

AGR Administration et Gestion des Réseaux

Pierre BETTENS
pbettens (à) heb.be

ESI - École Supérieure d'Informatique

16/02/2009

- Cours - laboratoires (37.5h)
 - Exposé oral
 - Manipulation
- Évaluation
 - Examen oral au terme du cours
 - Evaluation du *savoir* par le biais d'une question théorique ouverte
 - Evaluation du *savoir-faire* par le biais d'une manipulation

Organisation - Supports

- Slides
- Supports divers
 - Internet
 - <http://esi.namok.be>
 - <http://del.icio.us/pit/gar> Bookmarks
 - IRC ([#esi](http://irc.freenode.net))
 - Les news (via Google - <http://groups.google.com>)

Organisation - Supports

- Références
 - kirch** Administration réseaux sous Linux
Olaf KIRSH et Terry DAWSON
ed. O'REILLY
 - hunt** TCP/IP Administration de réseau
Craig Hunt
ed. O'REILLY
 - ldap** LDAP, installation et mise en oeuvre
Gerard CARTER
ed. O'REILLY
 - samba** Samba, installation et mise en oeuvre
Robert Eckstein, Davis Collier-Brawn, Peter Kelly
ed. O'REILLY
 - welsh** Le système Linux
Welsh, Dalheimer et Kaufmann
ed. O'REILLY

- Organisation
- Organisation - Supports
- Organisation du laboratoire
- Organisation du travail
- Introduction à Linux (*deuxième partie*)
- Rappels réseaux

- DNS - Domain Name Server
(avec manipulations)
- NIS - Network Information System
- NFS - Network File System
- SAMBA
(avec manipulations)
- PAM - Plugeable Authentication Modules
(avec manipulations)
- LDAP - Lightweight Directory Access Protocol
(avec manipulations)

- ACL - Access Control List
- Serveur d'impression
 - CUPS - Common Unix Printing System
- Serveur web
 - Apache
(avec manipulations)

• Introduction à Linux (II)

Rappel du cours SYS1
 Gestion des utilisateurs
 Scripts de démarrage
 Shutdown
 Sauvegarde de fichiers
 Exécution de tâches périodiques `cron`
 Manipulations

- Rappel du cours SYS1

- Système de fichiers
 - Arborescence
- Droits d'accès
- Notions de processus
- Shell, `bash`

- Gestion des utilisateurs

- User - Group - Other (`u-g-o`)
- User, le fichier `/etc/passwd`
- Group, le fichier `/etc/groups`
- Gérer un utilisateur (ajout/suppression)
- Gérer les groupes
- Diverses commandes

- Gestion des utilisateurs

- Ajout d'un utilisateur
`adduser ...`
 - Modification de `/etc/passwd`
 - Modification de `/etc/group`
 - Copie des fichiers "squelettes"
 - Positionnement du mask (`umask`)
 - Création (éventuelle) d'un répertoire `home`

- Gestion des utilisateurs

- Le fichier `/etc/shadow`
 - Notions plus pointues (durée de validité, ...)
 - Commande `chage`
- Désactivation (temporaire) d'un compte
 - `Chgt du passwd`
 - `Chgt du shell (/bin/false)`
- Suppression d'un utilisateur
`userdel`

- Scripts de démarrage
 - Démarrage du système
(première partie en très bref)
 - BIOS
 - Chargeur de démarrage
 - lilo
 - grub
 - Noyau
 - `initrd.img` éventuel
 - `vmlinuz` ou autre

- Scripts de démarrage
 - Système SysV
 - Exécution de `/etc/inittab`
 - Exécution des scripts `/etc/rc?.d`
Ces scripts sont en fait des liens vers un script
 - S pour start
 - K pour kill
 - Chaque répertoire comprend un ensemble de liens

- Shutdown
 - L'arrêt du système est une prérogative du root ... sauf mention du contraire !
 - Arrêt du système
 - Prévenir les utilisateurs !
 - Arrêt de chacun des scripts (`rc*.d`)
 - Arrêt du processus `init` (`id=1`)

- Shutdown
 - Diverses manières de faire
 - Commande `shutdown`
 - Commande `halt`
 - Commande `reboot`
 - Commande `init 'x'`
 - `Ctrl-Alt-Del` ... *pq et qd ça marche ?*
 - La série à éviter ...
 - bouton ON/OFF
 - Tirer la fiche ou couper le câble
 - Bouton reset

- Sauvegarde des fichiers

- *L'importance d'un backup n'apparait jamais aussi cruciale que le jour de la perte des données.*
- Définition d'une politique de sauvegarde
- Que sauvegarder ?
- Sur quel(s) support(s) ?
- Moyens
- Remarques

- Sauvegarde des fichiers

- Définition d'une politique de sauvegarde
 - Que sauvegarder ?
 - A quelle fréquence ?
 - Sur quel support ?
 - Quel peut-être la période d'indisponibilité ?
 - Quel coût engage-t-on ?
 - De quel type d'erreur se protège-t-on ?
 - Cause naturelle
 - Défaillance matérielle
 - Défaillance humaine

- Sauvegarde des fichiers - Que sauvegarder ?

- Fichiers personnels
 - *Par exemple*
 - Sauvegarde quotidienne. (sur disque dur - rapide)
 - Sauvegarde hebdomadaire (sur bande dans le local/batiment - accessible)
 - Sauvegarde mensuelle (sur bande dans un autre local/batiment - gros désastre)
- Fichiers systèmes
- Autres ..

