################ PAM ############################

Les modules PAM sont des bibliothèques dynamiques (par ex. pam_unix.so) fournissant les six primitives d'authentification définies dans la norme, regroupées dans quatre mécanismes :

Module type:
   Le mécanisme account fournit une seule primitive : il vérifie si le compte demandé est disponible (si le compte n'est pas arrivé à expiration, si l'utilisateur est autorisé à se connecter à cette heure de la journée, etc.).

   Le mécanisme auth fournit deux primitives ; il assure l'authentification réelle,
      éventuellement en demandant et en vérifiant un mot de passe, et il définit des « certificats d'identité » tels que
      l'appartenance à un groupe ou des « tickets » kerberos.

   Le mécanisme password fournit une seule primitive :
      il permet de mettre à jour le jeton d'authentification (en général un mot de passe),
      soit parce qu'il a expiré, soit parce que l'utilisateur souhaite le modifier.

   Le mécanisme session fournit deux primitives : mise en place et fermeture de la session.
      Il est activé une fois qu'un utilisateur a été autorisé afin de lui permettre d'utiliser son compte.
      Il lui fournit certaines ressources et certains services,
      par exemple en montant son répertoire personnel, en rendant sa boîte aux lettres disponible, en lançant un agent ssh, etc.

Control type:
      required:
      requisite: voir ### mdp et pam ###
      sufficient
      optional:

On peut égalemeent spéicifier des actions :
      ignore
      bad
      die
      ok
      done
      reset
Par exemple :
       required = [success=ok new_authok_reqd=ok ignore=ignore default=bad]

Module Path:
      /lib/security/

Sur debian 6 : le plus important : dans /etc/pam.d/
            common-auth
            commont-account
            commont-password
            common-session
            common-sesseion-noninteractive (pour les interactions sans shell

This PAM library is configured locally with a system file, /etc/pam.conf (or a series of configuration files located in /etc/pam.d/) to authenticate a user request via the locally available authentication modules.

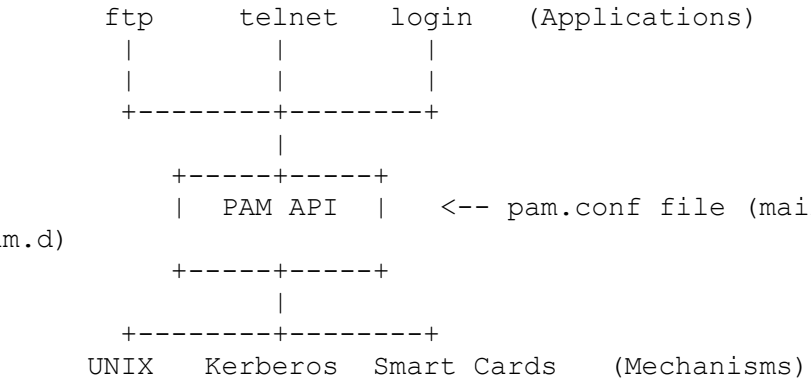

Figure 1: The Basic PAM Architecture

the PAM API loads the appropriate authentication module as determined by the configuration file, `pam.conf'

that PAM loadable object files (the modules) are to be located in the following directory: /lib/security/ or /lib64/security
Linux-PAM deals with four separate types of (management) task. These are: authentication management; account management; session management; and password management.

Exemple:

```
auth      required  pam_unix.so shadow nullok
```

module prompts the user for a password and then checks the password using the information stored in /etc/passwd and, if it exists, /etc/shadow.
The pam_unix.so module automatically detects and uses shadow passwords to authenticate users. pam_unix - Module for traditional password authentication

nullok instructs the pam_unix.so module to allow a blank password.
The argument nullok instructs the module to allow the user to change their password from a blank password, otherwise a null password is treated as an account lock.

User no passwd: if your user has sudo privileges,
you must enable its NOPASSWD option in the sudoers file. Otherwise, sudo will ask for a password even when you don't have one, and won't accept an empty password.

