Sécurité et administration réseaux

Yves Caniou <yves.caniou@ens-lyon.fr>

Module CCIR4 de l'UCBL – Master 2

2005-2006

(version du 14 février 2006)

Présentation du module CCIR4

Sécurité dans les systèmes informatiques

Contenu et objectifs du module

- Sécurité grâce à la configuration
  - Installation de machines (desktop/serveur)
  - Administration des machines
- Sécurité grâce à l'utilisation d'outils (audit)
  - Protocoles sécurisés
  - Services sécurisés et leur gestion

Motivations

- Outil informatique omniprésent
- Besoin de sécurité dans les équipements

Prérequis : Système d'exploitation et notion de réseau

Évaluation

- Un examen sur table (date et modalités à préciser)

Organisation du module

Choix des «systèmes d'exploitation»

- Produits Microsoft présentés par Anne-Lyse
  - Win 2000
  - Win NT
- Gnu/Linux dans ce cours

Pourquoi ces systèmes ?

- Implantations différentes d'Unix
- Les plus répandus avec des licences d'utilisation différentes

Alternatives vis-à-vis de la sécurité : OpenBSD ! (version 3.8 le 1 nov 2005)

Quelques références pour faire ce cours 1/2

Les docs de référence

- Le guide de référence Debian, le «Securing-debian-howto»

  `http://ploug.eu.org/doc/installation-debian.pdf`
  `http://www.debian.org/doc/manuals/securing-debian-howto/securing-debian-howto.fr.pdf`
- Les HOWTO's
- Les manpages !

Les magazines : Misc, Linux Mag', Hackin9, Login

Sites d'information

- `http://www.developpez.com/` : index de cours et tutoriels
- `http://www.commentcamarche.net/`

Liens complémentaires intéressants

- Base d'administration pour le superutilisateur (en français !)
  `http://www.loli-grub.be/contrib/tlepoint/BASE/version-internet.html`
- `http://people.via.ecp.fr/~alexis/formation-linux/formation-linux.html`
- `http://www.linux-france.org/prj/inetdoc/cours/`
- `http://www.fr.linuxfromscratch.org/view/blfs-1.0-fr/`

Quelques références pour faire ce cours 2/2

Support de cours d'autres personnes

- yst. d'exploitation de Martin Quinson : `http://www.loria.fr/~quinson`
- Client/Serveur d'Olivier Glück : `http://www710.univ-lyon1.fr/~ogluck/supports_enseig.html`

URL du cours

- `http://graal.ens-lyon.fr/~ycaniou/teaching/0506.html`

Plan du module, partie Unix :

0. Avant-Propos
   Système d'exploitation et un peu de droit
1. Installation d'une machine
   Arborescence Unix ; Avant l'install et pendant l'install
2. Administration d'une machine
   Ouverture d'une session locale ; Shell et scripts shell ; Gestion des packages ; Configuration de la machine ; Configuration du réseau ; Description des services au démarrage.
3. Admininstration d'un parc de machines GNU/Linux
   Fichiers distribués (NFS) ; Les pages jaunes (NIS) ; Serveur de noms (DNS) ; Annuaire distribué (LDAP) ; Les proxies ; Réseaux privés (VPN) ; Système de contrôle de version (CVS) ;

Première partie

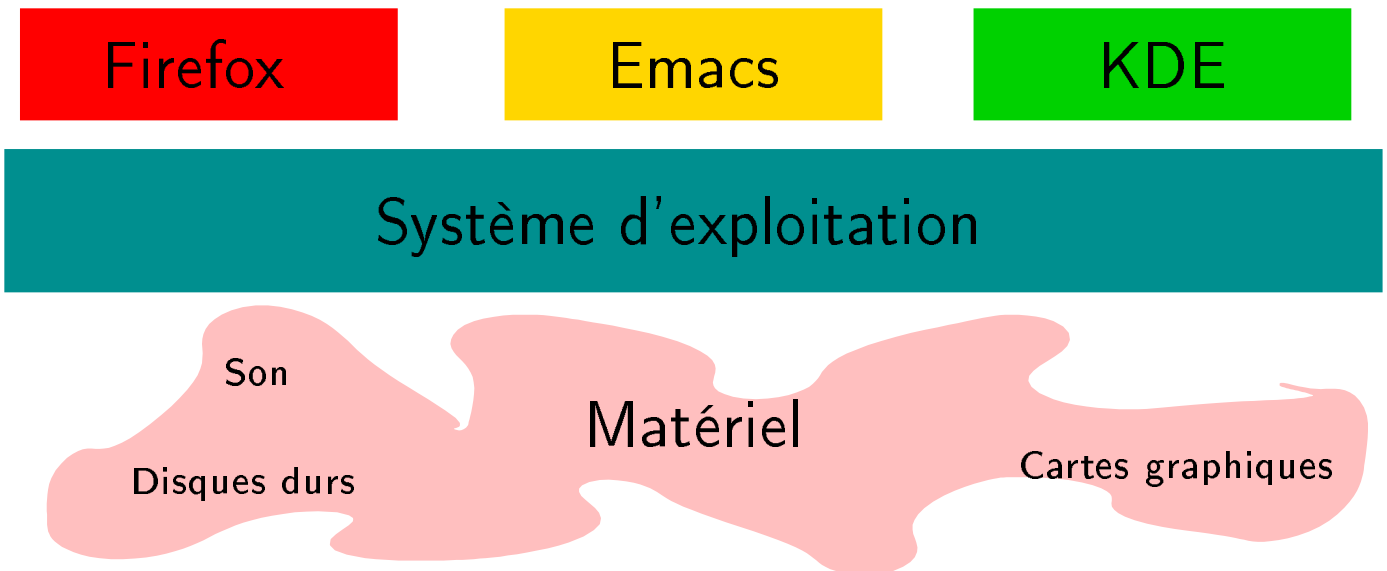
Avant-Propos

- Qu'est ce qu'un système d'exploitation ?
- Le système GNU/Linux
- La distribution Debian et sa licence
- Rappels des objectifs
- Un peu de droit et quelques définitions

Qu'est ce qu'un système d'exploitation ?

Logiciel entre les applications et le matériel

- Offre une interface unifiée aux applications
- Gère (et protège) les ressources



Le système GNU/Linux

Le noyau : linux

- 1991 : Linus Torvald écrit un OS Unix-like.
- Linux est accessible en libre téléchargement sur le net, GPL !
- Nombreux développeurs, modifications supervisées par Linus
- De nombreuses architectures, de nombreux drivers

Les logiciels GNU

- Projet datant de 1984, «Free software Foundation»
- Un ensemble complet de softs, complémentaires (emacs, gcc, bash, gnome)

Les distributions

- Faciliter l'installation, la maintenance des logiciels
- Packages : exécutables, scripts, documentations, bibliothèques...
- Plusieurs distributions : Mandrake, Red-Hat, Suse, Slackware

La distribution Debian et sa licence

- Née en 1993 : Ian Murdock
- Exigence : la liberté → promouvoir les logiciels libres
- De très nombreux packages, de très nombreuses documentations !
- De nombreuses architectures supportées
- Gestion performante des packages (dépendance, sécurité)
- Mises à jour faciles
- Stable, testing, unstable

Licence GPL (*Gnu Public License*)

- Copiez et donnez des copies du système !
- Adaptez les logiciels à vos besoins (vous pouvez même distribuer le résultat)
- Attention : logiciels fournis sans aucune garantie
- Obligation de fournir le code source
- Rq : existence d'une branche `non-free`

Nos objectifs dans ce module

- Installer le système GNU/Linux de la distribution Debian stable (Sarge)
  « Les packages vont de unstable vers testing après une semaine complète où pas un bug critique de signalé. Pour les stations de travail, on utilisera plutôt testing, pour les serveurs, plutôt stable. Attention dans ce cas, car testing reçoit plus lentement correctifs sécurité.»
- Administrer et sécuriser l'installation
- Administration de plusieurs machines en réseau
  - NFS, base centralisée d'utilisateurs...
- Sécuriser le système
  - Connexions sécurisées et PKI
  - Firewall et proxy, filtrage de mails
  - Utilisation de partitions chiffrées

Première partie

Avant-Propos

- Qu'est ce qu'un système d'exploitation ?
- Le système GNU/Linux
- La distribution Debian et sa licence
- Rappels des objectifs
- Un peu de droit et quelques définitions

Tromper les individus et Droit

- Escroquerie : définie aux **art. 313-1 à 313-3 du Code Pénal** :
  « Le fait, soit par l'usage d'un faux nom, soit par l'abus d'une qualité vraie, soit par l'emploi de manoœuvres frauduleuses, de tromper une personne physique et de la convaincre à remettre des fonds, des valeurs ou un bien quelconque ou à fournir un service ou à consentir un acte opérant obligation ou décharge »
  Jusqu'à 5 ans d'emprisonnement et 375 000 € d'amende
  - Phishing et scam
- Usurpation d'identité : patch sécurité Microsoft, mail
  - La Cnil[1] qualifie l'adresse électronique de donnée à caractère personnel
  - Sanction pratiquée pour usage illicite de fausse identité de 3 ans d'emprisonnement si bande organisée. En pratique, 4 mois avec sursis.
  - Exemple : 21 mois d'emprisonnement avec sursis + 39h travaux d'intérêt général en hôpital ou en maison de retraite pour le concepteur de Sasser.
- Fausses informations et hoaxes[2]
  - Mentir à la Cnil (loi **92-1336**)
  - Dénonciation mensongère quant au retrait de contenus illicites (loi **LCEN**, **art. 6-1-4**, 1 an d'emprisonnement et 15 000 € d'amende)
- Social Engineering

1 2 **Commission nationale de l'informatique et des libertés** : http://www.cnil.fr
http://www.hoaxbuster.com/

Tromper les systèmes et Droit

- STAD : relève du Code Pénal, **Chap III, « Des atteintes aux systèmes de traitement automatisé de données », art. 323-I à 323-7**
  - **art. 323-1** condamne le fait d'accéder et se maintenir frauduleusement dans une système. Jusqu'à 3 ans de prison et 45 000 € d'amende
  - **art. 323-2** sanctionne le fait d'entraver ou fausser le fonctionnement d'un Stad de 5 ans de prison et 75 000 € d'amende
  - **art. 323-3** condamne l'introduction, la suppression ou la modification frauduleuse de données. Jusqu'à 5 ans de prison et 75 000 € d'amende
  - **art. 323-3-1** sanctionne la détention et la mise à disposition de moyens permettant les délits cités dans les articles 323-1 à 323-3 de la même manière que les délits
  - **art. 323-4 à 323-7** sanctionnent la préparation des délits, l'intention, et prévoient des peines supp. comme interdiction droits civiques et fermeture des établissements utilisés pour commettre les délits.
- Spam
  - Spam : **LCEN art. 22** subordonne l'utilisation de courriels dans les opérations de prospection commerciale au consentement préalable des personnes concernées
  - *Spam-indexing* pour tromper les moteurs de recherche
    - Densité et choix des mots clefs
  - *Cloaking*, création de pages spécifiques à chaque robot
  - *Googlebombing*, liens entre sites pour gonfler artificiellement leur notoriété

Deuxième partie

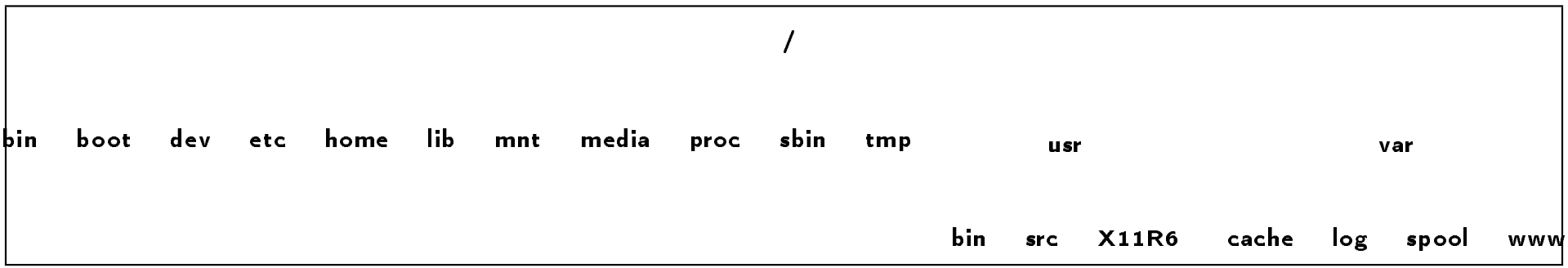
Installation d'une machine

- Rappel sur l'arborescence d'Unix
- Avant l'installation
  - Le BIOS de la machine
  - Partitionnement du disque dur
- Les étapes de l'installation

Organisation de l'arborescence d'Unix

Sous Linux, TOUT EST FICHIER → UNIQUE ARBORESCENCE

- sa racine est nommée / et l'administrateur est root
- Fichiers : normaux ; répertoires ; spéciaux ; liens symboliques



- `/bin` : **exécutables**
- `/boot` : **noyau vmlinuz et fichiers de démarrage**
- `/dev` : **rép. de fichiers spéciaux (communication avec les périphériques)**
- `/etc` : **rép. de fichiers de configuration et principaux scripts de paramétrage**
- `/home` : **racine des répertoires perso (...attention...)**
- `/lib` : **bibliothèques et les modules du noyau**
- `/mnt`, `/media` : **racine des points de montage (et /cdrom..)**
- `/sbin` : **exécutables pour l'administration du système**
- `/tmp` : **stockage des fichiers temporaires**
- `/usr` : **prog. accessibles à tout utilisateur (pas que...)**
- `/var` : **par exple fichiers d'impression, traces de connexions http**
- `/proc` : **pseudo-répertoire**

Le boot de la machine

Au *boot*, le BIOS (*Basic Input/Output System*) exécute une séquence d'instructions