- Sauvegarde des fichiers - Sur quel support ?

- Disquette
- ZIP
- CD
 - Capacité 700Mb
- DVD
 - Capacité 8-9 Gb
- Lecteurs USB
- Bandes
 - Capacité jusqu'à 40 Gb

- Sauvegarde des fichiers - Moyens

- Commande `rsync`
- Commandes `dump`, `restore`
 - Sauvegarde incrémentale
 - Sur bandes
- Commande `tar`

- Sauvegarde des fichiers - Remarques

- Les fichiers sur support externes sont vulnérables
- Protéger les supports externes
 - Vol
 - Destruction
 - ...
- Préparer les scénarios de restauration
(la restauration se fait toujours dans l'urgence)

- Exécution de tâches périodiques

- Daemon associé `cron`
- Format d'un fichier cron

```
#commentaire
#minute heure jour mois jour_semaine commande
0,15,30,45 12-13 * * 1-5 /home/login/allermanger
```

- Fichiers de configuration

- `/etc/crontab` (lance les fichiers cron)
- `/etc/cron.allow`
- `/etc/cron.deny`

- Exécution de tâches périodiques

- Commande `crontab`
 - `-e` Édite le fichier `crontab` de l'utilisateur
 - Utilise l'éditeur `/usr/bin/editor`
 - Le fichier se trouve là (ss debian) et ne peut être édité.
`/var/spool/cron/crontabs/'login'`
 - `-l` liste, `-r` remove

• Rappels réseaux

En théorie

Routing

- En théorie
 - Protocole TCP
 - Adresse IP
 - Adresse ethernet
 - *hostname*, nom d'hôte
 - *netmask*, masque de réseau
 - 192.168.208.0/18
 - 192.168.208.0, 255.255.192.0

- En théorie (II)
 - *gateway*, passerelle
 - *broadcast*
 - Adresse dont tous les bits de la partie hôte sont à 1
 - Routing
 - routage statique
 - routage dynamique
 - Les commandes
 - *ifconfig*
 - *netstat*
 - *route*
 - *ping*
 - *dig*

- Configuration de l'interface.

Trouver l'interface

- Appelation usuelle
 - *ethi* sous Linux
 - *dnet*i** sous Solaris
- Recherche
 - `dmesg | grep eth` pour trouver les interfaces ethernet
- `netstat -in`
- Commande *ifconfig*
- Commande *netstat*
- Commande *dig*

- Configuration de l'interface (ifconfig, netstat)

- Infos
 - Flag
 - R - running
 - B -broadcast
 - U - up
 - L -loopback
 - MTU - Maximum Transfert Unit (taille des paquets)
 - RX -*info* (paquets reçus)
 - TX - *info* (paquets envoyés)
 - *info* : OK - reçu, ERR - erreur, DRP - drop, OVR - overruns
- netstat -in
- ifconfig eth0

- Configuration de l'interface (ifconfig, netstat)

- Configuration de l'interface
 - ifconfig eth0 192.168.208.i netmask 255.255.192.0
 - voir /etc/network/interfaces
- Activer / désactiver
 - ifconfig eth0 up
 - ifconfig eth0 down
- Mode *promiscuous* (indiscret)
 - Par défaut l'interface ethernet ne passe aux protocoles des couches supérieures que les trames adressées au système local ... sauf en mode indiscret.

- Routage

- Routage minimal
 - Réseau isolé
 - Pas de sous-réseau
 - Une route par interface
- Routage statique
 - Nombre limité de routeurs
 - Pas / peu d'évolution
 - Table de routage construite et maintenue manuellement via `route`
- Routage dynamique
 - Plusieurs route menant à la même destination
 - Les protocoles de routage mettent à jour les table en fct de l'évolution du réseau
 - Recherche d'une "meilleure" route

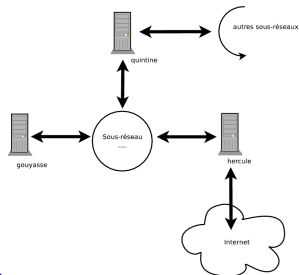
- Routage minimal

- Tests du réseau via `ping`
- Ajout d'une route :


```
route add default gw monGateway
```

- Routage statique

- Message ICMP Redirect
- Dans les scripts de démarrage
 - Exécution de `route`
 - Suppression des "protocoles de routage"



- Routage dynamique

- Protocoles de routage intérieurs
 - RIP - *Routing Information Protocol*
 - Compte le nombre de sauts (*hop*)
 - *daemon routed*
 - Hello
 - Tres peu utilisé
 - IS-IS - *Intermediate to Intermediate System*
 - Plus court chemin d'abord
 - OSPF - *Open Shortest Path First*
 - Adapté aux gros réseaux
- Protocoles de routage extérieurs
 - EGP
 - BGP
 - *via gated*

• DNS - Domain Name Server

Introduction

Résolveur

Serveur à cache seule

Serveur maître

Serveur esclave

- Lien entre les adresses IP et les noms

- Table d'hôtes
 - /etc/hosts*
- DNS

- Avantages du DNS par rapport à la table d'hôtes

- Le DNS permet de gérer un plus grand nombre d'hôtes
- Le DNS assure la dissémination de l'info

- Fonctionnement

- Si le DNS reçoit une requête sur un hôte pour lequel il ne possède aucune donnée
- Il fait suivre la requête à un *serveur ayant autorité*
- Lorsque le serveur lui répond, il maintient l'information dans un cache.
- La prochaine fois, il y répondra seul.

