources à partager
  - les permissions de partage définies dans ces sections ne peuvent pas outrepasser les permissions des fichiers du serveur hôte.

Smb.conf : [global] (1)

```
[global]

#donner le même nom de groupe de travail que celui des stations Windows
workgroup = FCTICE77
 # compte à utiliser pour les accès invités aux partages
guest account = nobody ;
# accès multi utilisateur
share modes = yes ;

# restreindre les sous-réseaux autorisés à se connecter au serveur
# on se limite aux adresses réseau 192.168.1.0
hosts allow = 192.168.1
# on peut exclure des machines de l'accès au réseau
hosts allow = 192.168.1 EXCEPT 192.168.1.15
# indique l'adresse IP du serveur et le masque de sous réseau
interfaces = 10.177.200.110/255.255.255.0
```

Smb.conf : [global] (2)

```
[global]

...

# indique l'emplacement du fichier printcap, récapitulant toutes les imprimantes
    installées sur le serveur Linux
printcap = /etc/printcap

 # partage toutes les imprimantes définies dans le fichier printcap
load printers = yes

 # utiliser un fichier de trace pour chaque machine qui se connecte
log file = /var/log/samba/log.%m

 # choisir le mode de sécurité : user ou share
security = user
```

Smb.conf : [homes]

Le répertoire personnel

```
[homes]
    # accès au rép. personnel de chaque utilisateur.
    # la valeur du champ comment apparaîtra dans le voisinage réseau
    # inutile pour cette section de préciser le path, c'est celui de l'utilisateur, en fait
    /home/%u

    comment = Répertoire personnel
    browsable = no
    writable = yes
    create mode = 0700
```

Smb.conf : [public]

Rendre un répertoire public

- Répertoire partagé totalement (lecture/écriture) par tous les users
  - le créer ou vérifier qu'il existe.
  - /home/ : lieu de regroupement des répertoires personnels:

```
mkdir /home/tmp
ls -l : renvoie les droits par défaut drwxr-xr-x
chmod 777 tmp : permissions d'accès et d'écriture pour tous
```

- Pour permettre le partage de ce répertoire commun /home/tmp
  - modifier la section `[public]` déjà présente et d'enlever les symboles ";" pour dé-commenter les lignes

Smb.conf : [public]

```
[public]

# Ce répertoire aura donc pour nom de partage " public "([public]),
# la valeur du champ comment apparaitra dans le voisinage réseau
# Le répertoire à partager est /home/samba
    comment =Répertoire public
    path = /home/samba

# il pourra être accessible par tous les utilisateurs
    public = yes

# il est accessible en écriture
    writeable = yes

# les fichiers créés sont en lecture seule, sauf pour le propriétaire
    create mode = 0755
```

Partager un répertoire : groupe

```
[stagiaire]

 # Ce répertoire aura donc pour nom de partage stagiaire
    comment = Partage pour le groupe stagiaire
# Le répertoire à partager est /home/stagiaire
    path = /home/rep-stagiaire
# il ne pourra pas être accessible par tous les utilisateurs
    public = no
# liste des utilisateurs autorisés
    valid users = stage1 stage2 stage3 ...
# ou mieux, indication du groupe autorisé
    valid users = @stagiaire jean
# on pourra y écrire (bien sûr par ceux qui peuvent y accéder..)
    writeable = yes
# les permissions par défaut des fichiers créés
    create mode = 0640
```

Partager des applications

```
[logiciels]

    comment = Applications partagées sur le serveur

# root doit créer ce répertoire et déléguer sa gestion à un groupe
 # Dans la suite, ce groupe sera appelé admin (contenant au moins l'utilisateur
    admin/admin) des droits de propriété et permissions
    path = /appli
    public = yes

# le rép. ne doit pas être accessible en écriture pour tous
    writeable =no

# le groupe admin peut seul installer les applications
    write list = @admin
```

Partager le lecteur CDROM

- On crée dans le fichier smb.conf une section cdrom et on indique le chemin d'accès **`path = /mnt/cdrom`**.

```
[cdrom]

# chemin d'accès au pseudo-répertoire de montage du CD
    path = /mnt/cdrom

# accessible à tous les utilisateurs
    public = yes

# l'écriture sera interdite
    writeable = no
```