- Configuration CMOS : Alt-F2 ; Alt-F9 ; F1 ; Del ; Esc ; Ctrl-Alt-Esc ...
  - Réglage heure/date
  - Sécurité sur le BIOS : réglage ordre de boot ; mot de passe
- Initialise horloge interne et contrôleur DMA
- Teste CPU, vérifie le BIOS
- Teste si *ROM* corrompue, teste *RAM*
- Vérifications des *devices*
  - Souris, clavier
  - lecteurs cdrom, dvd
  - disques durs
  - réseau
  - USB
- **Essaye de booter sur l'un des périphériques selon ordre donné**

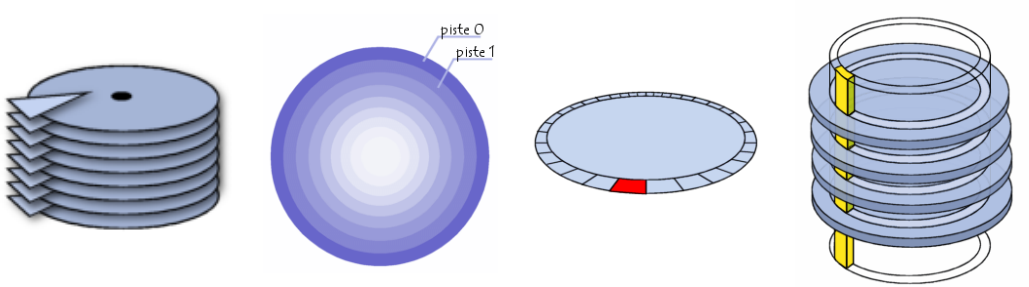
Changement du BIOS : un exécutable

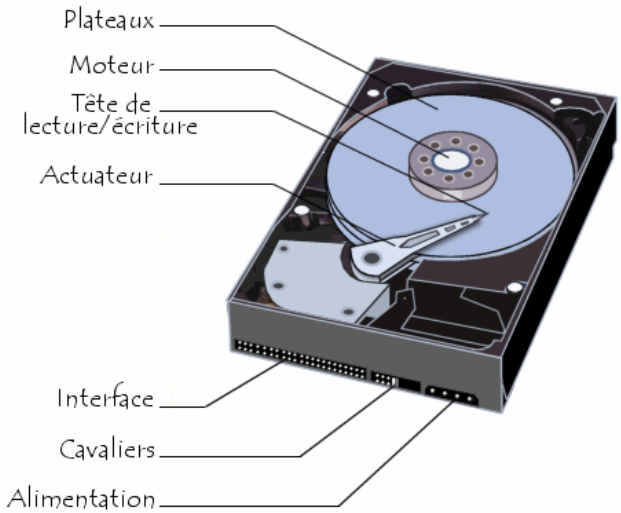
Attention aux pannes de courant..

Le disque dur ou *hard drive*

Au tout début était le disque dur vierge...

Avant CHS, maintenant LBA





Un disque dur...

- se partitionne $\longrightarrow$ multi-boot possible !
- se formate

À propos du partitionnement

- **4** *partitions primaires* maximum
- Moins de primaires et une *partition étendue* : contient partitions *logiques*

Formatage d'une partition crée le système de fichiers : ext2, ext3, ReiserFs...

(liste dans `/proc/filesystems`)

Les étapes de l'installation

- Rq : supports utilisés[3] (cdrom*, USB...) → en TP, cdrom puis réseau
- Niveau d'install : expert ou non, kernel 2.4 ou 2.6
- Un partitionnement en général
  - 1 partition /
  - 1 partition `swap` (taille double de RAM)
  - 1 partition `/home` ou `/users` (dépend si NFS)
- choix langue, réglage heure etc.
- Sélection des sources
- Attention sur
  - mots de passe en shadow (OUI)
  - `/home/` en lecture
  - Gestion protocole SSH2 (on y reviendra)
- La machine reboote avant l'installation de l'ensemble des packages !

Une install-party est prévue en TP, avec démontage de machine !

[3] `http://ploug.eu.org/doc/installation-debian.pdf`

Troisième partie

Administration d'une machine GNU/Linux

- Se connecter
- Shell et scripts shell
- Installer/configurer des packages sur Debian
- Configuration de la machine
  - Ajout d'utilisateurs/groupes
  - Connaître le hardware
  - Que l'affichage graphique soit !
  - Compiler son noyau
  - Retour sur les périphériques
  - Gestion des imprimantes
  - Le mail
- Configurer le réseau
- Description des services au démarrage
  - Les services au démarrage
  - Jobs synchrones/asynchrones
- Conclusions

Après le démarrage

Mode graphique, ici avec kdm

Mode console, ici avec welcome2l

- Rq préliminaire : certaines choses de ce qui suit peuvent se faire graphiquement
  Mais pas toujours de X → remote access ou console sur serveur
- Possibilité d'avoir plusieurs sessions X, plusieurs consoles (`tty`) : `/etc/inittab`
  → Ctrl-alt-F7, Ctrl-alt-F8... / alt-F1, alt-F2...
- De l'intérêt d'être un utilisateur et pas root...
  bugs, erreurs manip, etc.

Se logguer

Mode console :

- Un shell est lancé après authentification positive avec `/bin/login` en lisant le fichier `/etc/passwd`
- À chaque shell de connexion, fichier `~/.bash_profile` ou `/etc/profile` lu
- `~/.bashrc` ou `/etc/bashrc` lu si shell sans fonction de connexion
- Apparition d'un prompt...

Rq : noms dépendant du shell de l'utilisateur ! Ici, bash
Rq : `~/.bash_logout` lu quand utilisateur se déconnecte du système

Mode graphique :

- X exécuté (mécanismes décrits plus loin)
- Pas de shell de connexion !
- Authentification réussie $\longrightarrow$ ouverture session graphique

Exemple de fichiers .bashrc et .bash_profile

~/.bash_profile

```
# settings for french speaking users
# set LANG
export LANG=fr_FR@euro


if [ -f ~/.bashrc ]; then
  source  /.bashrc
fi
setxkbmap -layout fr
```

~/.bashrc

```
if [ -f /etc/bash_completion ]; then
  . /etc/bash_completion
fi

PATH=.:..:/usr/local/bin:$PATH:/sbin:/usr/sbin
alias ls='ls --color=auto'
alias ll='ls -l'
alias l='ls -l'

# Redéfinit le prompt...
PS1="\[\033[01;33m\]\h\[\033[00;39m\]:\[\033[/
01;36m\]\w\[\033[00;39m\]>"

alias mplayer="mplayer -vo xv -ao sdl -fs''
alias rmc="rm *~"
alias gv="gv -scale=-2 "
```

Troisième partie

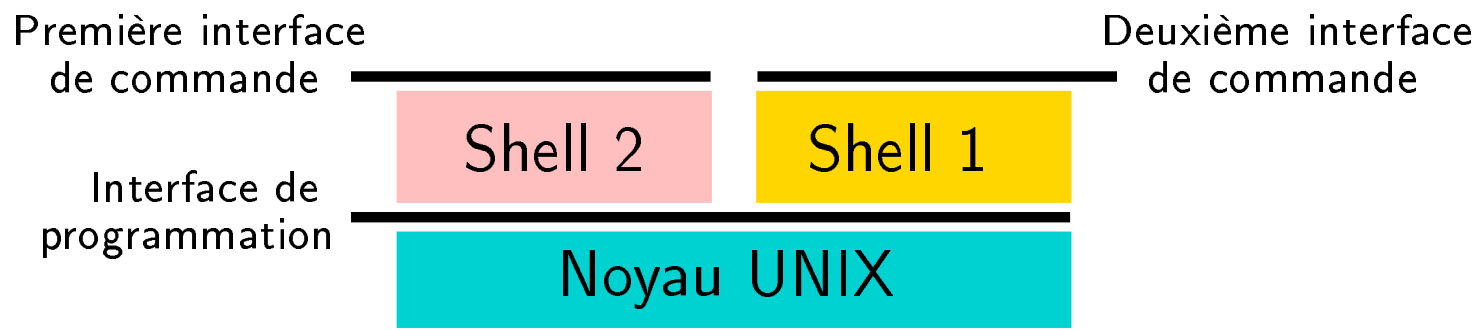
Administration d'une machine GNU/Linux

Interfaces d'un système d'exploitation

En général, deux interfaces

- Interface de programmation (**A**pplication **P**rogramming **I**nterface)
  - Utilisable à partir des programmes s'exécutant sous le système
  - Composée d'un ensemble d'appels systèmes (procédures spéciales)
- Interface de l'utilisateur, ou interface de commande
  - Utilisable par un usager humain, sous forme textuelle ou graphique
  - Composée d'un ensemble de commandes
    - Textuelles (exemple en UNIX : `rm *.o`)
    - Graphiques (exemple : déplacer l'icone d'un fichier vers la corbeille)

Exemple sous UNIX :



Exemple d'usage des interfaces d'UNIX

- Interface de programmation

  Ci-contre : programme copiant un fichier dans un autre

  (`read` et `write` : appels système)

- Interface de commande

  `$ cp fich1 fich2`

  recopie fich1 dans fich2

```
while (nb_lus = read(fich1, buf, BUFFSIZE )) {
  if ((nb_lus == -1) && (errno != EINTR)) {
    break ; /* erreur */
  } else if (nb_lus > 0) {
    ou_ecrire = buf ;
    while (nb_ecrits =
            write(fich2,ou_ecrire,nb_lus)) {
      if ((nb_ecrits == -1) && (errno != EINTR))
        break ;  /* erreur */
      else if (nb_ecrits == nb_lus)
        break ;  /* fini */
      else if (nb_ecrits > 0) {
        ou_ecrire += nb_ecrits ;
        nb_lus -= nb_ecrits ;
    } }
    if (byteswritten == -1)
      break ;
} }
```

- Documentation

  `man 1 <nom>` : documentation des commandes (par défaut)

  `man 2 <nom>` : documentation des appels système

  `man 3 <nom>` : documentation des fonctions

  d'autres sections, mais plus rares. Voir `man 7 man` ;)

Shells et scripts shell

- Plusieurs shells disponibles
  bash, csh, tcsh, ksh, zsh, ...mais installés sur machine ? → `/etc/shells`
- `~/.bashrc` : mettre en place environnement
- Exemple de variables d'environnement : `PATH`, `LD_PATH` (taper `$> env`...)
- Quelques commandes système : `ls`, `cd`, `mkdir`, `mv`, `rm`, `echo`
- Quelques autres commandes : `grep`, `man`, `passwd`, `ps`, `w`, `groups`

Exemples :

- `$> echo $PWD`
- `$> cat ~/.bashrc`

Pour changer de shell : `chsh`

Scripts shell - 1/8

On se sert de commandes shell séquentielles dans un fichier : script

Intérêts :

- Si tâche non ressource-intensive pour facteur vitesse
- Si pas de calculs complexes d'arithmétique...
- Si pas d'opérations extensives sur les fichiers
- Si pas besoin de structures complexes de données
- Sinon préférer `C`, `C++/java` ou `perl`, `python`, `ruby`...

Convention : `script.sh` (Ne pas oublier `chmod +x script.sh`)

Exemples :

```
#!/bin/bash
echo “Affiche $1”
# Commentaire
a=5 echo “a=$a”
```

```
#!/bin/bash
E_NOTROOT=67 # Non-root exit error.
ROOT_UID=0 # Only users with $UID 0 have root privileges
# Check if root
if [ "$UID" -ne "$ROOT_UID" ]
then
  echo "Must be root to run this script."
  exit $E_NOTROOT
fi
LOG_DIR=/var/log
# Variables are better than hard-coded values.
cd $LOG_DIR
cat /dev/null > messages
cat /dev/null > wtmp
echo "Logs cleaned up."
exit # The right and proper method of "exiting" from a script.
```

Scripts shell - 2/8

Caractères spéciaux : #, ;, ”, ’, ‘, ., *, \, $, ?

```
#!/bin/bash
######
# Caractere spéciaux

echo "The # here does not begin a comment."
echo 'The # here does not begin a comment.'
echo The # here begins a comment.
echo `seq 1 9`

echo ${PATH#*:}       # Parameter substitution, not a comment
# ${var#Pattern}, ${var##Pattern}
# Retire de $var le + court/long morceau de $Pattern qui matche le début de $var

# Exemple

echo ${PATH#*:}
echo $PATH
echo ${PATH#*/}
```

Scripts shell - 3/8

Affectation

Exemple 1

```
#!/bin/bash
###### # Affectation
echo -n "Values of "a" in the loop are : "
for a in 7 8 9 11
do
  echo -n "$a "
done

echo

for a in 1 2 3 4 5
do
  echo -e "Press keys : "
  read b
  c=$b$c
done
echo "Reversed pressed keys : $c"

arch=$(uname -m)
echo -e "Ran on a $arch \n\n"
```

Exemple 2

```
#!/bin/bash

for i in `ls`
do
  cp $i /dir/$i
  echo "$i copie "
done
# Ne pas oublier pas les ` qui forcent
#  l'exécution du ls

for dir in /dev /usr /users/bin /lib ; do
  num=`ls $dir|wc -w`
  echo "$num fichiers dans $dir "
done

for i ; do
  echo $i
done
```

Scripts shell - 4/8

Variables non typées : strings ou entiers

```
#!/bin/bash
######
# Variables non typées
a=2334                   # Integer.
let "a += 1"
echo "a = $a "           # a = 2335
echo                     # Integer, still.


b=${a/23/BB}             # Remplace “23” par "BB"
                         # This transforms $b into a string.
echo "b = $b"            # b = BB35
declare -i b             # Declaring it an integer doesn't help.
echo "b = $b"            # b = BB35


let "b += 1"             # BB35 + 1 =
echo "b = $b"            # b = 1
echo
```