- Hiérarchie des domaines
 - ◆ Domaine racine
 - ◆ Domaine de premier niveau
 - Géographique
be, fr, us, ...
 - Administratif
com, edu, gov, mil, net, int, org, (depuis le début)
aero, biz, coop, museum, pro, info, name (depuis 2000)
- Serveurs **racines**
 - ◆ a.root-servers.net
 - ◆ ...
 - ◆ m.root-servers.net
- dig @a.root-servers.net.

- Implémenté grace à BIND
Berkeley Internet Name Domain
- Client : le *résolveur*
- Serveur : daemon *named*
- Quatre niveaux de services
 - ◆ Résolveur uniquement
 - ◆ Serveur à cache seul
 - ◆ Serveur maître
 - ◆ Serveur esclave

- /etc/resolv.conf
- nameserver 'adresse'
 - Adresse représente l'adresse d'un serveur de noms
 - Jusqu'à 3 serveurs de nom autorisés
 - Les serveurs de noms sont interrogés dans l'ordre
 - Si aucune entrée nameserver .. alors interrogation locale.
- domain 'nom'
 - Nom de domaine par défaut
 - Les noms SANS points sont concaténés au nom de domaine par défaut
 - Si la variable d'environnement LOCALDOMAIN est définie elle prend le dessus
- search 'domaine'
 - Idem que domain mais avec plusieurs domaines

- options 'option ...'
 - debug (si compilé avec l'option)
 - ndots :n
défaut 1, nombre de point (+1) rencontré dans le nom pour lequel le nom de domaine est concaténé
- timeout :n
 - Délai initial
 - Défaut 5
- attempts :n
 - Nombre de fois que le résolveur retente une requête
 - Défaut 2
- rotate
 - Répartit la charge entre les différents serveurs de noms

- Fichier de configuration - `named.conf`
- Fichier d'accès à la racine - `db.root` (par exemple)
- Fichier d'hôte local - `db.local`
- Fichier de zone - `db.<mazone.org>` (par exemple)
- Fichier de zone inverse - `db.<192.168.208>` (par exemple)

- Syntaxe proche de C
 - Commentaires `/* */` ou `//` ou encore `#`
 - Déclaration se termine par `;`
 - String entre `" "`
 - Groupe entre accolades `{ }`
- Commande de configuration
 - `acl` - Définit une liste de contrôle d'accès d'adresses IP
 - `include` - Inclut un autre fichier
 - `key` - Définit les clés de sécurité pour l'authentification
 - `logging` - Définit ce qui doit être loggé
 - `options` - Définit les options de configuration globale et des valeurs par défaut
 - `server` - Définit les caractéristiques d'un serveur distant
 - `zone` - Définit une zone

Une zone est une partie de l'espace de nom de domaine pour laquelle le serveur de noms a autorité

- Format de fichier de zone
 - `[nom] [ttl] IN type donnée`
- nom
 - Nom de l'objet du domaine
 - Le nom est relatif au domaine courant sauf si il se termine par un `.` S'il est blanc, il se rapporte au dernier objet du domaine nommé
- ttl
 - *Time-to-live*
 - Généralement vide, la valeur de la directive **\$TTL** est utilisée
- IN
 - enregistrement de ressource internet
- type
 - Identifie la nature de l'enregistrement
 - SOA, NS, A, PTR, MX, CNAME, TXT
- donnée
 - Information spécifique au type d'enregistrement.
 - Exemple : pour un champ de type A, la donnée est l'adresse IP

- Enregistrement SOA
 - Numéro de série en 10 chiffres - `aaaammjxx`
 - Temps de rafraichissement
 - Temps en secondes entre les vérifications du numéro de série par les secondaires
 - Temps de réémission
 - Temps en secondes entre les vérifications du numéro de série par les secondaires si la première vérification a échoué
 - Temps d'expiration
 - Si un secondaire n'arrive pas à contacter le serveur primaire de la zone, il continue à répondre aux requêtes pendant la durée donnée
 - TTL

- Directives

- \$TTL

- Valeur par défaut du TTL pour les enregistrements.
 - Soit un nombre de secondes (valeur chiffrée)
 - Soit une combinaison de chiffres et de lettres *w, d, h, m, s*

- \$ORIGIN

- Définit le nom de domaine par défaut
 - Ecrase la valeur du domaine définie par la déclaration de zone

- \$INCLUDE

- Inclu un fichier externe (à l'endroit de la directive)

- \$GENERATE

- Génère une série d'enregistrements
 - Ces enregistrements ne diffèrent que par une valeur numérique
- \$GENERATE 1-4 \$ CNAME \$.1to4
- Génère
- 1 CNAME 1.1to4 - 2 CNAME 2.1to4 - 3 CNAME 3.1to4 - 4 CNAME 4.1to4

- cat /etc/bind/named.conf

```
include "/etc/bind/named.conf.options";

// prime the server with knowledge of the root servers
zone "." {
    type hint;
    file "/etc/bind/db.root";
};

zone "localhost" {
    type master;
    file "/etc/bind/db.local";
};

zone "127.in-addr.arpa" {
    type master;
    file "/etc/bind/db.127";
};

(...)

include "/etc/bind/named.conf.local";
```