/etc/nologin
l'administrateur doit effectuer certains travaux de maintenance, il ne faut pas que les utilisateurs puissent se connecter au système.
Exemple de fichier:
cat nologin

Systeme indisponible jusqu'à 13:00 (maintenance)

Si quelqu'un tente de se log au démarrage, il verra se message si user != root

Configuration Debian

/etc/pam.conf ignoré comme /etc/pam.d existe

################################ SHELL
##

I\ Histoire

Shells—or command-line interpreters—have a long history,
but this discussion begins with the first UNIX® shell. Ken Thompson (of Bell Labs) developed the first shell for UNIX called the V6 shell in 1971.

The Bourne shell (1977 bsh/sh) led to the development of the Korn shell (ksh), Almquist shell (ash), and the popular Bourne Again Shell (or Bash) and the C shell (csh)

########################## TP
##

Pour lister les shell installés:
cat /etc/shells

en mettre un par défaut: chsh -s /bin /bash par exemple

Le shell par défaut est indiqué par user dans /etc/passwd
Pour switch d'un shell à un autre, il faut juste lancer la commande de lancement de shell

Pour vérifier qu'une commande existe:
command -v ls
echo $?
Si retourne 0, la commande existe
Sinon n'existe pas

La commande:
command -v : affiche le contenu de l'alias/le chemin d'accès à la commande
command -p : Exécute directement la commande indiquée depuis le Path sans passer par les alias
Donc command -p ll ne marchera pas car il recherche la commande ll dans le path
et command -p ls fonctionnera

On peut supprimer les accents en ligne de commandes avec tr/sed

```
      chaine="éééééééé"
      echo $chaine | iconv -f utf8 (format de base) -t (to)
ascii//TRANSLIT

ou à partir d'un fichier vers un fichier de sorti
      iconv -f utf8 -t ascii//TRANSLIT <fichierinput> fichieroutput

$IFS:
      Séparateur standard du shell
      Internal Field Separator
      Définit charactères pour délimiter les mots dans une chaine de
caractère
      Pour initialiser l'IFS:
            $IFS=$' \t\n'
      Avec IFS on peut découper par ':' (fichier /etc/passwd)
      Les variables $@ et $* permettent d'afficher la liste des arguments
passés à un script shell
MAJ/MIN:

      fhh@mafalda ~ $ MyString="coMMent AlleZ voUs ?" ;
      fhh@mafalda ~ $ echo "En minuscule : \"${MyString,,}\"" ;
      En minuscule : "comment allez vous ?"
      fhh@mafalda ~ $ echo "En majuscule : \"${MyString^^}\"" ;
      En majuscule : "COMMENT ALLEZ VOUS ?"

      Il faut spécifier ,, ou ^^ pour l'affichage

BIPPER:

      rmmod -v pcspkr
      cat /etc/modprobe.d/blacklist.conf
      blacklist pcspkr

      ou sans droit root
      setterm -blength 0

LOG USERS:
      lastlog

          List the last login time of all system users. This references
the /var/log/lastlog file.


Shell interactif:
      On parle de shell interactif si lit/ecrit à partir d'un terminal
      echo $- -> contient i alors interactif

      Test expression  [[ et commande built-in [ (echo, bind...)
      Le shell exécute commande spécifié par !# /bin/...


BASH
      ~/.bash_profile  Per-user, after /etc/profile. If this file does
not exist, ~/.bash_login and ~/.profile are checked in that order. The
skeleton file /etc/skel/.bash_profile also sources ~/.bashrc.
      ~/.bash_logout   After exit of a login shell.
      /etc/profile     Sources application settings in
/etc/profile.d/*.sh and /etc/bash.bashrc.
      ~/.bashrc  Per-user, after /etc/bash.bashrc.
```

TRAPS:
. The trap statement catches these sequences and can be programmed to execute a list of commands upon catching those signals.
The return status of the trap command itself is zero unless an invalid signal specification is encountered

```
#!/bin/bash
# traptest.sh

trap "echo Booh!" SIGINT SIGTERM
echo "pid is $$"

while :                # This is the same as "while true".
do
      sleep 60    # This script is not really doing anything.
done
```

A chaque fois qu'on fera un CNTRL C -> affiche booh

########################## Création d'user ##################

useradd -m -g initial_group -G additional_groups -s login_shell username

L'option -m : crée home directory

passwd username

Fichier important: /etc/login.defs Configuration control definitions for the login package
(UID Min, Max, Passwd max/min days,login retries...)