Scripts shell - 5/8

Variables spéciales : $0, $1, etc. ; $*, $@

```
#!/bin/bash
echo "Il y a $# paramètres"
index=1

for arg in "$*" ; do
# Doesn't work properly if "$*" isn't quoted
  echo "Arg #$index = $arg"
  let "index+=1"
done # $* sees all arguments as single word
echo "Entire arg list seen as single word."

echo

index=1 # Reset count.
# What happens if you forget to do this?

echo "Listing args with \"\$@\" :"
for arg in "$@"
do
  echo "Arg #$index = $arg"
  let "index+=1"
done # $@ sees arguments as separate words.
echo "Arg list seen as separate words."
```

```
index=1 # Reset count.
echo "Listing args with $* (unquoted) :"
for arg in $*
do
  echo "Arg #$index = $arg"
  let "index+=1"
done # Unquoted $* sees arguments as
     # separate words.
echo "Arg list seen as separate words."
exit 0

# Comment accéder au 12e argument ?

######

# utilisation de shift,
# voir les variables internes
# echo $PWD
# echo ${GROUPS[1]}
# echo $HOME
# echo $HOSTNAME
# echo $PATH
# echo $PS1 → regarder le ∼/.bashrc
```

Scripts shell - 6/8

- Quoting : quoting et escaping ($\backslash n$, $\backslash t$)
- Exit status
- Tests
  - Opérateurs de test sur fichier : `-e`, `-f`, `-s`, `-d`

    ```
    if [ -x "$filename" ] ; then cmd fi
    ```
  - D'autres opérateurs de comparaison :
    - sur les entiers : `-eq`, `-ne`, `-gt`, `-ge`
    - sur les strings : =, == et != (literal/pattern matching), <, >...
  - Les opérateurs de calcul
  - Les boucles : `for i in [list] ; do cmd done`
  - Contrôler les boucles :
    - `while [ test ] ; do cmd done`
    - `break` ; `continue`
- Fonctions

Scripts shell - 7/8

Exemple 2 : lister «en arbre» de façon récursive (∼ `ls -R`)

```
#!/bin/bash

if [[ $# -ge 2 || $# -eq 0 ]] ; then
  echo "Input the file/directory to list"
  exit
fi

list_file () # list the file given in param
{
  echo -e $3"list $1"
}

list_directory () # list dir given in param
{
  echo "Listing directory $2$1"
  cd $2$1
  for i in `ls -a` ; do
# echo ${i}
    if [[ "$i" != "." && "$i" != ".." ]]
    then
      if [ -d $i ] ; then
        list_directory $i $2$1/ "__"$3
        cd $2$1
      else
        list_file $i $2$1 "__"$3
      fi
    fi
  done
}
path2here=""

if [ -f $1 ] ; then
  list_file $1 $path2here ""
elif [ -d $1 ] ; then
  list_directory $1/ $path2here ""
else
  echo "Input is neither a file or a dir"
fi
```

Scripts shell - 8/8

Rqs :

- À propos de lancer un script et de `source`
- Je n'ai pas trop insisté sur certains caractères spéciaux
  → `ls [^ ml]*.tex`, `ls *ab ?png`
- Je n'ai pas parlé explicitement d'expressions régulières
  → `http://www.easyprogs.com/index.php?callPage=./Programmation/Shell-Batchs/index.php`
- Je n'ai pas parlé de `grep`, ni de `sed` ou encore de `awk`
- Vous devez au moins comprendre les shell scripts !

Doc de Bash en français plutôt pas mal :

`http://pages.videotron.com/bash/doc/docbashfr.html`

Troisième partie

Administration d'une machine GNU/Linux

La commande dpkg

Rqs :

- dpkg permet d'installer/désinstaller packages
- Ne gère pas les dépendances ! → éviter en temps normal

Les options :

- dpkg -i package1.deb package2.deb → installation de packages
- dpkg -r [--purge] package1 → désinstallation [+ effacement fichiers config]
- dpkg-reconfigure package → lance le script de reconfig de package
- dpkg -S fichier → recherche du package contenant fichier
- dpkg -L package → liste les fichiers installés par package
- dpkg --get-selections/--set-selections → donne/restaure status de l'ens des packages

Exemples :

- dpkg -S /bin/bash
- dpkg --get-selection > packages_installes.txt

La commande apt-get

kézako ?

- Utilise le fichier `/etc/apt/sources.list`
- Couche intelligente de gestion des packages

Utilisation

- apt-get update → met à jour la liste des packages
- apt-get [-s] install package → [simule] installation de package
- apt-get upgrade → met à jour l'ens de l'installation
- apt-get dist-upgrade → évolution de stable vers testing vers unstable
- apt-get remove [--purge] package → désinstallation [+ fichiers config] de package
- apt-get clean → vide cache
- apt-cache search keyword → chercher mot clé
- apt-cache show package → donne info sur package

Remarque : packages stockés dans /var/cache/apt/archives

Exemples :

- apt-cache search mplayer
- apt-cache search tux

D'autres outils

Texte

- `tasksel` : pour sélectionner des groupes de packages debian
- `dselect` : système de menus texte
- `aptitude` : censé remplacer `dselect`

Graphiques

- kpackage : interface graphique aux gestionnaires de paquetages RPM, Debian, Slackware et BSD intégrée à KDE
- gnome-apt, gdeb : outils Gnome

Troisième partie

Administration d'une machine GNU/Linux

Ajout d'utilisateurs/groupes

Groupes

- Notion de groupes très importante : droits (audio, cdrom mais aussi lp, src)
- Ajout group : `addgroup [--gid ID]`
- Effacer : `delgroup`
- Liste dans `/etc/group`

Utilisateur

- `adduser` ou `useradd`

  adduser [--home REP] [--shell SHELL] [--no-create-home] [--uid ID] [--group | --ingroup GROUPE | --gid ID] utilisateur

- Après coup : `adduser user group`
- Effacer : `deluser`
- `/etc/adduser.conf` contient infos relatives à création compte

Rqs :

- Répertoire `/etc/skel` contient fichiers qui seront copiés par défaut dans tout nouveau comptes utilisateur
- Le nom de ce répertoire est définit dans `/etc/adduser.conf`
- `passwd` pour changer les mots de passe !

Troisième partie

Administration d'une machine GNU/Linux

Connaître le hardware de sa machine

- Les books livrés avec matériel
- Pour mieux configurer, s'informer : utilisation des commandes
  - `lspci [-vv]` (contenue dans le package `pciutils`)
  - `dmesg`
- Regarder les fichiers de log dans `/var/log`
  - `/var/log/dmesg`
  - `/var/log/syslog`

Attention aux droits !

Troisième partie

Administration d'une machine GNU/Linux

L'affichage graphique : *X window Consortium*

Késako ?

- Appelé aussi 'X' ou 'X11'
- L'un des plus grand succès OpenSource
- Un standard pour les OS Unix
- Procure l'affichage graphique pour de multiples architectures

Mais encore...

- Indépendant de l'OS et du hardware
- Connexion transparente via le réseau
- Supporté par la plupart des fournisseurs de hardware
- Plus de 30.10^6 utilisateurs

Fonctionne sur le modèle client/serveur (et sera vu en CCIR5 :)

Dans Debian, XFree86 et maintenant x.org...

L'affichage graphique avec X.org

Installation/Configuration

- Packages : `x-window-system`, `x-window-system-core`, `xbase-clients`, `xserver-xorg`, `x11-common`...
- Progs/Config/Logs
  - `/usr/X11R6/`
  - `/etc/X11/xorg.conf`
    (note sur clavier, drivers (vesa, proprio), souris/touchpad et résolutions)
    (`dpkg-reconfigure xserver-xorg` ou édition *à la main*)
  - `/var/log/Xorg.$i.log`

Exemple

- de `/etc/X11/xorg.conf`
- de `/var/log/Xorg.0.log`

Lancer X (mais pas trop loin)

Mode graphique : alors X est déjà lancé (via xdm, gdm ou kdm)

- Il y a eu exécution de `/etc/X11/Xsession` (script sh)
- `~/.xsession`
- → Reste plus qu'à choisir le *desktop*/*window manager* voulu

Mode console :

- `/usr/X11R6/bin/startx` : front-end pour `/usr/bin/X11/xinit`
- `~/.xinitrc`

Rq : sur l'accélaration 3D

- avec `glxinfo`
- `DRM` dans le noyau, `DRI` pour les packages
- avec les logs !

Troisième partie

Administration d'une machine GNU/Linux

Compiler le noyau

Pourquoi ? puisqu'il y en a un fourni dans la distribution...

- Noyau plus petit, plus rapide
- Nécessité de drivers, éventuellement patches
- Corrections de failles

Noyau et numérotation

- branches stables 2.2, 2.4 et 2.6...
- mais numérotation 2.6 spéciale
- lequel choisir ?

Trouver les informations utiles

- `/proc/cpuinfo`
- `lspci [-vv]`

Compiler son noyau linux sur mesure - 1/2

Méthode différente de ce que préconise debian (chacun ses goûts et c'est si simple)

Récupérer le noyau

- À partir de `http://www.kernel.org` et vérifier sa signature
  packages `wget`, `coreutils`, `bzip2`
- Les patches : ceux de kernel.org != ceux de la distrib
- Rq : les sources se mettent généralement dans `/usr/src`

Configuration

- Rôle du fichier `.config` et sa réutilisation
- À l'aide des infos précédentes, on est capable de sélectionner les trucs qui «vont bien» :)
  → `make xconfig ou make menuconfig`

Compiler son noyau linux sur mesure - 2/2

Compilation

- `make bzImage`
- `make modules`
- `make modules_install`

Installation

- `mv arch/i386/boot/bzImage /boot/K-NUM`
- Éditer `/etc/lilo.conf` ou `/boot/grub/menu.lst` et changer ce qu'il faut
  - Exemple d'un `/etc/lilo.conf`
- Dans le cas de lilo, exécutez `/sbin/lilo`
- CONSERVEZ PLUSIEURS ENTRÉES !

Exemple : comment les machines de TPs du Nautibus sont gérées ?

Troisième partie

Administration d'une machine GNU/Linux

Retour sur les périphériques

Rappel : un périphérique est adressé par l'intermédiaire d'un fichier

- Périphérique → pseudo-fichier dans `/dev/`
- Son système de fichier doit être ajouté à l'arborescence !
  On parle de montage et point de montage

Monter un périphérique : disque dur, cdrom, usb...

Numérotation des périphériques :

- Périphériques IDE (disques durs, lecteurs multimédia) : hda, hdb, hdc, hdd
- Périphériques scsi : sda, sdb... ; scd..
- Rq : les n° partitions → hda1, hda2 ou sdb1, etc.

Fichiers

- `/etc/fstab` : informations statiques sur le montage des systèmes de fichier
  → ce qui peut être monté, les options et les droits
- Savoir ce qui est monté : `/bin/mount`
  → affiche le contenu de `/etc/mtab`

Retour sur le disque dur

Nécessite dans le noyau : IDE et/ou SCSI

Commandes

- sur partitionnement : `cfdisk, fidsk`
- pour savoir les espaces disques : `df` et `du`
  option `-h` utile

Configurer le disque dur : hdparm

- Commande dangereuse !
- Seules les options -tT permises en TP !
- Permet notamment de configurer taille zone lost+found

VLM : Volume Logical Manager

- Gestion de l'espace disque
- La taille d'une arborescence qui n'est plus dépendante des partitions !
- Plus d'infos : `http://lea-linux.org/leapro/pro_sys/lvm.html`

Les périphériques USB - 1/2

Nécessite dans le noyau : `USB Mass Storage`

Comment monter une clé USB ?

- `dmesg` → donne message à l'introduction d'un périphérique USB
- Faire un mount grâce à cette info

Rq : si un port USB déjà pris, alors device peut changer d'une fois sur l'autre...
→ embêtant pour points de montage statiques déf dans `/etc/fstab` (.. icônes desktop)
FIXME : Donner un EXEMPLE concret de comment on fait !

Solution :

- Utilisation du package `udev`
- Fichiers de configurations dans `/etc/udev/`
- `/etc/usev/rules.d` contient les règles à parser
  → ajout d'un fichier de règles

Les périphériques USB - 2/2

Solution :(suite)

- Dans /var/log/dmesg ou /var/log/syslog

```
date time host kernel : usb 1-1 : new high speed USB device using ehci_hcd and address 2
date time host kernel : scsi0 : SCSI emulation for USB Mass Storage devices
date time host kernel : usb-storage : device found at 2
date time host kernel : usb-storage : waiting for device to settle before scanning
date time host kernel :   Vendor : PHILIPS   Model : CDRW/DVD SCB5265  Rev : TD15
date time host kernel :   Type :   CD-ROM                              ANSI SCSI revision : 00
date time host kernel : Attached scsi generic sg0 at scsi0, channel 0, id 0, lun 0,  type 5
date time host kernel : usb-storage : device scan complete
```

- Utilisation informations «Vendor» et «Model» pour forcer le device
- Fichier de règles contient

```
BUS="scsi", SYSFSmodel="CDRW/DVD SCB5265", NAMEall_partitions="mycdrom"
```

- Réexécution de udev → création de /dev/mycdrom1
- On peut changer /etc/fstab !