- cat /etc/bind/db.root

- Récupéré tel quel, il contient les adresses des serveurs racines

```
// extrait
.                3600000      IN      NS      A.root-servers.net
A.ROOT-SERVERS.NET. 3600000      A       198.41.0.4
A.ROOT-SERVERS.NET. 3600000      AAAA    2001:503:BA3E::2:30
(...)
```

- cat /etc/bind/db.local

- Permet de convertir l'adresse de rebouclage en localhost

- Excepté le nom de machine, fichier identique sur ttes les machines

```
;
; BIND data file for local loopback interface
;
$TTL 604800
@ IN SOA localhost. root.localhost. (
    2          ; Serial
    604800    ; Refresh
    86400     ; Retry
    2419200   ; Expire
    604800 )   ; Negative Cache TTL
;
@ IN NS localhost.
@ IN A 127.0.0.1
@ IN AAAA ::1
```

- `cat /etc/bind/db.127`
- Résolution inverse pour l'adresse de reboilage

```

;
; BIND reverse data file for local loopback interface
;
$TTL 604800
@ IN SOA localhost. root.localhost. (
    1          ; Serial
    604800    ; Refresh
    86400     ; Retry
    2419200   ; Expire
    604800 )   ; Negative Cache TTL
;
@ IN NS localhost.
1.0.0 IN PTR localhost.

```

- `cat /etc/bind/named.conf.local`
- Ajout au fichier `named.conf.local` (inclu dans `named.conf`) de la (des) zone(s) à traiter

```

...

zone "esi.be" {
    type master;
    file "/etc/bind/db.esi.be";
};

zone "208.168.192.in-addr.arpa" {
    type master;
    file "/etc/bind/db.192.168.208";
};

```

- `cat /etc/bind/db.esi.be`
- Principalement des enregistrements A et CNAME

```

$TTL 86400
@ IN SOA .....
; Serveurs de noms et de mail
    IN NS guyasse.esi.be.
    IN MX 10 monisp.be.
; Definition de localhost
localhost IN A 127.0.0.1
; Notes de la zone
guyasse IN A 192.168.208.1
nsl IN CNAME guyasse.esi.be
quintine IN A 192.168.208.2
.....

```

- `cat /etc/bind/db.192.168.208`
- Principalement des enregistrements PTR

```

$TTL 86400
@ IN SOA guyasse.esi.be ...
...
    IN NS guyasse.esi.be
1 IN PTR guyasse.esi.be
2 IN PTR quintine.esi.be

```

- La différence avec un serveur maître réside dans les fichiers de zone. Ceux-ci sont **écrits** (par le daemon) sur base d'une requête au serveur maître et ne contiennent pas **à priori** les informations sur la zone.

- cat /etc/bind/named.conf.local

```
...
zone "esi.be" {
    type slave;
    file "/etc/bind/db.esi.be";
    masters { 'adresse ip du maitre' ; };
};

zone "208.168.192.in-addr.arpa" {
    type slave;
    file "db.192.168.208";
    masters { 'adresse ip du maitre' ; };
};
```

- Contrôle du processus
 - Utilisation du script bind9
 - /etc/init.d/bind9 start|stop|reload
 - Commande **rndc** de gestion du processus
 - status
 - stop
 - start /restart
 - reload
 - stats
 - trace / notrace
 - querylog

- cat /etc/bind/rndc.conf

```
key rndc_key {
    algorithm "hmac-md5";
    secret "...";
};

options {
    default-server localhost;
    default-key rndc_key;
};
```

- Permet le contrôle du processus **named** à distance et sécurisé
- Default serveur* représente la machine à contrôler

- /etc/named.conf ajout

```
controls {
    inet 127.0.0.1 allow {localhost; } keys {rndc_key; };
};

key "rndc_key" {
    algorithm hmac-md5;
    secret " ... idem que l'autre ...";
};
```

- Named autorise certaines adresses IP à le contrôler

- Test de la configuration

```
dig <nom de domaine>
```

- nslookup est *deprecated*

• NFS - Network File System

Introduction
 Daemon
 Droits d'accès
 Commandes associées

- NFS - *Network File System*
- Permet le partage de fichiers en réseau
- Idéalement transparent pour l'utilisateur
- Avantages
 - Réduit l'espace disque total puisque partage
 - Simplifie la gestion centralisée
 - Utilise le *set* de commandes habituel
- Approche client / serveur
 - Le serveur
 - Système qui rend les répertoires disponibles
 - **export**
 - Le client
 - Système qui attache des répertoires distants à son *filesystem*
 - **mount**
- Initialement développé par *Sun Microsystems*

- `nfsd` [`nservers`]
 - prend en charge les requêtes des clients
 - partie serveur
 - `nservers` spécifie le nombre de daemon qui tournent
- `mountd`
 - traite les demandes de montage des clients
 - lancés par les serveurs
- `nfslogd`
 - responsable du journal de NFS
- `rquotad`
 - relatif aux quotas des utilisateurs
 - tourne sur les clients et les serveurs
- `lockd`
 - gère les verrous sur les fichiers
 - tourne sur les clients et les serveurs
- `statd`
 - tourne sur les clients et les serveurs
 - surveillance de l'état du réseau (pour la gestion des locks)

- Pourquoi ?
 - fournir de l'espace à des clients sans disque
 - éviter la duplication des données
 - offrir des données et programmes centralisés
 - partager des données
- Fichier `/etc/exports`
 - Exemple


```

          /usr/man gouyasse(rw) quintine(ro)
          /usr/local (ro)
          
```
 - Format : répertoire [`machine(options)`] ...
 - *Wildcard* et/ou adresses IP autorisés
- Particularités de Solaris
 - commande `share`
 - fichier `dfstab`