############## MDP ET PAM ##############################

required: tous les modules utilisant ce controle doivent passer par le succès
pour que la vérif soit accordé

requisite: Comme required sauf que l'utilisateur est averti immédiatement

optionnal

sufficient

Les modules se configurent dans /etc/security

Donc le fichier pwquality.conf
(man pam_pwquality)

Une application non définie -> /etc/pam.d/other
Exemple:
```
auth required pam_deny.so
account required pam_deny.so
password required pam_deny.so
session required pam_deny.so
```

Install module -> mov /lib/security pam*.so

################## Information Users ####################

Les informations des users sont stockés dans
	/home/
	/etc/sudoers
	/etc/group
	/ec/gshadow shadow pour groupe
	/etc/passwd/
	/etc/shadown

User database:
	account:password:UID:GID:GECOS:directory:shell
	GECOS : optional field informational purpose

commande chfn/usermod/

	Add user to other group:
		usermod -aG additionalGroup username
		(Si pas de -a, le user est retiré des groupes non indiqué en param dont il est déjà présent :/)

	Mark password as expired (chage modifie les informations de validité d'un passwd)
		-d lastday
		-E Date fin validitéqq
		chage -d 0 username : force le user a changer son mdp à login

	Tout supprimer du user:
		userdel -r username (-r : home directory, mail also del)

	Vérifier l'intégrité des fichiers de mdp
		pwck -s

Pour voir les valeurs attribués par défaut par useradd:
 /etc/default/useradd

pour décrire un utilisateur: chfn

Pour le chiffrement des mdp: la commande /user/sbin/pwconv transfère mdp crypté dans /etc/shadow

#################### Information Groupes #############"

	/etc/group
	/etc/gshadow

Chaque groupes peuvent avoir des admin, membres et mdp
	gpasswd -A pour définir admin groupe, -M membres

	groupmod, groupdel,groupadd,groupmems (groupe primaire), groups, grpck(vérifie intégriter des groupe),

Chaque utilisateur doit faire partie d'un groupe, son groupe primaire, définit à l'initialisation par défaut.
Ce groupe est spécifié dans /etc/password par le gid.

	groupdel userir : supprime un groupe au hasard

/etc/group:
	4 champs: nom du groupe: x pour remplacer un mdp non attribué:gid:liste membre du groupes

################ Variable d'environnements ####################

Pour regarder l'ensemble des variables d'environnements: printenv

On peut run des commandes à travers les variables d'environnement:

	env $EDITOR=vim xterm
		permet de lancer un shell xterm sans modifier la valeur de la variable EDITOR

On peut définir des variables d'environnements locale à l'user:
	par /etc/security/pam_env.conf -> dans /etc/environment EDITOR=NANO (variable=valeur)

		bash configuration (bash_profile)

		~/.profile

		systemd par ~/.config/environment.d/*

NOTE: mail config dans : ~/.config/environments/emaildefaults

################ Link ##################################

Hard link et le fichier originel ne sont pas distinguable, possède même inode

############ TIME ###################################

Hardware clock (RTC Real Time Clock) :
	hwclock --show

Set HW clock from Sys clock
	hwclock --systohc

SoftW Clock (System) calcule depuis le kernel en seconde depuis 1st Janvier 1970 UTC
	/etc/adjtime

	Regarder l'heure:
		timedatectl

	Mettre l'heure à jour
		timedatectl set-time "2014-05-26 11:13:54"

Deux types de normes:
	localtime: dépend de la timezone
	UTC (Coordinated Universal Time): global time standard, indépendant des zones.

	Pour mettre le HW Clock en mode localtime : timedatectl set-local-rtc 1 (0 pour UTC)

Tout dépend de /etc/adjtime, si non présent, systemd suppose HW clock en UTC