Monter une partition windows

- Pour monter une partition Windows distante depuis linux avec samba
  - Utiliser le system de fichier smbfs

```
LABEL=/redhat6.0    /redhat6.0       ext2    defaults          1 2
LABEL=/zelda0       /zelda0          ext2    defaults          1 2
goldorak:/home/goldorak1  /homegoldo  nfs   noauto,users      0 0
LABEL=/zelda1       /zelda1           ext2    defaults          1 2
/dev/hdb6           swap              swap    defaults          0 0
/dev/hda5           swap              swap    defaults          0 0
//laury/commun      /commun           smbfs   defaults          0 0

...
```

Résolution des noms NetBios

- Le problème essentiel pour permettre la communication entre les réseaux utilisant NetBios et TCP/IP est la "résolution des noms", c'est-à-dire l'utilisation d'un service réseau qui se charge de la traduction
  - nom NetBios de machine / adresses IP
- Il y a 4 procédés de résolution, que le "démon" **nmbd** s'efforce de mettre en oeuvre. Leur ordre d'utilisation est fixé par la clause name resolve order dans /etc/smb.conf.
  - Par exemple :

```
name resolve order = wins host bcast lmhosts
```

Résolution des noms NetBios

- Par défaut, en l'absence de service de résolution de noms (paramètrage standard dans **/etc/smb.conf**), la résolution est tentée par diffusion (*broadcast*), Samba fait alors du "porte à porte".
- Si on utilise la méthode lmhosts, le fichier **/etc/lmhosts** doit être renseigné sur chaque machine, comme /etc/hosts, sous forme d'une table :
  - adresse IP (ou nom DNS) nom NetBios
- La commande **`nmblookup nom`** fournit si possible l'adresse IP de la machine, connaissant son nom Netbios.
-
```
$ nmlookup serveur querying serveur on 10.177.200.255
10.177.200.100 serveur
```

Serveur Wins

- La meilleure méthode de résolution des noms semble d'activer un serveur WINS (= *Windows Internet Name Server*) sur un serveur Samba
- Chaque station est affectée à un serveur Wins, et sait ainsi auprès de quelle machine elle doit se faire enregistrer, c'est-à-dire faire noter la correspondance entre son nom Netbios et son adresse IP.
- Configuration station
  - Voisinage réseau / Propriétés propriétés TCP/IP onglet configuration WINS Activer la résolution WINS et ajouter l'adresse IP du serveur WINS
  - et bien sûr redémarrer ..

Serveur Wins : serveur Samba

- Dans son fichier /etc/smb.conf, mettre en premier la méthode de résolution par wins et choisir une machine Samba dans le sous domaine qui supportera le serveur wins.
  - S'il s'agit de la machine qu'on configure (c'est le plus souvent le seul serveur Samba) on déclare wins support = yes, sinon on précise l'adresse IP après wins server = .
- Comme il ne doit y avoir qu'un seul serveur WINS dans le domaine, on doit n'activer qu'une seule des 2 options :
  - *# Activer un serveur Wins pour la résolution des noms* `NetBios name resolve order = wins host lmhosts bcast wins support = yes`
  - *# alors ne pas déclarer l'adresse IP d'une autre machine comme étant serveur Wins ;* `wins server = xxx.xxx.xxx.xxx`

Samba : contrôleur de domaine

- Un *contrôleur principal de domaine (PDC)* est un service chargé du contrôle de l'authentification des requêtes de connexion sur un réseau, par nom de connexion (login) et mot de passe (password).

- Samba peut assurer ce service en quelque sorte en émulant les fonctionnalités d'un serveur NT. Il en résulte de meilleures fiabilité et sécurité de la connexion sur le mode client/serveur, à la place d'un pseudo-voisinage réseau et ses aléas ...

  - Paramétrage du serveur : /etc/smb.conf.
  - Paramétrage des stations

Paramétrage du serveur (1)

Voici le paramétrage standard dans /etc/smb.conf.

```
# dans la section [global]

[global] workgroup = FCTICE77
netbios name = SERVEUR
server string = Serveur Samba

# active le service PDC
domain logons = yes

# sécurité au niveau utilisateur
security = user

# les mots de passe doivent être encryptés dans le fichier /etc/smbpasswd
encrypt passwords = yes
smb passwd file = /etc/smbpasswd
```

Paramétrage du serveur (2)

```
# pour être aussi serveur de temps
time server = yes

# niveau d'exécution du serveur ?
os level = 34

# ce serveur est le controleur du domaine
domain master =yes

# pour forcer la demande d'authentification pour tout partage (non
    recommandé) ;
 revalidate
```

Remarques

Le nom donné à la rubrique workgroup est le nom choisi pour le domaine.