- Fichier autorise l'accès de machines
- Les droits d'accès Unix sont de rigueur
- Droits d'accès basés sur les **uid** et **gid** ... c'est donc mieux (ou pas) s'ils correspondent d'une machine à l'autre.
- L'utilisateur `root`
 - directive `root_squash`
 - `uid root -> uid nobody`
 - directives `squash_uids`, `squash_gids` et `all_squash`
 - directive `map_daemon`, permet de faire correspondre un UID à un autre (voir `rpc.ugidd`)

- Commande `exportfs`
 - `-a` lors de l'init
 - `-r` pour une lecture
- Construit le fichier `/var/lib/nfs/xtab`
 - Contient les infos sur les fichiers exportés
 - Lu pas `mountd`
- Possibilité d'export "temporaire"
 - `exportfs hercule :/usr/local -o rw - pour l'ajout`
 - `exportfs -u hercule :/usr/local - pour la suppression`

- Commande `showmount`

- Permet de voir les répertoires exportés pas une machine

- `showmount -e gouyasse`

```
export list for gouyasse
/usr/man gouyasse,quintine
/local (everyone)
```

- Commande `mount`

- `mount machine :répertoire-distant répertoire-local`
 - *machine* est un serveur NFS
 - *répertoire-distant* un répertoire exporté
 - *répertoire-local* doit exister
 - Ajout éventuel du type de *filesystem*
 - `-t nfs`

- Commande `umount`

- Fichier `/etc/fstab`

- Les répertoires exportés peuvent apparaître dans le fichier

```
gouyasse:/usr/man /usr/man nfs rw 0 0
```

- Propose des options supplémentaires aux options habituelles du fichier

NIS - Network Information Service

yellow pages

NIS - Network Information Service

yellow pages

• NIS - Network Information Service

yellow pages

Introduction

Daemon

Commandes associées

- Base de données administrative (comparable au DNS mais **différent**)

- Similitudes

- Contrôle centralisé
 - Propagation automatique des fichiers de configuration importants

- Différences

- Gère des petits réseaux privés (pas Internet)
 - NIS partage des infos plus variées (dans ses *tables NIS*)
 - La table d'hôte de NIS contient moins d'informations que celle de DNS

- **Définition**

NIS convertit plusieurs fichiers standards en bases de données qui peuvent être interrogées via le réseau, ces bds sont appelées *tables NIS*

- Quels fichiers ?
 - /etc/passwd
 - /etc/group
 - /etc/ethers (utilisé par le protocole RARP)
 - /etc/hosts
 - /etc/networks
 - /etc/protocols
 - /etc/services
 - /etc/aliases
- Ces fichiers sont transformés en table
/etc/networks -> networks.byname networks.byaddr
- Les tables NIS sont stockées dans /var/yp/<nom du domaine>

- *ypserv*
Daemon responsable de la partie serveur de NIS
- *ypbind*
Daemon permettant la liaison au serveur et à ses tables NIS

NIS - les commandes (serveur)

- *ypcat* - fournit la liste des tables NIS
- *domainname* <nom du domaine> - vérifie et met en place le nom du domaine NIS
(défini pour le boot dans /etc/sysconfig/network, NISDOMAIN=...)
- *cd /var/yp; make* - construction des tables NIS
- *ypserv* - lancement du serveur NIS
- *ypbind* - processus de liaison
- *ypwich* - renseigne le serveur NIS

NIS - les commandes (client)

- *domainname* <nom du domaine> - *idem serveur*
- *ypbind*

• Samba

Introduction
 Daemon
 Configuration
 Utilitaires

- Permet la communication entre machines hétérogènes
- Mets en oeuvre le protocole **SMB** (natif sous MS Windows)
- Administration centralisée sur le serveur
- Site associé : <http://samba.org>
- Installation
 - samba
 - samba-common
 - smbfs
 - smbclient

Samba - *daemon*

- `smbd`
 - *daemon* responsable du partage des ressources
 - File sharing
 - Printing services
- `nmbd`
 - *daemon* NetBios
 - Comprend et répond aux requêtes NetBios sur TCP/IP produites par SMB
 - Permet la participation au "Network Neighborhood"
- Activation
 - lancement des *daemons*, `nmbd`, `smbd`
 - utilisation du script `/etc/init.d/samba` (start/stop/..)
 - via `inetd`

Samba - `smb.conf`

- Configuration centralisée dans le fichier `/etc/samba/smb.conf` (vérifier la situation)
- Fichier divisé en **sections**
 - Débute par [**nom du partage**]
 - Une section se termine par le début de la suivante (ou fin de fichier)
 - Chaque section correspond à un partage, (excepté pour la section *global*)
 - Sections particulières
 - **global** - Configuration générale de Samba
 - **homes** - Correspond au répertoire HOME de l'utilisateur
 - **printers** - Définit le partage des imprimantes.

- Format de fichier

```
parametre = valeur
```

- Les commentaires commencent pas # ou ;

- Exemple

```
# A sample share for sharing your CD-ROM with others.
[cdrom]
comment = Samba server's CD-ROM
; valid users = user1, user2
writable = no
locking = no
path = /cdrom
public = yes
```

- Samba comprend une série de variables ...
- %l - adresse IP du client
- %m - nom netbios du client
- Ces variables permettent l'écriture de scripts personnalisés

- On ajoutera, par exemple,

```
[monJoliPartage]
...
include /etc/samba/smb.conf.%m
...
```

- Si le fichier existe il est inclu ... sinon non.