Autre Exemple, bien plus complet

http://www.debian-administration.org/articles/126

(Dé)Montage automatique : automount - 1/2

Nécessite options dans le noyau

But : monter automatiquement

- les disques amovibles
- les systèmes de fichiers partagés à travers le réseau

Utilisation d'autofs

- Une partie Noyau, une partie package
- Un démon → `/etc/rc.d/init.d`

Fichiers de configuration

- `/etc/auto.master`

```
misc /etc/auto.misc --timeout=60
```

- `/etc/auto.misc`

```
kernel     -ro,soft,intr ftp.kernel.org :/pub/linux
cdrom      -fstype=iso9660,ro           :/dev/cdrom
floppy     -fstype=vfat                 :/dev/fd0
```

- `/etc/auto.net` : shellscript
- `/etc/auto.smb` : shellscript
- `/etc/default/autofs`

(Dé)Montage automatique : automount - 2/2

Problèmes

- Monte les répertoires à la demande !
  - `ls` dans un rép. géré par automount ne donne pas forcément d'info
    → `cd nomDuRép`... ce qui suppose qu'on connaisse le nom :)
  - Impossibilité d'utiliser des patterns pour accéder aux fichiers
    → `ls /home/e*` ne fonctionne pas
  - `pwd` donne un chemin «aléatoire»
    utilisez la commande Pwd
- Au bout d'un moment d'inutilisation, un répertoire monté est démonté
  - `pwd` peut vous répondre : `.. can't read`
  - `emacs` peut vous dire : `can't write`
  - Une solution : `cd .`

Troisième partie

Administration d'une machine GNU/Linux

Les imprimantes - 1/2

Plusieurs utilitaires dont CUPS (Common Unix Printer System) et lpr

lpr

- Utilise le démon `/usr/sbin/lpd` lancé par le script `/etc/init.d/lpd`
- Default printer «lp», mais variable d'environnement PRINTER
- Désignation de l'imprimante à utiliser : option `-Pprinter`
- Fichier de configuration : `/etc/printcap`
- Les spools : `/var/spool/lpd/printerName$i`

Utilisation en ligne de commande : `lpr -P maPrinter bla.ps`

les imprimantes - 2/2

Fichier de configuration :/etc/printcap

De la manpage...

```
Name  Type    Default          Description
af    str     NULL             name of accounting file
lf    str     /dev/console     error logging file name
lp    str     /dev/lp          local printer device, or port@host for remote
mx    num     1000             max file size (in BUFSIZ blocks) ; 0=unlimited
pc    num     200              price per foot or page in hundredths of cents
pl    num     66               page length (in lines)
pw    num     132              page width (in characters)
rm    str     NULL             machine name for remote printer
rp    str     “lp”             remote printer name argument
rs    bool    false            remote users must have a local account
sd    str     /var/spool/lpd   spool directory
sh    bool    false            suppress printing of burst page header
```

Un exemple de /etc/printcap

```
lp|printerName:\
        :lp=:rm=serveurName.fr:rp=printerName:\
        :sd=/var/spool/lpd/printerName:lf=/var/log/lp-errs:sh:\
        :mx#0:
```

Troisième partie

Administration d'une machine GNU/Linux

L'agent de transport du Mail : MTA

Configuration d'`exim4`

Un client Mail : MUA (*Mail User Agent*)

Il existe de nombreux clients mails !

Textes

- Mutt
- Pine

Graphiques

- Mozilla Thunderbird
- Kmail

Propose

- Gestion des protocoles POP, IMAP et leur version sécurisée
- Possibilité d'envoi du mail direct ou par MTA
- Des filtres

Filtrer les Spams - 1/2

À propos des filtres Bayesiens

FIXME : un peu de théorie sur les filtres bayesiens

Filtrer les Spams - 2/2

Des packages

- Spamassassin (trademark de l'Apache Software Foundation)
  - `spamd` et `spamc`
  - Supporte l'apprentissage bayesien
  - `/etc/default/spamassassin`, `/etc/spamassassin`
- Bogofilter
  -

Remarque : l'usage de ces deux packages est complémentaire

Antivirus

Grâce aux packages `clamav`, `clamsmtp`, `clamcour`, `amavisd`

Fournit

- `clamscan` : scanne les fichiers et les répertoires
- `freshclam` : permet de mettre à jour la base de données
- `clamsmtp` : proxy SMTP scanneur de virus
- `clamcour` : filtre de courrier entrant pour scan clamav
- `amavisd` : interface entre MTA et scanner de virus

Remarque : on privilégiera l'installation de filtres de Spams et Antivirus sur le serveur mail, pour des raisons évidentes de performance

Troisième partie

Administration d'une machine GNU/Linux

Les interfaces réseau

Nécessite des options particulières dans le noyau

Interface loopback

- lo
- Relie la machine à elle-même
- IP associée est 127.0.0.1
- Nom DNS associé est localhost

Autres interfaces

- eth0, eth1...
- ppp0, ppp1...
- wlan0, wlan1...
- irlan0..

Liens intéressants

- `http://www.linux-france.org/prj/inetdoc/cours/config.interface/`
- la manpage accessible via la commande : `man interfaces`

Configuration du réseau

Configuration

- Filaire : `/sbin/ifconfig`
  - Nécessite package : `net-tools`
  - Exemple :

    ```
    ifconfig eth0 192.168.1.1 netmask 255.255.255.0 broadcast 192.168.1.255
    ```
- Wireless : `/usr/sbin/iwconfig`
  - Nécessite package : `wireless-tools`
  - Exemple :

    ```
    iwconfig eth1 essid "ENS" mode managed key open [1] 24CAB
    ```
- `/sbin/route` [add] IPpasserelle gw netmask 255.255.252.0
- Plus facile

  `/usr/sbin/dhclient` si protocole dhcp (cf CCIC2)

Les fichiers de configuration

- `/etc/hosts` : correspondances statiques de noms d'hôtes
- `/etc/networking/interfaces`

Exemple 1 : dynamique avec dhcp

```
auto eth0
iface eth0 inet dhcp
```

Exemple 2 : allocation statique

```
auto eth0
iface eth0 inet static
                    address 192.168.0.12
                    netmask 255.255.255.0
                    gateway 192.168.0.1
```

- `/etc/resolv.conf`

```
search labo.fr
nameserver IP premier serveur DNS
nameserver IP deuxième serveur DNS
```

- `/etc/host.conf`

```
order hosts,bind
multi on
```

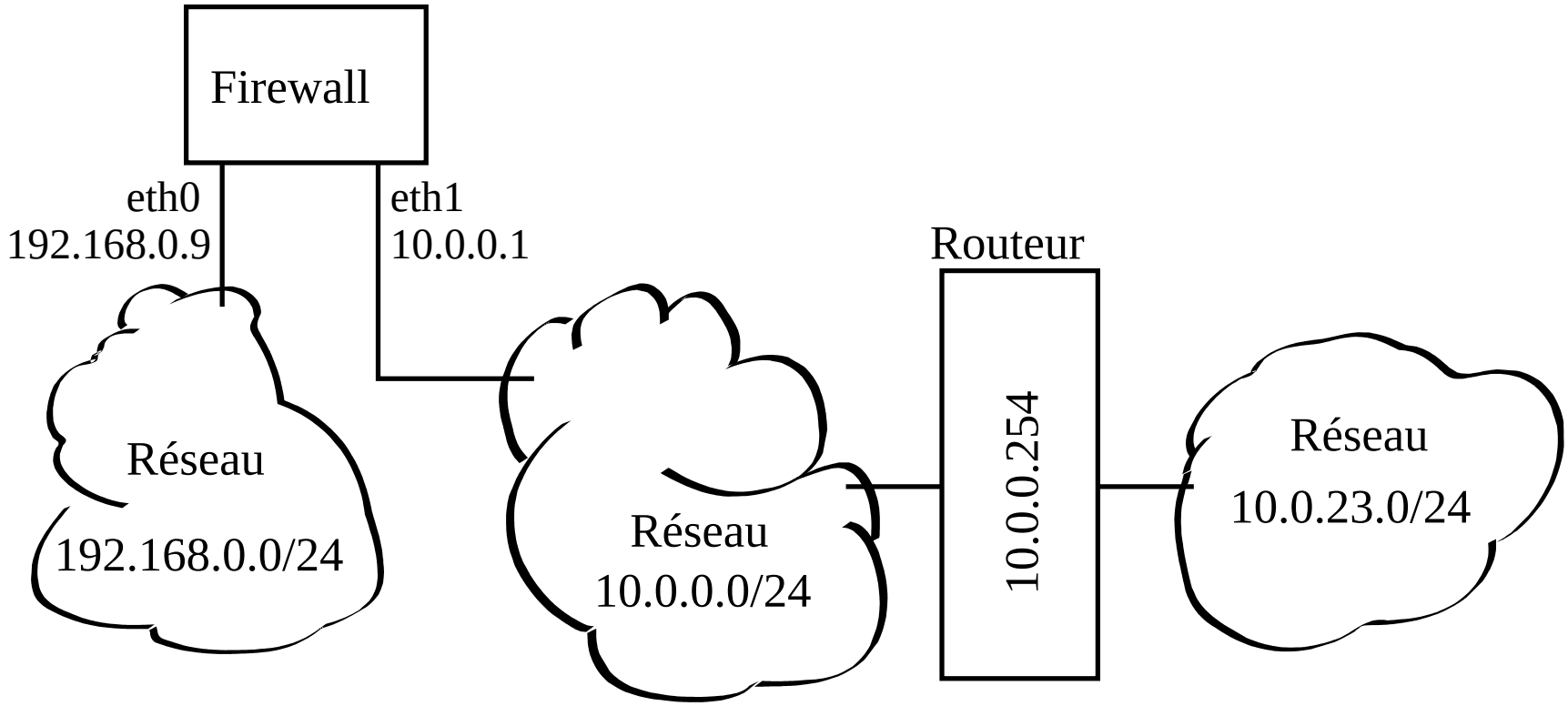
Fichier de configuration sur un exemple - 1/2

Exemple tiré du net[5] : quelles sont les configurations réseau à faire pour le firewall ?
Le firewall est connecté au réseau externe par son interface eth0 dont l'adresse IP est 192.168.0.9, le masque est 255.255.255.0 et dont la route par défaut est 192.168.0.1. L'interface interne eth1 a pour IP 10.0.0.1 avec un masque 255.255.255.0. Un autre réseau interne, 10.0.23.0/24 est situé derrière un routeur interne dont l'IP dans le réseau 10.0.0.0/24 est 10.0.0.254.

[5] http://guillaume.rince.free.fr/spip/article.php?id_article=62

Fichier de configuration sur un exemple - 1/2

Exemple tiré du net[5] : quelles sont les configurations réseau à faire pour le firewall ?
Le firewall est connecté au réseau externe par son interface eth0 dont l'adresse IP est 192.168.0.9, le masque est 255.255.255.0 et dont la route par défaut est 192.168.0.1. L'interface interne eth1 a pour IP 10.0.0.1 avec un masque 255.255.255.0. Un autre réseau interne, 10.0.23.0/24 est situé derrière un routeur interne dont l'IP dans le réseau 10.0.0.0/24 est 10.0.0.254.



[5] http://guillaume.rince.free.fr/spip/article.php?id_article=62

Fichier de configuration sur un exemple - 2/2

/etc/networking/interfaces

```
# The loopback interface
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
       address 192.168.0.9
       netmask 255.255.255.0
       network 192.168.0.0
       broadcast 192.168.0.255
       gateway 192.168.0.1

# The first network card
auto eth1
iface eth1 inet static
       address 10.0.23.1
       netmask 255.255.255.0
       network 10.0.0.0
       broadcast 10.0.0.255

# Routes statiques
up route add -net 10.0.23.0 netmask 255.255.255.0 gw 10.0.0.254
down route del -net 10.0.23.0 netmask 255.255.255.0 gw 10.0.0.254
```

Note : Pour que le routage entre les interfaces eth0 et eth1 devienne effectif, il faut modifier le fichier `/etc/sysctl.conf` en y ajoutant le paramètre : `net/ipv4/ip_forward=1`

Tester le réseau - 1/2

Validation de bon fonctionnement sur un réseau IP

- Adresse IP de l'interface de boucle locale : lo,
- Adresse IP de l'interface du poste de travail : eth0 ou ppp0,
- Adresse IP du destinataire de la passerelle par défaut,
- Adresse IP extérieure au réseau local.

Outils

- `/bin/ping`
  - Validation inter-processus : `ping 127.0.0.1`
  - Fonctionnement interface seule : `ping 192.168.0.2`
  - Communication vers le routeur : `ping 192.168.0.1`
  - Fonctionnement vers l'extérieur : `ping 216.239.59.104`
    - Attention : pas forcément de réponse au ping
    - Question : comment la communication se déroule ici ?
  - Test DNS : `ping google.fr`
    → Si échec, regarder `/etc/resolv.conf`

Tester le réseau - 2/2

Outils - suite

- `/sbin/route` :
  - Connaître l'état de la table de routage de l'hôte
  - Configurer de nouvelles routes
  - N'a rien à voir avec le routage dynamique d'un routeur
    → ne sert qu'à poser des routes statiques entre interfaces.

```
Table de routage IP du noyau
Destination     Passerelle      Genmask         Indic Metric Ref    Use Iface
140.x.y.0       *               255.255.255.0   U     0      0        0 eth0
default         140.x.y.1       0.0.0.0         UG    0      0        0 eth0
```

- `/usr/bin/traceroute`
  - Liste les routeurs traversés pour atteindre un hôte
  - Donne des informations sur la route suivie

$> traceroute google.fr

```
traceroute : Warning : google.fr has multiple addresses ; using 216.239.57.104
traceroute to google.fr (216.239.57.104), 30 hops max, 40 byte packets
 3  193.55.215.26 (193.55.215.26)  1.431 ms  1.543 ms  1.320 ms
 4  lyon-g3-1-14.cssi.renater.fr (193.51.184.130)  1.329 ms  1.304 ms  1.532 ms
11  google-eu-customers-7.GW.opentransit.net (193.251.249.18)  168.032 ms  167.817 ms  171.875
12  66.249.94.10 (66.249.94.10)  165.740 ms  168.448 ms 66.249.94.8 (66.249.94.8)  165.783 ms
13  216.239.47.194 (216.239.47.194)  197.687 ms 216.239.49.170 (216.239.49.170)  166.131 ms  16
14  216.239.49.2 (216.239.49.2)  168.564 ms 216.239.49.93 (216.239.49.93)  170.017 ms  168.397
15  * *
```

Troisième partie

Administration d'une machine GNU/Linux

Les services au démarrage

Description de `/sbin/init` et de `/etc/inittab`

Description des services lancés dans `/etc/init.d`

Liens symboliques dans `/etc/rc$i.d/` où $i

- Default runlevel. The runlevels used by RHS are :
- 0 - halt (Do NOT set initdefault to this)
- 1 - Single user mode
- 2 - Multiuser, without NFS
- 3 - Full multiuser mode
- 4 - unused
- 5 - X11
- 6 - reboot (Do NOT set initdefault to this)

Exemple : à propos de `S20welcome2l`, `S23ntp-server`, `S99kdm`

Troisième partie

Administration d'une machine GNU/Linux

- Se connecter
- Shell et scripts shell
- Installer/configurer des packages sur Debian
- Configuration de la machine
  - Ajout d'utilisateurs/groupes
  - Connaître le hardware
  - Que l'affichage graphique soit !
  - Compiler son noyau
  - Retour sur les périphériques
  - Gestion des imprimantes
  - Le mail
- Configurer le réseau
- Description des services au démarrage
  - Les services au démarrage
  - Jobs synchrones/asynchrones
- Conclusions

Programmer l'exécution de tâches récurrentes - 1/2

Grâce aux démons `cron` et `anacron`

Présentation de `cron`

- Chargé de lancer d'autres programmes de manière périodique et automatique
- Se réveille toutes les minutes, inspecte les tables, recharge si modifiées
- `/etc/crontab`, `/etc/cron.d/`
- `/etc/cron.hourly/`, `/etc/cron.daily/`, `/etc/cron.weekly/`, `/etc/cron.monthly/`
- Chaque utilisateur peut définir sa `crontab` : `crontab -e`
  → définit ainsi les progs à lancer périodiquement

Pour quoi faire ?