C'est le nom à indiquer sur chaque client Windows

La valeur de `netbios name` est le nom du serveur Samba qui l'identifie sur le réseau.

A défaut, le serveur sera visible sur le réseau sous son nom DNS d'hôte

`domain logons = yes` va activer le service de contrôleur de domaine.

l'option `revalidate` renforce la sécurité, mais devient vite fastidieuse pour les usagers

Gestion des utilisateurs Samba

- Exemple pour transformer les utilisateurs locaux d'une machine en utilisateurs Samba

```
cat /etc/passwd | mksmbpasswd.sh > /etc/smbpasswd
```

- Cette commande ne peut regénérer les mots de passes cryptés, il faut donc les retaper pour chaque utilisateur

```
smbpasswd username
```

- Ne pas oublier de mettre dans smb.conf

```
encrypt passwords = yes
smb passwd file = /etc/smbpasswd
```

Paramétrage des stations

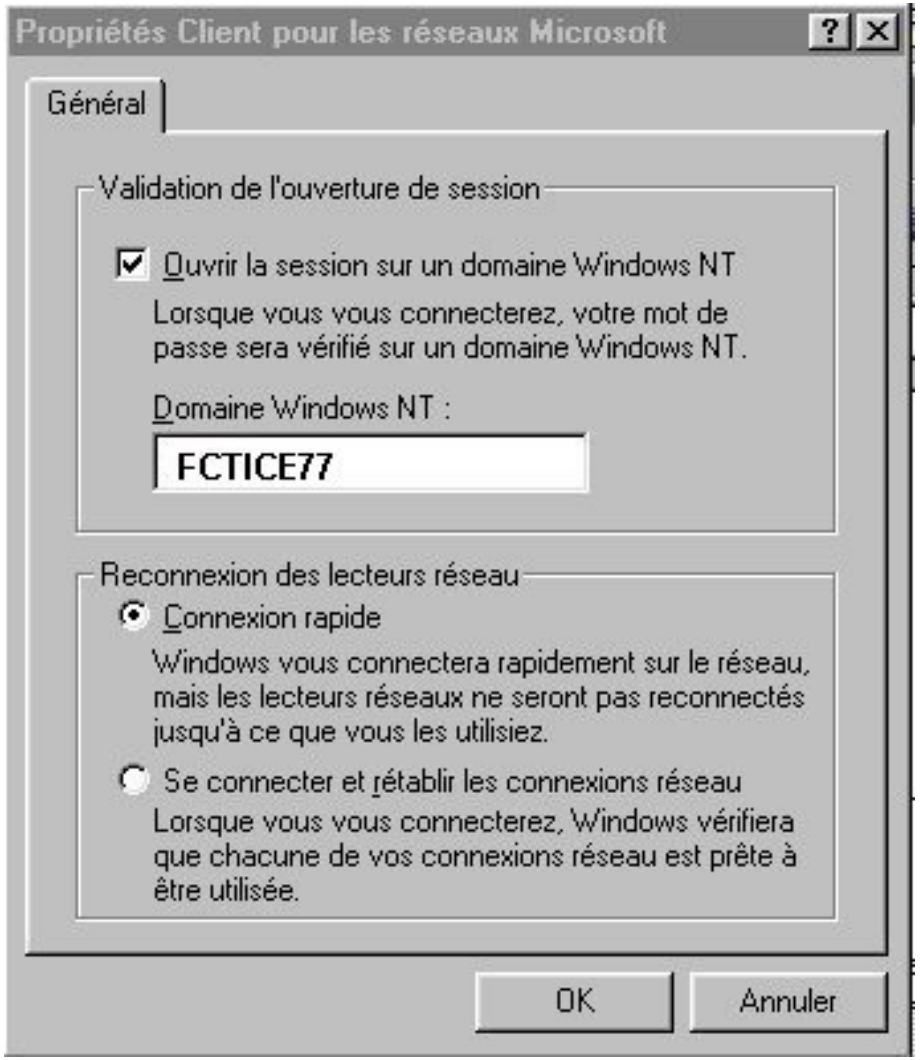
⇨ Sélectionner

- Voisinage réseau/propriétés
- Client pour les réseaux Microsoft/Propriétés
- Cocher *Validation de l'ouverture de session*
- Ecrire le nom Workgroup, ici FCTICE77
  comme nom de *Domaine Windows NT*
- Reconnexion des lecteurs réseau : cocher *Connexion rapide*
- Et bien sûr ... redémarrer !