Samba - configuration du *browser*

- *browser* ou explorateur réseau permet d'examiner les partages
- l'option *browseable* *yes|no* permet de montrer / cacher un partage sur le réseau (\$ sous MS Windows)
- une **liste de *browsing*** est maintenue sur le réseau
- le **master browser** la détient
- ce **master browser** est élu
 - les élections dépendent de
 - os level
 - role
 - local master
 - netbios name
 - preferred master *yes|no*

Samba - authentification

- Types d'authentification
 - **share** - authentification 'à la ressource'
 - **user** - authentification lors de la connexion
 - **server** - comme pour user mais le serveur s'adresse à un autre serveur pour l'authentification
 - **domain** - contrôle via un 'contrôleur de domaine' (responsable de l'authentification)

- `testparm /etc/samba/smb.conf`
 - Permet de vérifier la validité syntaxique du fichier de conf
- `/etc/init.d/samba [start|stop|restart]`
 - Relance le *daemon* samba
 - Le script s'appelle *smb* ou *samba* suivant les distributions
- `smbmount`
 - package debian *smbfs*
 - syntaxe `smbmount //<netbios name>/<share name> <mount point>`
- `smbclient`
 - commande à tout faire
 - FTP
 - impression
 - envoi de messages
 - ...
- `smbpasswd`
 - ajoute un utilisateur "samba"

- Disponibilité du service (`/etc/services`)
- Permet la configuration de samba via un interface web (plutôt que l'édition du fichier `smb.conf`)
- Gestion par `initd`
- Le serveur écoute sur le port 901
`http://localhost:901`

• PAM - Pluggable Authentication Module

- Principe
 - Certaines applications nécessitent une authentification
 - `login`
 - `sudo`
 - `su`
 - ...
 - Systèmes d'authentification évoluent
 - `/etc/passwd`
 - `/etc/shadow`
 - Annuaire LDAP
 - ...
 - Cette évolution impose la réécriture d'une partie de code de **chaque** application nécessitant une authentification
 - **L'idée** ; on délègue l'authentification à des modules dynamiques
- Définition
 - *Pluggable Authentication Module* sont des bibliothèques responsables d'une partie de l'authentification.

- Bibliothèque
 - `/lib/security`
 - Une application est développée pour se lier avec ces bibliothèques
- Avantage
 - L'administrateur system configure le comportement de ces applications (ssh, ftp, login, ...) via PAM
 - La configuration se fait dans `/etc/pam.d/` (un fichier par application)
 - Anciennement la configuration se faisait dans un fichier `/etc/pam.conf` unique
 - Configuration **fine**
 - Refus simple de connexion
 - Connexion "limitée"; plage horaire, ressources, ...
- Condition
 - Il faut que l'application soit *PAM enabled*

- Format des fichiers
 - `module-type control-flag module-path args`
- `module-type`
 - **authenticate**
 - Identifie le *user* comme étant qui il prétend
 - Vérifie l'appartenance à un groupe
 - **account**
 - Pas d'authentification mais des permissions/restrictions en fonction des ressources
 - temps (moment de la journée)
 - ressources système (nombre d'utilisateurs connectés)
 - lieu (root se logge d'une console pas d'un terminal)
 - **session**
 - Destiné aux actions à exécuter avant/après la mise à disposition du service
 - **password**
 - Utilisé pour renouveler le jeton d'authentification

- Format des fichiers
 - `module-type control-flag module-path args`
- `control-flag`

Gère la manière de réagir au "résultat" du module.

 - **Rem** : Les modules sont **empilés**, et exécutés dans l'ordre .. le résultat de l'un influence le suivant
 - *required*
 - Exigé pour la réussite du *module-type*
 - Un echec n'est renseigné qu'à la fin de la pile d'appel
 - *requisite*
 - Idem que *required*
 - Mais s'interrompt dès l'échec ... n'attend pas l'exécution de toute la pile
 - *sufficient*
 - La réussite de ce module est suffisante .. on ne continue pas la pile d'appel en cas de réussite
 - *optional*
 - Optionnel .. n'influence pas la suite

- Format des fichiers
 - `module-type control-flag module-path args`
- `module-path`
 - Nom du module
 - S'il commence par / c'est un nom complet sinon `/lib/security`
- `args`
 - Arguments pour le module, dépend de celui-ci
 - `debug, no-warn, use-first-pass, ...`

- Exemple

```
auth required /lib/security/pam_securetty.so
auth required /lib/security/pam_env.so
auth sufficient /lib/security/pam_ldap.so
auth required /lib/security/pam_unix.so try_first_pass
```

- Déroulement

- Vérification dans `/etc/securetty` que la connexion peut se faire sinon échec ... à la fin
- Positionnement des variables d'environnement
- Authentification via LDAP (`/etc/ldap.conf`)
 - Si réussite, fin
- En cas d'échec de `pam_ldap`, authentification Unix .. avec le `passwd` précédent

- Tester si l'application `pamsgm` supporte PAM
- Ajouter le fichier `pamsgm` dans `/etc/pam.d`

```
$ cat /etc/pam.d/pamsgm
auth required pam_permit.so
auth required pam_warn.so
```