- À propos de `/usr/bin/locate` et `/usr/bin/updatedb`
- Pour des sauvegardes !

Programmer l'exécution de tâches récurrentes - 2/2

Exemple : réveil tous les matins à 7h29 en jouant un ogg

```
> crontab -e

29 7 * * *  ogg123 ~/Muzix/song.ogg 1>/dev/null 2>&1
```

Explications

- les minutes
- les heures
- le jour du mois
- le mois
- le jour de la semaine (Lundi=1, Mardi=2, etc.)

Rq : possibilité de définir des intervalles, etc.

- * : a chaque unité de temps
- 2-5 : les unités de temps (2,3,4,5)
- */3 : toutes les 3 unités de temps (0,3,6,...)
- 5,8 : les unités de temps 5 et 8

Troisième partie

Administration d'une machine GNU/Linux

- Se connecter
- Shell et scripts shell
- Installer/configurer des packages sur Debian
- Configuration de la machine
  - Ajout d'utilisateurs/groupes
  - Connaître le hardware
  - Que l'affichage graphique soit !
  - Compiler son noyau
  - Retour sur les périphériques
  - Gestion des imprimantes
  - Le mail
- Configurer le réseau
- Description des services au démarrage
  - Les services au démarrage
  - Jobs synchrones/asynchrones
- Conclusions

1er bilan

Globalement, on sait comment installer/administrer une machine...

- Le démarrage de la machine et les initscripts
- Les mécanismes relatifs à la connexion (console et graphique)
- Des notions sur les shells et sur les scripts shell
- Comment gérer les utilisateurs, quelques périphériques...
- Configuration réseau
- Tâches périodiques
- Superviser via les logs du système

Rien de concret encore sur la sécurité !

Exemple de résultat de la commande `lspci`

```
0000:00:00.0 Host bridge: Intel Corporation Mobile 915GM/PM/GMS/910GML Express Processor to DRAM
Controller (rev 03)
0000:00:02.0 VGA compatible controller: Intel Corporation Mobile 915GM/GMS/910GML Express Graphics
Controller (rev 03)
0000:00:02.1 Display controller: Intel Corporation Mobile 915GM/GMS/910GML Express Graphics Contro
(rev 03)
0000:00:1c.0 PCI bridge: Intel Corporation 82801FB/FBM/FR/FW/FRW (ICH6 Family) PCI Express Port 1
03)
0000:00:1d.0 USB Controller: Intel Corporation 82801FB/FBM/FR/FW/FRW (ICH6 Family) USB UHCI #1 (re
03)
0000:00:1d.1 USB Controller: Intel Corporation 82801FB/FBM/FR/FW/FRW (ICH6 Family) USB UHCI #2 (re
03)
0000:00:1e.0 PCI bridge: Intel Corporation 82801 Mobile PCI Bridge (rev d3)
0000:00:1e.2 Multimedia audio controller: Intel Corporation 82801FB/FBM/FR/FW/FRW (ICH6 Family) AC
Audio Controller (rev 03)
0000:00:1f.0 ISA bridge: Intel Corporation 82801FBM (ICH6M) LPC Interface Bridge (rev 03)
0000:00:1f.1 IDE interface: Intel Corporation 82801FB/FBM/FR/FW/FRW (ICH6 Family) IDE Controller (
03)
0000:00:1f.3 SMBus: Intel Corporation 82801FB/FBM/FR/FW/FRW (ICH6 Family) SMBus Controller (rev 03
0000:01:00.0 Ethernet controller: Broadcom Corporation NetXtreme BCM5751 Gigabit Ethernet PCI Expr
(rev 01)
0000:02:01.0 CardBus bridge: Texas Instruments PCI6515 Cardbus Controller
0000:02:01.5 Communication controller: Texas Instruments PCI6515 SmartCard Controller
0000:02:03.0 Network controller: Intel Corporation PRO/Wireless 2200BG (rev 05)
```

retour

Quatrième partie

Admin. d'un parc de machines GNU/Linux

- Présentation Générale
- Network File System : NFS
- Network Information Server : NIS
- Domain Name Server : DNS
- Lightweight Directory Access Protocol : LDAP
- Installer un proxy avec Squid
- Virtual Private Network : VPN
- Le protocole DHCP
- Concurrent Version System : CVS
- Conclusions

De nouveaux problèmes ?

Quelles sont les différences importantes avec ce qu'on a vu ?

On veut

- Que les données soient accessibles depuis chaque machine
  → NFS (*Network File System*)
- Que seules les personnes autorisées puissent se connecter
  → NIS (*Network Information Service*)
- Qu'on puisse contacter une machine du parc par son nom
  → DNS (*Domain Name Server*)
- Disposer d'un annuaire sécurisé consultable par tous
  → LDAP (*Lightweight Directory Access Protocol*)
- Qu'une machine se connecte « automatiquement » au réseau
  → Utilisation du protocole dhcp
- ... et seulement si elle est autorisée à le faire

Un air de déjà vu

En CCIR5, vous avez vu en particulier

- Les mécanismes du portmapper
- Le NFS
- Le DNS

→ Sur ces notions, un bref rappel

Quatrième partie

Admin. d'un parc de machines GNU/Linux

- Présentation Générale
- Network File System : NFS
- Network Information Server : NIS
- Domain Name Server : DNS
- Lightweight Directory Access Protocol : LDAP
- Installer un proxy avec Squid
- Virtual Private Network : VPN
- Le protocole DHCP
- Concurrent Version System : CVS
- Conclusions

Accéder aux fichiers distants

NFS

Objectifs

- Avoir un système de fichiers distants
- Accès transparent à ses données pour l'utilisateur

  → Tout se passe comme en local (`ls`, `mv`, etc.) !

Pourquoi faire ?

- Pouvoir accéder à ses données de plusieurs endroits
- Administration

Protocoles existants

- NFS : *Network File System*
- SMB : *Server Message Block*
- OpenAFS, Coda, InterMezzo

D'autres infos : `http://www.linux-france.org/prj/inetdoc/cours/admin.reseau.nfs/`

Configuration côté server

Dans /etc/exportfs

- Les partitions à exporter
- Par exemple

/etc/exportfs

```
/home 192.168.0.0/255.255.255.0 (rw,sync,all_squash,anonuid=1232,anongid=1322)
```

- Que fait la commande `exportf -rv` ?

Dans /etc/hosts.deny et /etc/hosts.allow

- Décrivent les services autorisés/interdits et pour quelles machines
- Exemple

/etc/hosts.deny

```
ALL: ALL
```

/etc/hosts.allow

```
portmap mountd nfsd rsynd: 192.168. 10. 132.28.4.
sshd: 192.168. 10. 132.28.4. 132.28.18.
```

- `man hosts_access` pour la syntaxe du langage de contrôle d'accès

Lancer le script : /etc/init.d/nfs-kernel-server start

Rq : ce script n'existe pas sur toutes les distributions : sur debian, que fait-il ?

Configuration côté client

Dans /etc/fstab

- Contient les infos statiques sur les partitions montables
- Exemple

/etc/fstab

```
/dev/hda6           none     swap     sw                          0     0
/dev/hda5           /        ext3     defaults,errors=remount-ro  0     1
192.168.0.1:/home   /users   nfs      defaults                    0     0
```

Créer les répertoires qui seront montés

Tester l'accès au serveur et les partitions exportées

- `rpcinfo -u 192.168.0.1 nfs`
- `showmount -e 192.168.0.1`

Lancer le script : `/etc/init.d/nfs-common start`

Rq : ce script n'existe pas sur toutes les distributions : sur debian, que fait-il ?

→ Il faut encore tester les droits d'accès et les droits en lecture/écriture !

NFS et la sécurité - 1/2

À propos d'identifiant

- Les accès ont lieu avec les uid et gid de l'utilisateur sur la machine cliente
  Rq : sauf si l'option all_squash est précisée
  → Pas forcément la même ID que sur le serveur !
- Problème pour root...
  → Par défaut, `root` devient `nobody`
  → Possibilité de préciser l'option `no_root_squash`

Solutions

- Donner le même identifiant sur les machines clientes que sur serveur
  → Impossible ou trop lourd !
- NIS qui sera vu plus loin

NFS et la sécurité - 2/2

Ne pas croire forcément tout ce qu'il y a dans un cours ! :)

Pourquoi ?

→ Nouvelle version de NFS, donc nouvelles fonctionnalités

NFSv4

- Depuis peu de temps mais déjà utilisée
- Pour le partage via Internet ?
- Techno de cache agressive
- Regroupement des requêtes réseau (*Compound request*)
- Sécurisation négociée et chiffrement des données : Kerberos 5, Certificats (SPKM), Clefs publiques/privées (LIPKEY)
- Capacité pour les clients de maintenir des sessions ou de les récupérer malgré crash serveur ou panne du réseau
- Support d'attributs fichier nommés par le user

Coda

Une alternative à NFS : Coda version 6.0.12 le 21 sept. 2005

- Évolution d'AFS
- Date de 1987
- Système de fichier distribué

  → donc plusieurs serveurs répliqués
- Utilise un système de cache
- Supporte les déconnexions volontaires ou non !
  - Panne serveurs
  - Panne réseau

  → Seule les données des serveurs manquants

  → Réintègre les modifications lors de la reconnexion
- Sécurité
  - Authentification des utilisateurs
  - ACL (*Access Control List*) permettent une gestion plus fine que les droits UNIX

Dans le noyau Linux !

Quatrième partie

Admin. d'un parc de machines GNU/Linux

- Présentation Générale
- Network File System : NFS
- Network Information Server : NIS
- Domain Name Server : DNS
- Lightweight Directory Access Protocol : LDAP
- Installer un proxy avec Squid
- Virtual Private Network : VPN
- Le protocole DHCP
- Concurrent Version System : CVS
- Conclusions

Gestion des utilisateurs distants

Network Information Server : NIS

- Introduit par Sun en 1985 (*Yellow Pages*, `yp` à l'origine)
- Pas un standard mais largement répandu
- Base de données distribuées pour le partage d'infos système sur le réseau comme les login, passwords, répertoires de connexion et infos sur les groupes càd, `/etc/passwd`, `/etc/hosts`, `/etc/groups`, etc.

Pourquoi faire ?

- Réduire le temps d'admin. d'un parc de machine !
- Simplifier la gestion des comptes, mots de passes, etc.
- Exemple : la création d'un utilisateur sur le serveur NIS permet à chaque machine du parc d'avoir accès à ses infos `login`

Rq : le serveur NIS n'est pas nécessairement le serveur NFS !