⇨ Résultats

A la prochaine connexion, la boite de dialogue sera modifiée.
Le 3ème champ *Domaine* y sera ajouté.
A chaque demande de connexion, nom et mot de passe sont exigés, leur existence et validité sont vérifiés sur le contrôleur du domaine indiqué, sinon la connexion au réseau est refusée.

Paramétrage des stations



Scripts de connexion

- Principe

- Samba accepte la demande de script de connexion Windows (c-à-d les fichiers de commande **.bat**) liés à la connexion d'un utilisateur, et renvoie le fichier correspondant sur la station Windows, afin que celle-ci l'exécute.

- Ce mécanisme permet donc d'adapter complètement et dynamiquement la configuration du poste client au profil de l'utilisateur.

Scripts de connexion : config

```
[global]
 # ---- voir ci-dessus
 # le script de connexion porte le nom Windows %U de l'utilisateur cette
      clause suppose l'écriture d'un partage appelé [netlogon]
logon script = %U.bat
# Pour indiquer le chemin du répertoire personnel dans le script
# Ceci va permettre de le connecter avec use net H: /home
logon home = \\%L\%U
.....
[netlogon]
comment = Service de connexion réseau
# répertoire d'accueil choisi pour les scripts de connexion
path =/home/netlogon
# ce partage est privé, invisible et protégé en écriture
public = no
writeable = no
browseable = no
```

Scripts de connexion : variables

- %U : Nom utilisateur
- %m : nom NetBios de la machine cliente
- %T : date et heure de la [dé]connexion
- %I : N° IP de la station
- %S : nom du partage courant
- %L : nom NetBios du serveur
- %G : nom du groupe principal de %U

Scripts de connexion : exemple

- Script pour l'utilisateur stagex

- ATTENTION ! Il s'agit d'un fichier "batch" qui va s'exécuter sur la station Windows juste après l'authentification. Il doit donc être écrit avec un éditeur de texte DOS, sur la station, puis ensuite placé sur le serveur dans /etc/netlogon par root ou un utilisateur autorisé.

 # fichier /home/netlogon/stagex.bat

```
net use H: /home
net use L: \\serveur\logiciel
net use P: \\serveur\public
net use S: \\serveur\stagiaire
net time \\serveur /set/y
```

Scripts de connexion : exemple

- Effets

- Sur le client Windows, après authentification de la requête de connexion, le script attaché à l'utilisateur positionne les lettres des lecteurs réseaux qui "pointent" vers des partages valides et accessibles.

- En particulier le lecteur H : désigne bien le répertoire personnel

- **Installation de Openoffice pour le domaine**

  - Naturellement, il s'agit ici d'installer la version Windows sur le serveur Samba.

Installation pour un domaine

- Se connecter comme root ou administrateur du partage [logiciel]
- Le lecteur réseau L: pointe vers \\serveur\appli
- Clic sur l'archive OpenOffice.exe de 62 Mo ! Indiquer le répertoire de décompression L:\Openoffice-source, et Unzip
- Lancer l'installation par Exécuter : L:\OpenOffice-source\...Setup.exe /net
  Indiquer L:\OpenOffice comme répertoire d'installation
- Installation des composants locaux sur chaque station :
  Lancer L:\OpenOffice\soffice.exe, choisir *Installation d'utilisateur standard* et indiquer un répertoire local, par défaut C:\Office51 est indiqué.

Installation pour un domaine

- L'installation a créé une entrée dans le Menu Démarrer du disque de la station, dans C:\Windows\Menu Démarrer\programmes, qui pointe vers L:\StarOffice\soffice.exe.

  Si on veut que tous les utilisateurs en bénéficient, il suffit de le recopier dans tous les répertoires /home/user/Demarrer/Programmes, bien sûr avec un petit script pour automatiser ...

- Lors du premier lancement sur un nouvelle station, le processus d'installation local sera déclenché.

- H: désigne bien le répertoire personnel