- Lancer le programme `pamsgm`
 - Module `pam_permit` autorise tout le monde
 - Module `pam_warn` logge dans `syslog`

- Ecrire un programme `PAM enabled`

```
#include <security/pam_appl.h>
#include <security/pam_misc.h>
...
pam_authenticate();
...
```

```
cc -o application .... -lpam -lpam_misc -ldl
```

LDAP - Lightweight Directory Access Protocol

- Définition

- LDAP est un protocole d'accès à un annuaire.
- Un **annuaire** est une base de données spécialisée,
 - stocke des données légèrement typées
 - les données sont structurées en arbre
 - un annuaire est très performant en **lecture** mais pas en écriture

- Exemples

- annuaire de personnes, type "pages blanches"
- comptes Unix
- carnet d'adresses + photos
- données d'identification
- parc matériel
- ... *tout ce qui peut-être nommé et attaché à de l'information*

- Annuaire

- Lectures rapides
- Stocke des objets et leurs attributs (typés)
- Organisation en arbre
- Réplication simple (chaque modification est reportée dans les annuaires secondaires, ...)
- Stocke gde quantité de données mais de faible volume

- SGBD

- Rapidité d'accès en lecture et écriture
- Typage fort

- LDAP fournit

- Un *protocole* permettant l'accès à l'information
- Un *modèle d'information*, définit le type d'informations
- Des conventions de *nommage*, définissent comment l'information est organisée
- Un modèle *fonctionnel*, définit comment on accède à l'information
- Un modèle de *sécurité*
- Un modèle de *duplication*, définit la répartition entre différents serveurs
- Des APIs pour développer des applications
- LDIF, un format d'échange de données

- Le protocole définit

- comment s'établit la communication client-serveur
- permet à l'utilisateur de se **connecter**, **rechercher**, **comparer**, ...
- des mécanismes de chiffrement
- des règles d'accès
- un protocole serveur-serveur, pour la synchronisation, réplication, ..
- *Pour info ...*
 - LDAP est initialement une passerelle d'accès à des annuaires **X500**

- **Modèle de données hiérarchique**
- Chaque noeud de l'arbre correspond à une entrée de l'annuaire
- Les entrées correspondent à des *objets*, ayant des *attributs*
- Chaque serveur contient une entrée spéciale, **rootDSE** (*root directory specific entry*) qui contient la description de l'arbre
- objetClass *top* : permettra de définir la "véritable" racine de l'arbre
- L'arbre est appelé *Directory Information Tree*, **DIT**
- L'ensemble des définitions relatives aux données, s'appelle un **schéma**

- Les classes d'objet (*objectClass*) modélisent les objets et leurs attributs
- Une classe est définie par
 - un nom
 - un OID (*object ID*)
 - des attributs obligatoires
 - des attributs optionnels
 - un type (structurel, abstrait ou auxiliaire)
 - **structurel** - description d'un objet basique, personne, groupe, entité organisationnelle de la société, ...
 - **abstrait** - propre à LDAP, top, alias
 - **auxiliaire** - permettent d'ajouter de l'info complémentaire à un objet structurel, *mailRecipient*, ...
- Un attribut est défini par
 - un nom
 - un oid
 - syntaxe et règles de comparaison
 - format de valeur

- **OID**
 - Les objets et leur oid sont normalisés (RFC2256)
 - *oid* est une séquence de nombres entiers
- ```
2.5 - fait référence au service X500
1.3.6.1.4.1.4203 - openLDAP
```
- On ne modifie pas les schémas existants (pas propre, risque d'incompatibilité)
  - Notion d'héritage entre objets
  - Pour l'ESI, (1.3.6.1.4.1.23162)
- Les classes d'objet forment une hiérarchie
  - La racine est l'objet *top*
  - Chaque objet hérite de son parent
  - On précise la classe d'un objet à l'aide de *objectClass*

- ```
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetPerson
```
- **classe person**
 - a comme attributs : *commonName, surname, description, seeAlso, telephoneNumber, userPassword*
- **classe organizationalPerson**
 - ajoute les attributs : *organizationUnitName, title, ...*
- **classe inetPerson**
 - ajoute les attributs : *mail, uid, photo, ...*

- Lorsqu'une entrée est créée, le serveur vérifie si la syntaxe est conforme sur base du schéma associé, c'est le *Schema checking*
- /etc/ldap/schema/local.schema

```
objectClass esiPerson
superior inetOrgPerson
requires
  sn,
  cn
allows
  uidNumber,
  gidNumber,
  homeDirectory,
  dateArrivee,
  dateDepart
```

- Formats différents suivants l'implémentation
 - slapd.conf, U-M slapd, OpenLDAP, Netscape Directory
 - ...
- Il existe deux objets abstraits particuliers qui permettent de faire des liens entre les noeuds ou entre des annuaires
 - *aliases*
 - *referrals*
- Un annuaire LDAP peut être constitué d'un seul serveur ou de plusieurs
 - Serveur seul
 - Service *referral*
 - Service *duplication*

- L'identifiant unique (clé dans un SGBD) est le **DN**
- **DN**, *Distinguished name* est le nom unique dans l'annuaire, il représente le chemin absolu depuis *top*
- Exemple :uid=pbt, ou=prof, dc=esi, dc=be
- Il se compose
 - des attributs obligatoires
 - de la liste des **ou** *organisationnal unit*
 - des organisations **o**