Architecture de NIS

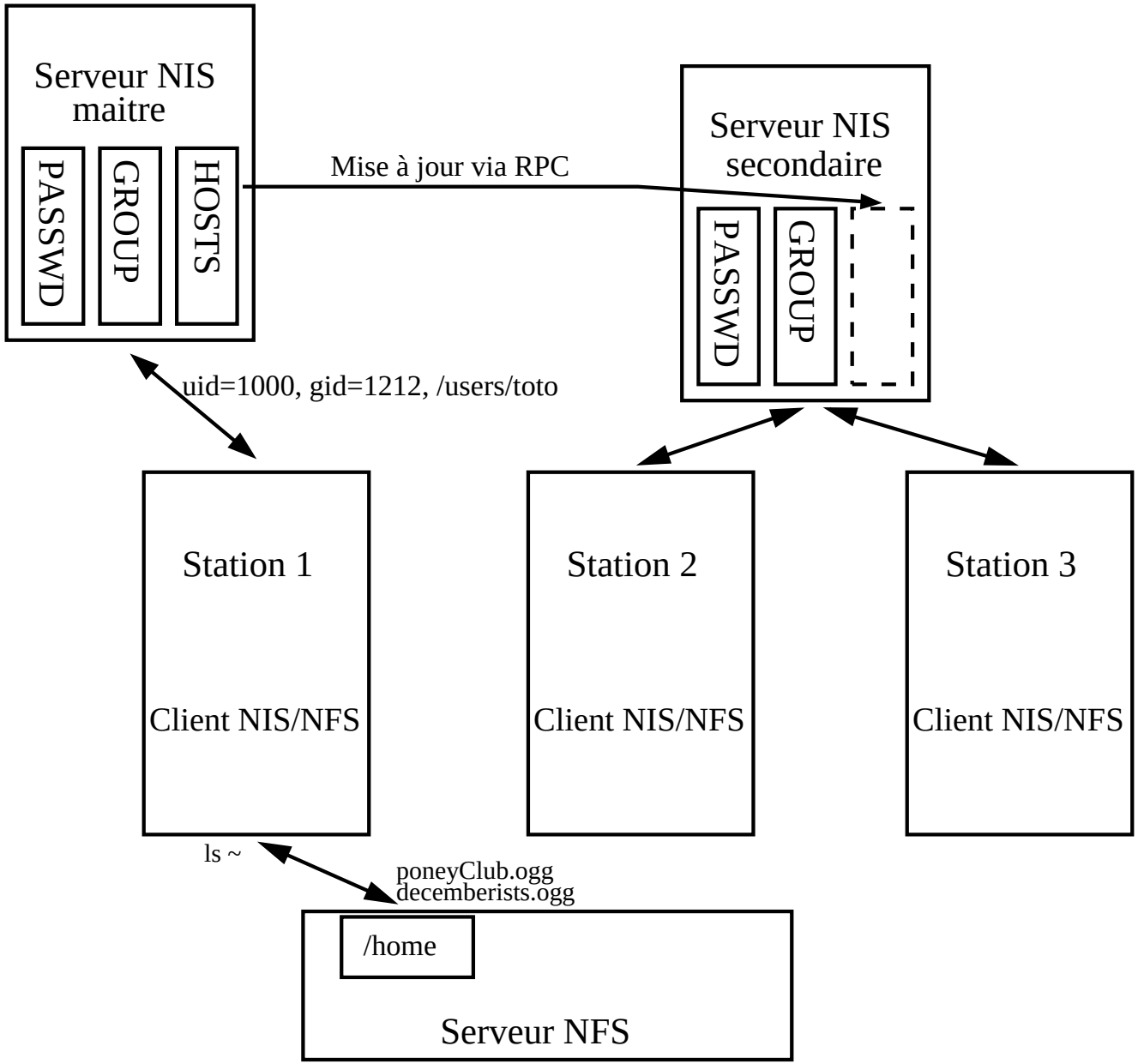
Découpage en domaines

- Selon un modèle Client/Serveur au dessus des SUN-RPC
- Où chaque domaine comprend
  - Un serveur maître qui gère les « maps » (ou cartes)
    (des informations contenue dans la base de données)
  - [0-9]* serveurs esclaves jouant le rôle de serveurs secondaires
    (des réplicats consultables pour diminuer la charge)
  - Des clients NIS qui consultent les serveurs NIS (maître ou secondaires)

RQ :

- Les esclaves diffusent les maps sans pouvoir les modifier
  → Sinon problème de cohérence
- Seul le serveur maître peut modifier une map !

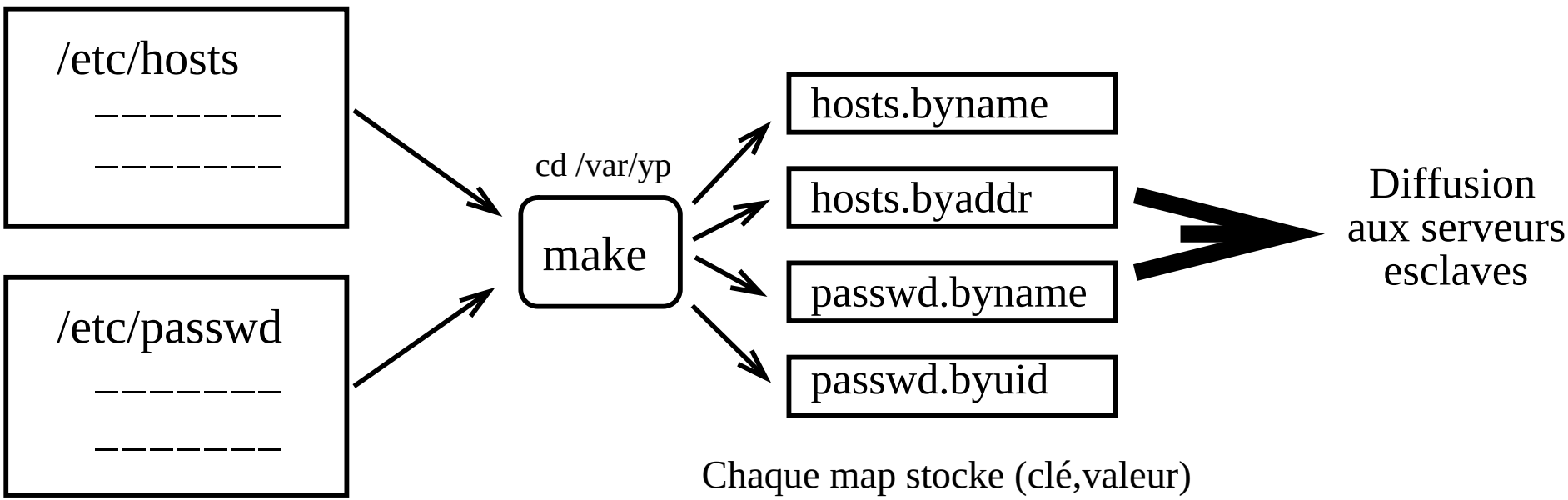
Schéma de fonctionnement



Fonctionnement

- Maps stockées sur le serveur dans `/var/yp/nomDeDomaine`
  RQ : Un répertoire `/var/yp` doit exister sur le client !

- Quand le fichier source d'une map est modifiée (donc sur le serveur)
  - Regénérer la map correspondante : `make -C /var/yp`
  - Propoger les changements aux serveurs esclaves grâce à la commande `yppush`



NIS côté client - 1/2

Configuration du client NIS - en direct

- Utilise les programmes clients : `ypbind`, `ypwhich`, `ypcat`, `yppoll`, `ypmatch`
- *Binding* pour interroger le serveur NIS
  - Renseigner le nom de domaine avec `domainname` ou dans `/etc/defaultdomain`
  - Désigner explicitement le serveur, pas de broadcast (sécurité !)
- `ypbind` doit toujours tourner : recherche régulièrement le serveur NIS
  - `ypbind -broadcast` : non ! Pas sécure...
  - `ypset nomServeurNIS ; ypbind`
  - `ypwhich` permet d'afficher le nom du serveur NIS

RQ :

- un client NIS peut voir le contenu d'une map grâce à `ypcat`
- S'assurer que `/sbin/portmap/` est lancé et que `/var/yp/` existe

NIS côté client - 2/2

Configuration du client NIS - via les fichiers de conf

- Fichier `/etc/yp.conf`

/etc/yp.conf

```
# plusieurs entrée de ce genre sont possibles
domain grid5000 server 192.168.69.2
```

- Ajouter en fin de chaque fichier qu'il faut consulter les maps associées

```
root@192.168.69.2#> tail -1 /etc/passwd
+::::::
root@192.168.69.2#> tail -1 /etc/group
+::::
```

- Mettre l'option `compat` dans `/etc/nsswitch.conf`

RQ :

- S'assurer que `/sbin/portmap/` est lancé et que `/var/yp/` existe
- `/etc/init.d/nis start`

NIS côté serveur - 1/2

Un serveur maître NIS fait tourner

- `ypserv` pour répondre aux requêtes clientes
- `ypbind` s'il est lui-même client NIS
- `ypxfrd` pour répondre aux requêtes de màj des maps par les esclaves
- `rpc.yppasswdd` pour assurer les demandes de changement de password

Installation et exécution

- `apt-get install nis` qui lance les services NIS
  → on les arrête puisque rien n'est configuré encore...
- Relancer le portmapper
- Renseigner la variable `NISSERVER` dans `/etc/default/nis`
- `ypinit -m` sur le serveur maître pour initialiser NIS
- `ypinit -s masterServer` pour les serveurs esclaves
- `/etc/init.d/nis start` pour lancer NIS

RQ : Un serveur NIS esclave fait tourner

- `ypserv` pour répondre aux requêtes clientes
- `ypbind` s'il est lui-même client NIS

NIS côté serveur - 2/2

À propos des netgroups

- Système NIS permet de définir groupes de machines ou d'utilisateurs
  → Permet d'autoriser/interdire accès à une ressource
- Défini dans `/etc/netgroup` qui constitue une map
- Groupe défini par liste de triplets (`machine,user,nisDomain`)

Exemple d'utilisation des netgroups

```
root@192.168.90.2#> cat /etc/netgroup
mes_mach (192.168.69.1,,grid5000) (192.168.69.2,,grid5000)
net_admins (,ogluck,) (,root,)
mes_users (,toto,) (,titi,)

root@192.168.90.1#> cat /etc/exports
# J'autorise mes_machines à monter /home...
/home @mes_mash (rw,root_squash,async)

root@192.168.90.2#> tail -1 /etc/passwd
# Je rejette les lignes de la map pour les users titi et toto
-@mes_users :
# J'accepte les lignes de la map passwd pour root et pour ogluck
+@net_admins::::::
```

Quelques remarques - 1/2

Pour s'assurer que tout fonctionne

- Services démarrés

```
root@192.168.69.2#> rpcinfo -u 192.168.90.2 ypserv
program 100004 version 1 ready and waiting
program 100004 version 2 ready and waiting

root@192.168.69.2#> rpcinfo -u 192.168.69.2 ypbind
program 100007 version 1 ready and waiting
program 100007 version 2 ready and waiting

root@192.168.69.2#> ypwhich
192.168.90.2
```

- Contrôle de l'accès au serveur NIS

```
root@192.168.90.2#> cat /etc/ypserv.securenets
# This file defines the access to your NIS server
255.0.0.0       127.0.0.0
255.255.255.0   192.168.69.0
```

- Et faire quelques tests pour être sûr que les droits sont ok, etc.

Quelques remarques - 2/2

Défauts des NIS

- Pas d'authentification des clients NIS :
  → Connaître le nom de domaine suffit pour interroger le serveur
  → Possibilité de connaître le contenu des maps !
- Maps transmises en totalités mêmes si légères modifications de leur contenu
- En ce moment, cassé dans debian...
- Shadow passwords sur NIS douteux voire inexistant
- Un peu vieux maintenant... → LDAP

→ Voir le HOWTO : `http://www.linux-nis.org/nis-howto/HOWTO/`
et la synthèse NFS-NIS :
`http://www.linux-france.org/prj/inetdoc/cours/admin.reseau.synthese-nfs-nis/`

Quatrième partie

Admin. d'un parc de machines GNU/Linux

- Présentation Générale
- Network File System : NFS
- Network Information Server : NIS
- Domain Name Server : DNS
- Lightweight Directory Access Protocol : LDAP
- Installer un proxy avec Squid
- Virtual Private Network : VPN
- Le protocole DHCP
- Concurrent Version System : CVS
- Conclusions

S'appeler par son petit nom

Objectif : assurer la correspondance @IP $\Leftrightarrow$ nom d'hôte

Définition

- Base de données distribuée sur une hiérarchie de serveurs de noms
  - Car Internet trop gd, trop de requêtes
  - Tolérance aux pannes... une panne, tout le net tombe :)
  - Délais de réponse
  - Mises à jour continuelles de la base
- Protocole applicatif, modèle client/serveur

Et ?

- Permet le *Host aliasing*
- Permet le *Mail serveur aliasing* : seveur mail et web ont même pseudo avec IP différente
- Permet répartition de charge : rotation d'@IP du serveur DNS

Fonctionnement

Remarques

- Si un serveur n'a pas les données dans sa base, contacte un autre serveur
- Trois types de serveurs DNS
  - Serveurs de noms locaux en charge de la résolution des requêtes locales
  - Serveurs de noms racine pour propager la requête
  - Serveurs de noms de source autorisée, contenant les corresp. « officielles »

Schéma

FIXME : Schéma

La suite

Sur les slides d'Olivier

Remarques : Et pour de la doc, faire un tour sur le site **commercial** : `http://guides.ovh.com/`.
Vous y trouverez des infos complémentaires, y compris sur la configuration de votre machine : `http://guides.ovh.net/DomaineFrProblemesZoneCheck/contenu.html`

Remarque sur la configuration

Dans `named.conf`, ajouter sa zone et résolution inverse par ex.[6]

```
zone "linux.bogus" {
   notify no ;
   type master ;
   file "linux.bogux" ;
} ;
```

```
zone "196.168.192.in-addr.arpa" {
   notify no ;
   type master ;
   file "db.192.168.196" ;
} ;
```

Avec les fichiers en conséquence correctement remplis, par ex.

```
@ IN SOA ns.linux.bogus.
hostmaster.linux.bogus. (
           199802151 ; serial
           8H ; refresh, seconds
           2H ; retry, seconds
           1W ; expire, seconds
           1D ) ; minimum, seconds
;
  NS ns ; Inet Address of name server
  MX 10 mail.linux.bogus
  MX 20 mail.friend.bogus.
;
localhost A 127.0.0.1
ns A 192.168.196.2
mail A 192.168.196.4
```

```
@ IN SOA ns.linux.bogus.
hostmaster.linux.bogus. (
           199802151 ; Serial
           8H ; Refresh
           2H ; Retry
           1W ; Expire
           1D) ; Minimum TTL
  NS ns.linux.bogus.

1 PTR gw.linux.bogus.
2 PTR ns.linux.bogus.
3 PTR donald.linux.bogus.
```

[6] Voir le **DNS-HOWTO** : `http://www.tldp.org/HOWTO/DNS-HOWTO.html`

Quatrième partie

Admin. d'un parc de machines GNU/Linux

- Présentation Générale
- Network File System : NFS
- Network Information Server : NIS
- Domain Name Server : DNS
- Lightweight Directory Access Protocol : LDAP
- Installer un proxy avec Squid
- Virtual Private Network : VPN
- Le protocole DHCP
- Concurrent Version System : CVS
- Conclusions

Ranger des infos dans un annuaire distribué

Le protocole LDAP

- Norme définissant comment les infos sont échangées entre client et serveur
- Définit la manière dont les données sont représentées

Pour la suite

LDAP est considéré comme non vu cette année

Quatrième partie

Admin. d'un parc de machines GNU/Linux

- Présentation Générale
- Network File System : NFS
- Network Information Server : NIS
- Domain Name Server : DNS
- Lightweight Directory Access Protocol : LDAP
- Installer un proxy avec Squid
- Virtual Private Network : VPN
- Le protocole DHCP
- Concurrent Version System : CVS
- Conclusions

Un hublot vers Internet

Objectif : faire tampon entre le réseau local et le net

- Proxy HTTP, FTP, ... → proxy applicatif !
- Serveur mandataire
  → mandaté par une application pour faire la requête sur Internet à sa place
  ... exmple d'un service d'abonnement sur identité du proxy
- Donc aussi
  ⇒ Empêcher les machines de se connecter « librement » sur le net
  ⇒ Empêcher un attaquant de dresser la carte du réseau interne
  ⇒ Mais souvent remplacé par routeur/firewall...

Fonctionnalités

- Fonction de cache : stocker temporairement
  ⇒ économiser bande passante et minimiser temps de réponse
- Filtrage : *logs* sur les requêtes par user + filtres sur requêtes, sites, contenus...
- Authentification
- *Reverse-Proxy* : accès à un site interne de l'extérieur

Existants : plusieurs packages... .. → Squid !

Configurer Squid

Squid et sa configuration ne seront pas vus cette année...

Quatrième partie

Admin. d'un parc de machines GNU/Linux

- Présentation Générale
- Network File System : NFS
- Network Information Server : NIS
- Domain Name Server : DNS
- Lightweight Directory Access Protocol : LDAP
- Installer un proxy avec Squid
- Virtual Private Network : VPN
- Le protocole DHCP
- Concurrent Version System : CVS
- Conclusions

Comme si on y était

Mais pas cette année...

Quatrième partie

Admin. d'un parc de machines GNU/Linux

- Présentation Générale
- Network File System : NFS
- Network Information Server : NIS
- Domain Name Server : DNS
- Lightweight Directory Access Protocol : LDAP
- Installer un proxy avec Squid
- Virtual Private Network : VPN
- Le protocole DHCP
- Concurrent Version System : CVS
- Conclusions

Brancher, c'est connecté !

Dynamic Host Configuration Protocol

- Protocole client/serveur
- Configurer les postes clients de façon automatique
- Idéal sur un LAN
- Le protocole derrière bcp de MHbox des providers

Fonctionnalités

- Donne une adresse IP unique à un client
- Les bonnes infos concernant le réseau
  - Renseigne sur le masque de sous réseau
  - Adresses des serveurs primaires et secondaires DNS
  - L'IP de la passerelle permettant d'accéder au net

Fonctionnement

- Client (0.0.0.0) envoie DHCPDISCOVER en broadcast
- Serveur retourne DHCPOFFER
- Client envoie DHCPREQUEST en broadcast contenant IDserveur_dhcp
- Serveur envoie DHCPACK

Configuration du serveur - 1/2

Le démon

- Dispose d'un intervalle d'adresses pour l'allocation dynamique
- Donne un bail reconductible au client
- Possibilité de gérer des réseaux logiques (avec relais sur les routeurs grâce à `/etc/init.d/dhcrelay`)
- Lancé par le script `/etc/init.d/dhcpd`
  → À éditer pour indiquer quelles interfaces réseau écouter via la variable `INTERFACES`

Remarques

- Fichier de configuration `/etc/dhcpd.conf`
- Option `allow unknown-clients` mais peut n'accepter que @MAC données
- Option `router` pour spécifier la passerelle par défaut
- Notion de variables globales
- Lancez `/etc/init.d/dhcpd restart` après modif

Pour plus d'infos, un site vraiment bien fait :
`http://christian.caleca.free.fr/dhcp/`

Configuration du serveur - 2/2

/etc/dhcpd.conf

```
ddns-update-style none ;
ddns-updates off ;

# toutes les addr MAC acceptées
allow unknown-clients ;

# Durée de vie du bail
max-lease-time 360000 ;
default-lease-time 360000 ;

# Infos à donner aux clients
option domain-name-servers 192.168.0.1 ;


option routers 192.168.0.1 ;
```

```
subnet 192.168.0.0 netmask 255.255.255.0 {
     # default gateway
     option routers 192.168.0.1 ;
     option subnet-mask 255.255.255.0 ;

     # Seting up an ip address is better here
     # option domain-name-servers ns.domain.org ;
     option domain-name-servers 80.10.246.1 ;
     option domain-name-servers 80.10.246.132 ;

     range dynamic-bootp 192.168.0.10 192.168.0.20 ;

     # default-lease-time 21600 ;
     # max-lease-time 43200 ;
     # we want the nameserver to appear at a fixed
     # address

    host corba
     {
     hardware ethernet 00:20:18:24:41:19 ;
     fixed-address 192.168.0.21 ;
     }
}
```

Configuration du client

/sbin/dhclient

- Il existe plusieurs clients : pump, dhcpxd, dhcpcd ou dhclient
- Fichier de configuration /etc/dhclient.conf
- Des scripts exécutés automatiquement avant et après obtention du bail
- Historiques des baux dans /var/lib/dhcp/dhclient.leases
- Adresse serveur conservée et recontactée lors de prochaine requête
- Fichier /etc/network/interfaces (cf exemple) ;

/var/lib/dhcp/dhclient.leases

```
lease {
interface "eth0" ;
fixed-address 140.77.13.46 ;
option subnet-mask 255.255.255.0 ;
option routers 140.77.13.1 ;
option domain-name-servers
140.77.1.32,140.77.1.183 ;
option domain-name "cri2000.ens-lyon.fr" ;
```

```
option dhcp-lease-time 7200 ;
option dhcp-message-type 5 ;
option dhcp-server-identifier 140.77.1.183 ;
option dhcp-renewal-time 3600 ;
option dhcp-rebinding-time 6300 ;
renew 1 2006/1/16 16 :54 :16 ;
rebind 1 2006/1/16 17 :39 :16 ;
expire 1 2006/1/16 17 :54 :16 ;
}
```

Quatrième partie

Admin. d'un parc de machines GNU/Linux

- Présentation Générale
- Network File System : NFS
- Network Information Server : NIS
- Domain Name Server : DNS
- Lightweight Directory Access Protocol : LDAP
- Installer un proxy avec Squid
- Virtual Private Network : VPN
- Le protocole DHCP
- Concurrent Version System : CVS
- Conclusions

Un coin de paradis pour les développeurs

Développeurs au sens large

- Code
- Livres, articles...

Objectifs

- Permettre un travail collaboratif
- Stocker les versions successives pour chaque modification
- Ne stocker que les différences pour gagner de l'espace

Programmes alternatifs

- Front-ends graphiques : `Cervisia`
- Subversion (v 1.3.0), LibreSource
- Code : Git (linux), `BitKeeper` (propriétaire, de BitMover)

Fonctionnement

Où

- Un machine serveur « backupée »
- ... contenant l'archive CVS

Les commandes

- Charger un projet
- Ajouter un projet
- Ajouter un fichier
- Dans un projet, un `CVROOT`

De nombreuses documentations sur le Oueb !

- `http://www.nongnu.org/cvs/`, le site de référence
- `http://www.idealx.org/doc/cvs.fr.html`

Fonctionnement

Exemple pour ajouter un projet

- Au tableau

Des sites où déposer les projets

- sourceforge.net
- savannah

Quatrième partie

Admin. d'un parc de machines GNU/Linux

- Présentation Générale
- Network File System : NFS
- Network Information Server : NIS
- Domain Name Server : DNS
- Lightweight Directory Access Protocol : LDAP
- Installer un proxy avec Squid
- Virtual Private Network : VPN
- Le protocole DHCP
- Concurrent Version System : CVS
- Conclusions

Conclusion

Je n'ai pas parlé de

- SAMBA et de réseaux hétérogènes (`apt-cache search` + `man`)
- Voisinage réseau LAN
  $\longrightarrow$ voir le service d'information LISA : "help:lisa" dans konqueror et à l'URL :

  `http://www.linux-france.org/prj/inetdoc/cours/admin.reseau.neigh/admin.reseau.neigh.lisa.html`

- et de plein d'autres choses encore...

Par contre, j'ai abordé

- Les MTA, imap et pop, smtp et j'en oublie

Cinquième partie

Sécuriser un système

- Introduction
- Sauvegarder le système
  - Sauvegarder les données
  - Sauvegarder une installation
- Sécuriser son système
  - Un système sûr
  - De l'éducation des utilisateurs
- Les communications ou la sécurité orientée systèmes distribués et réseau
  - Être sur ses gardes
  - D'un point de vue utilisateur
  - D'un point de vue administrateur
- Conclusions

Qu'est ce qu'on entend par là ?

Se prémunir d'erreur : crash matériel, erreur de manipulation

Se prémunir des intrusions

- Savoir où sont les failles pour les prévenir
- Une charte du bon usage des ressources informatiques à faire signer
- Un parc de machines à jour
- Limiter les utilisateurs et les permissions sur le système
- Décider de, et se limiter aux services utiles
  - Désactivation/désinstallation de services
  - Ajout de filtres *Firewall* (ou pare-feu) ou de tcpwrappers
  - Consolider les services pour limiter l'impact sur le système si intrusion
- Mise en place d'outils de détection d'utilisation non autorisée

- Un local technique qui ferme à clé...

Cinquième partie

Sécuriser un système

Sauvegardes et outils de sauvegarde

2 types de sauvegardes :

- sauvegardes régulières → (ana)cron job
  → Sauvegarde des fichiers du `home` NFS une fois par jour sur bandes
  → Conservation d'au moins un mois de *backups*
  → Attention où sont entreposées les bandes
- sauvegardes pour restauration après un crash → ghosts

Mais cela commence dès l'installation, où il est sage de

- Faire un check du disque dur
- Faire un check de la RAM avec memtest86(+)

Cinquième partie

Sécuriser un système

Images Ghosts

Plusieurs utilitaires disponibles

- `partimage` pour sauvegarder des partitions en fichiers compressés
- `mondorescue`[7] : `mindi` et `mondo`
  - Pour créer disque boot/root fondé sur le système
  - Pour sauvegarder des partitions : de très nbx systèmes de fichiers supportés (dont ext2/3, reiser, xfs, jfs, vfat, ntfs, nfs, smbfs)
  - Permet de récupérer intégralement les données de la machine

[7] `http://www.mondorescue.org/`

Cinquième partie

Sécuriser un système

Surveiller son système : lire les logs

`/var/log/syslog` :

- Fichier de log principal
- Contient tous les messages du noyau (aussi dans kernel.log)
- Contient tous les messages des serveurs (aussi dans daemon.log)
- Contient tous les messages de la cron...

`/var/log/auth.log` : raconte tout ce qui concerne les authentifications

Craquer une machine au boot

Si accès physique

- Si accès au BIOS, changement ordre de boot → mot de passe !
  - Le rescue disc ! (D7 ou CD)
  - La clé USB
  - Network (il faut avoir une autre machine et un câble)
- Le single user !
  → Option failsafe au démarrage proposé par Lilo peut permettre d'obtenir les accès root (sans mot de passe) pour la maintenance du système

→ Accès à la machine dangereux

d'où l'utilisation de clients légers

Cadenasser les tours

- Vol
- Ajout de matériel

Avoir un disque de secours

Disquette

- tomsrtbt[8]
  - Bien documenté
  - Comprend de nombreux outils utiles

CDROM

- Les CDs d'installation (option rescue ou demander un shell)
- Les distributions CD : Knoppix, hackin9 live (Aurox)

[8] `http://www.toms.net/rb`

La sécurité dès l'installation

Mettre en place le « shadowing »[9]

- Users et mots de passe sont (étaient) stockés dans `/etc/passwd` encodé
  Fichier lisible par tout le monde : uid et gid !
  Utilisation de `crack` ou `john the ripper`

```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:
username:passwd:UID:GID:full_name:directory:shell
```

- Plus de sécurité : infos sur users et mots de passe dans `/etc/shadow` chiffré

```
root:HDJIKW1.PA:11015:0::7:7::
username:passwd:last:may:must:warn:expire:disable:reserved
```

D'autres choix importants

- Utiliser le protocole SSH2 (vf plus loin)
  → Potentiellement en refusant tout accès par défis-réponse, i.e., authentification par clé
- Utilisation d'`apt-key` dans Debian (cf plus loin)

[9] `http://www.unixtech.be/docs/howtos/Shadow-Password-HOWTO-2.html`

Maintenir le système à jour

Mises à jour de sécurité Debian :

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

S'informer de ce qui est

Il existe plusieurs sites oueb qu'il faut parcourir, des mailings lists auxquelles il faut s'abonner.

- `http://www.insecure.org/`

FIXME : donner autres sites sur sécurité (rapport de bugs, de failles, etc.)

Utilisation de partitions chiffrées

Nécessite dans le noyau les *loop device*

Mise en place

- Partionnenemt avec `cfdisk`
- Mise en place du pseudo-device `/dev/loop0`
  `losetup -e algoChiffrement /dev/loop0 periph`
- Formatage
  `mkfs -t type /dev/loop0`

Accès en lecture

- Passer par `/dev/loop0` pour lire la partition
  `losetup -e algoChiffrement /dev/loop0 periph`
- Monter la partition sur un répertoire existant
  `mount -t type /dev/loop0 /mnt/partChiffrement`
- Les accès sont transparents
- Pour terminer, `umount` et `losetup -d /dev/loop0`

Cinquième partie

Sécuriser un système

Exemple d'une simple gêne

Les systèmes réels ont des failles

Les systèmes ne protègent pas de tout

- `while true ; do mkdir toto ; cd toto ; done` (en shell)
- `while(1) { fork() ; }` (en C)
- `while(1) { char *a=malloc(512) ; *a='1' ; }` (en C)

Réponse classique de l'OS : gel (voire pire)
On suppose que les utilisateurs ne sont pas mal intentionnés (erreur ?)