- LDIF permet de représenter les données
- Utilisé pour
 - importer / exporter des bds
 - faire des modifications sur des entrées

```
dn : cn=PbT, ou=prof, o=esi, c=Be
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
cn: PbT
mail : pbettens@heb.be
...
```

- Permet de faire des modifications, **ajouts, suppression**, ...
- Exemple d'ajout

```
dn: cn=Juste Leblanc, ou=sales, o=Ed Oreilly, c=fr
changetype: modify
add: telephonenumber
telephonenumber: (408) 123 - 456
```

- Exemple de suppression

```
dn: cn=Juste Leblanc, ou=sales, o=Ed Oreilly, c=fr
changetype: delete
```

- Le modèle fonctionnel définit les opérations de bases pouvant être exécutées sur le serveur
 - search, compare, add, modify, delete, rename, bind ...
 - (voir webographie pour les détails)
- Pour une recherche, on devra définir le *scope* de celle-ci
 - search scop = base
permet de rechercher un élément
 - search scope = onlevel search
permet de rechercher sur le niveau enfant
 - search scope = subtree
permet la recherche dans tout l'arbre "enfant"

- *Déployer un annuaire nécessite une réflexion au sein de la société. Ces aspects sortent du cadre de cette présentation ... mais en bref*
- Aspects organisationnels
 - Nature des données stockées
 - Que doit servir l'annuaire ?
 - Maintient des données à jour, sources des données, pérennité
 - Confidentialité, authentification, contrôle d'accès, ...
- Choix du schéma
- Choix du modèle de nommage
 - Nombre d'entrées actuelles et évolution
 - Type des entrées
 - Nombre de serveurs et répartition des données sur ceux-ci
 - choix du DN *Distinguished name*
 - choix du suffixe, exemple DC=esi, DC=be
- Duplication ? Réplication ?
- ...

- **OpenLDAP** est une implémentation libre de LDAP
- `http://openldap.org`
- *Related software*
 - Transport, OpenSSL (`http://openssl.org`)
 - Authentification, Kerberos
 - Threads, OpenLDAP supporte *POSIX pthreads*

- Installation
 - Via les *packages*
 - slapd
 - ldap-utils
 - Méthode "traditionnelle"
 - configure; make; make install ...
- *daemons*
 - slapd, pour la gestion de l'annuaire
 - slurpd, pour la réplication

- Configuration
 - *via* le fichier `/etc/ldap/slapd.conf`
 - Adaptation du fichier de configuration
 - Adresse IP du serveur LDAP
 - Position du DN de l'annuaire
 - SSL *yes/no*
 - Schema(s) supplémentaire(s) éventuel(s)
- Script
 - Gestion du serveur *via*
 - `/etc/init.d/slapd start|stoprestart|force-reload`

- Organisation de l'annuaire
 - Vérification des **schemas**
 - (probablement), création d'un schema particulier
`/etc/ldap/schema/local.schema`
- Gestion du contenu
(*fichier(s) LDIF*)
 - Ajout d'utilisateur, `ldapadd`

- Serveurs
 - OpenLDAP
 - Netscape Directory Server
 - Innosoft's Distributed Directory Server
 - ...
 - D'autres supportent les requêtes
 - Novell'NetWare Directory Services
 - Microsoft Active Directory
 - Lotus Domino
- Type de clients
 - Logiciels avec accès natif, Netscape Communicator, MS Outlook, *Browsers*
 - Accès *via* passerelle
 - Utilisation des API, Java ; Perl, C,
 - Natif ds l'OS, MS Windows NT5, PAM LDAP, NIS *versus* LDAP

• Serveur web Apache

- Un *serveur web* permet la propagation de l'information sur un réseau IP
- **Apache** est un logiciel fournissant le "service" serveur web
- Installation
 - `apt-get install apache2 apache2-doc libapache2-mod-php`
- Fichier de configuration
 - `httpd.conf` (obsolète dans la version apache2)
 - `apache2.conf`
 - `conf.d/`
 - `mod-enabled/` (*versus* `mod-available/`)
 - `ports.conf`
 - ... bref `ls /etc/apache2`

- Script de gestion du(des) *daemon(s)*
 - `/etc/init.d/apache2 [start|stop...]`
 - Nombre de *daemons* (**essaim**)
 - `StartServers 5`
 - `MinSpareServers 5`
 - `MaxSpareServers 10`
- Les pages web ... emplacement
 - **Directive** `DocumentRoot`
 - `#cat sites-enabled000-default`
- Chargement des modules
 - Apache propose des modules logiciels offrant diverses fonctionnalités
 - voir `/etc/apache2/mods_enabled` qui contient des liens vers certains fichiers dans `mods_available`

- Journalisation
 - Préciser l'emplacement des fichiers *log* - `ErrorLog`
 - Quantité d'infos - `LogLevel`
 - Format des logs - `LogFormat`
- Contrôle d'accès
 - `Order deny, allow`
 - `Deny from all`
 - `Allow from esi.be`
 - Possibilité d'autoriser l'accès sur base de login/password ... voir directive `Auth*`

- Points NON abordés
 - Serveurs mandataires (*proxy*)
 - Sécurité / encryptage (certificat)
 - Hôtes virtuels

- **freemind** - <http://freemind.sourceforge.net>
Génération d'un Mind Map
- Génération des slides sur base du Map freemind
 - Scripts Perl
 - freemind2s5.pl de Vincent Oberle
 - freemind2beamer.pl modification incomplète du script
 - Format PDF
 - \LaTeX
 - package *beamer*