> *Unix was not designed to stop people from doing stupid things, because that would also stop them from doing clever things.*
> – Doug Gwyn

Deux types de solutions

Technique : mise en place de quotas

Sociale : «éduquer» les utilisateurs trop gourmands

Utilisation de GnuPG - 1/4

Présentation

- GNU Privacy Guard
- Équivalent libre de GPG
- N'utilise aucun algo breveté, supporte ElGamal, DSA, RSA, AES, 3DES, Blowfish, Twofish, CAST5, MD5, SHA-1, RIPE-MD-160 et TIGER
- Supporte les dates d'expiration de clé et de signature
- Support intégré des serveurs de clés HKP

Des front-ends graphiques : `kgpg`

Exemple d'utilisation de GPG - 2/4

Pour l'utilisateur

Pourquoi ?

- Chiffrer certaines données
- De façon transparente pour ses mails
  projet Ägypten inclut les normes S/MIME et X.509v3
- Pour signer

Comment ?

- Un message est chiffré avec la clé publique du destinataire
- Un message est signé avec sa clé privée
- Gérer un trousseau de clés :
  - `gpg --gen-key` pour générer sa clé + phrase clé
  - `gpg --export [--armor]`
  - `gpg --import`
  - `gpg --list-keys`
  - `gpg -e dest [message]` et `gpg -d [message]`

Lire :

`http://webber.dewinter.com/gnupg_howto/english/GPGMiniHowto.html`

Exemple d'utilisation de GPG - 3/4

Pour l'administrateur

- Pour le noyau[10]
  - → Récupérer la clé publique :
    `gpg --keyserver wwwkeys.pgp.net --recv-keys 0x517D0F0E`
  - → Vérification qu'il s'agit de la bonne clé importée :

gpg --fingerprint

```
/root/.gnupg/pubring.gpg
------------
pub 1024D/517D0F0E 2000-10-10
    Empreinte de la clé = C75D C40A 11D7 AF88 9981 ED5B C86B A06A 517D
0F0E
uid                  Linux Kernel Archives Verification Key
<ftpadmin@kernel.org>
sub 4096g/E50A8F2A 2000-10-10
```

  - → Récupérer le noyau **ET** sa signature
  - → Vérification des sources du noyau :
    `gpg --verify linux-2.6.x.tar.bz2.sign linux-2.6.x.tar.bz2`

[10] http://www.kernel.org/signature.html

Exemple d'utilisation de GPG - 4/4

Pour l'administrateur

- Pour `apt-key`
  - Les packages Debian sont signés !

→ Récupérer la clé pulique :

```
gpg --keyserver wwwkeys.pgp.net --recv-keys 0x4F368D5D
```

→ Vérification qu'il s'agit de la bonne clé importée :

gpg --fingerprint

```
/root/.gnupg/pubring.gpg
------------
pub 1024D/4F368D5D 2005-01-31 [expire : 2006-01-31]
    Empreinte de la clé = 4C7A 8E5E 9454 FE3F AE1E 78AD F1D5 3D8C 4F36
8D5D
uid Debian Archive Automatic Signing Key (2005) <ftpmaster@debian.org>
```

→ Ajouter la clé à notre trousseau apt :

```
gpg --export --armor 4F368D5D | apt-key add -
```

→ Pour lister les clés du trousseau :

```
apt-key list
```

→ Pour effacer une clé du trousseau :

```
apt-key del KeyID
```

D'autres petites choses...

Charte informatique

ayant pour objet de définir les règles d'utilisation des moyens informatiques et de rappeler les responsabilités des utilisateurs

- Pas de prêt de mots de passe
- Pas de prêt de session
- Pas de mot de passe près de l'ordinateur
- Un écran verrouillé

- Attention aux `alias`, utiliser `whereis`

- Pas d'utilisateurs ? :)

Cinquième partie

Sécuriser un système

Prendre conscience qu'il y a un risque

Le réseau peut être

- Écouté
- Analysé

Les machines sont scannées

- Savoir quels services sont ouverts
- Essayer d'exploiter les failles de sécurité

Lire :

`http://www.linux-france.org/prj/inetdoc/cours/intro.analyse/`

Quelques outils - 1/4

`Ethereal`[11]

Fonctionnalités

- → Donne graphiquement les mêmes résultats que `tcpdump`
- → Nécessite les droits `root` car mode promiscuous
- → Analyse les trames captées sur les interfaces (dt WIFI) : *sniff*
- → Reconnaît et présente de nombreux protocoles

Parades

- → Protocoles chiffrés
- → Utilisation de switchs (hubs commutés)
- → Réduction de la portée pour le WIFI

Ethereal est disponible sous Linux, Mac OS X,Windows, Solaris, et gratuit

[11] `http://www.ethereal.com/`

Quelques outils - 2/4

Nessus[12]

Fonctionnalités

- → Permet de scanner une ou plusieurs machines
- → Permet aussi de tester différentes attaques pour savoir si une ou plusieurs machines sont vulnérables.
- → Très utile lors de tests de pénétration (pen test) et fait gagner un temps incroyable.

Fonctionnement

- Composé d'une partie serveur contenant une base de données de différents types de vulnérabilités
- Composé d'une partie client via lequel l'utilisateur s'authentifie sur le serveur
- L'utilisateur demande au serveur de tester une ou plusieurs machines
- Récupération des résultats des tests

Nessus est disponible sous Linux et Windows, et gratuit

[12] `http://www.nessus.org`

Quelques outils - 3/4

Nmap[13]

Fonctionnalités

→ Repérer les serveurs offrant des services particuliers et de les identifier : *scan*

→ Permet de déterminé le système d'exploitation

Nmap est disponible sous Linux et Windows, et gratuit

[13] http://www.insecure.org/nmap/

Quelques outils - 4/4

Nagios[14], ou la surveillance de services...

Non vu cette année

[14] http ://www.nagios.org/

Une machine sur la défensive : le Firewall - 1/2

Nécessite les bonnes option dans le noyau

Pourquoi et comment ?

- Protéger un réseau contre accès maltintentionné
- Une commande : `iptables`
- De très nombreuses règles

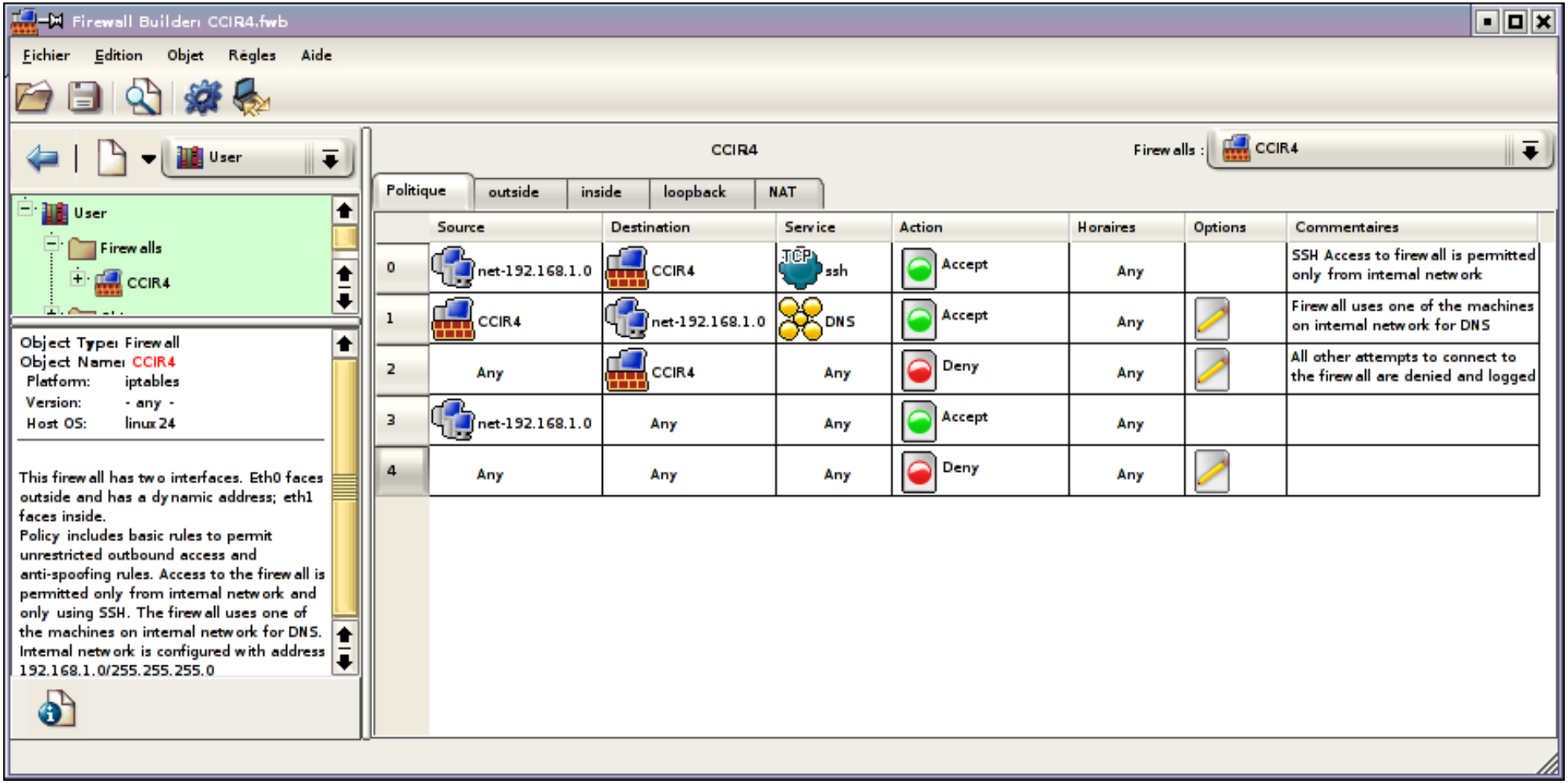
Front-End : Firewall Builder[15]

- Tout se fait à la cliquouille
- ... mais on doit avoir de bonnes notions

[15] `http://www.fwbuilder.org`

Une machine sur la défensive : le Firewall - 2/2

fwbuilder : snapshot et par l'exemple...



Exemple de configuration d'un Firewall

```
#!/bin/sh
# Insertion des modules de traces de connexion (pas nécessaire si intégrés au noyau).
modprobe ip_tables
modprobe iptable_filter
modprobe ip_conntrack
modprobe ip_conntrack_ftp
modprobe ipt_state
modprobe ipt_LOG

# permet les connexions local uniquement
iptables -A INPUT -i lo -j ACCEPT
# free output on any interface to any ip for any service (equal to -P ACCEPT)
iptables -A OUTPUT -j ACCEPT
# autorise les réponses à des connexions déjà établies
# et permet les nouvelles connexions en relation avec celles déjà établies (par
# exemple active-ftp)
iptables -A INPUT -m state -state ESTABLISHED,RELATED -j ACCEPT
# Enregistre tout le reste : Quelle est la dernière vulnérabilité de Windows ?
iptables -A INPUT -j LOG -log-prefix "FIREWALL :INPUT "
# Met en place une politique saine : tout ce qui n'est pas accepté > /dev/null
iptables -P INPUT DROP
iptables -P FORWARD DROP
iptables -P OUTPUT DROP
# soit verbeux pour les adresses dynamiques (pas nécessaire dans le cas des addr IP
statiques)
echo 2 > /proc/sys/net/ipv4/ip_dynaddr
# désactive ExplicitCongestionNotification - trop de routeurs les ignorent encore
echo 0 > /proc/sys/net/ipv4/tcp_ecn
```

Des machines sur la défensive

FIXE : Packages Portsentry, etc.

Cinquième partie

Sécuriser un système

Communiquer via SSH

Au tableau, en TP...

Utiliser Kerberos

- Serveur d'authentification externe
  - Centre de distribution de clés (KDC pour key distribution center)
  - Serveur de tickets (TGS pour ticket granting server)
- Permet de s'assurer de l'identité de son interlocuteur
- Fournit également des moyens de protéger la confidentialité et l'intégrité des données
- Un protocole : système de tickets au lieu de mots de passe en texte clair
- Repose sur des clés symmétriques

Déroulement

- Client fait requête sur serveur de clés KDC : les 2 connaissent K_C + mdp
- Serveur répond ticket T_{TGS} permettant requêtes au serveur de ticket TGS
- Ticket chiffré avec T_{TGS} FIXME :

Cinquième partie

Sécuriser un système

PKI

PKI (Public Key Infrastructure)

- Système de gestion des clefs publiques
- Permet de gérer des listes importantes de clefs publiques et d'en assurer la fiabilité
- Offre
  - Confidentialité : données illisibles par chiffrement
  - Authentification : identification de l'origine de l'information
  - Intégrité : les données n'ont pas subi de modification
  - Non-répudiation : émetteur ne peut nier l'envoi du msg
  - .. dans l'entreprise et lors d'échanges d'information avec l'extérieur

Quatre services principaux

- Fabrication de bi-clés
- Certification de clé publique et publication de certificats
- Révocation de certificats
- Gestion la fonction de certification

Montrer l'organisation, msg d'erreur

PKI gratuite : openSSL[16]

[16] http ://www.openssl.org

Des Sockets Sécurisées

- Procédé de sécurisation des transactions effectuées via Internet
- Communication sécurisée (chiffrée) entre deux machines (un client et un serveur) après une étape d'authentification
- Indépendant du protocole utilisé
- Transparent pour le client

Cinquième partie

Sécuriser un système

- Introduction
- Sauvegarder le système
  - Sauvegarder les données
  - Sauvegarder une installation
- Sécuriser son système
  - Un système sûr
  - De l'éducation des utilisateurs
- Les communications ou la sécurité orientée systèmes distribués et réseau
  - Être sur ses gardes
  - D'un point de vue utilisateur
  - D'un point de vue administrateur
- Conclusions

Conclusion

Je ne suis pas entré dans les détails à propos...

- d'IPSec[17]
- des systèmes de détection d'intrusion
- des rootkits
- des failles et exploits
- d'IP Spoofing
- de PAM
-
- et hélas de plein d'autres choses...

Mais vous pouvez trouver plein d'informations dans l'excellent tutoriel présent à l'URL `http://www.linux-france.org/prj/inetdoc/securite/tutoriel/`

[17] `http://www.securiteinfo.com/crypto/IPSec.shtml`