
Administration Unix

Le cas de Solaris 9

Ronan.Keryell@enst-bretagne.fr

Laboratoire Informatique & Télécommunications

Département Informatique

École Nationale Supérieure des Télécommunications de Bretagne

<http://www.lit.enstb.org/~keryell>

8–10 octobre 2003

Administration système ?

1

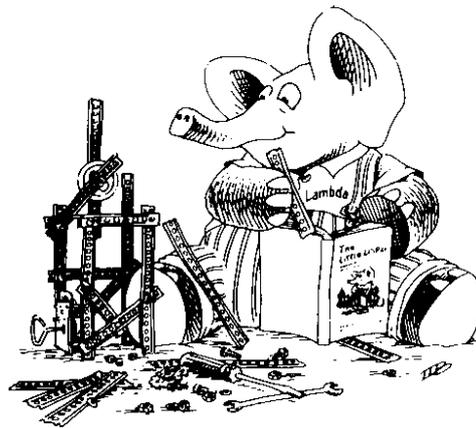
Assurer un service informatique, robuste, efficace

- Garder ordinateurs et réseaux en état de marche
- Installer ordinateurs, système d'exploitation et nouveaux logiciels
- Installer réseaux informatique
- Faire sauvegardes de sécurité et restaurations
- Créer et effacer comptes utilisateurs
- Gérer le quotidien, réparer les défauts qui peuvent survenir
- Définir des politiques d'usage et les procédures
- Éthique : respect de la vie privée,...



- Tour de table
- Rappels sur Unix
- Administration
- Installation automatique (jumpstart de Solaris)
- Configuration automatique (cfengine de GNU)
- Mise en pratique

Voir la
table des matières cliquable



Beaucoup d'hyperliens dans ce cours



- Unix est un système d'exploitation performant, portable, universel et très complet
- ... et (logiquement) complexe
- ⇔ Administration pouvant être assez compliquée car tout est possible et configurable
- ⚠ Sirènes graphiques : une interface graphique ne fait que cacher la complexité qui réapparaît en cas de problème...
- Nécessité de comprendre comment cela fonctionne !
- Cours plutôt ciblé Solaris mais ceux avec des machines SVR4 et mêmes les autres peuvent être concernés



- Années 60 : collaboration de Bell Telephone Laboratories de AT&T, General Electric et MIT sur le projet de système d'exploitation multi-utilisateur *Multics* : *MULTiplexed Information and Computing Service*
 - ▶ MIT, Bell Telephone Laboratories de AT&T, General Electric
 - ▶ Notion de « Computer Grid » (qui revient de nos jours...)
 - ▶ Offrir puissance de calcul pour toute la ville de Boston : le Minitel avant l'heure
 - ▶ Plus difficile que prévu ~> abandonné en 1969 mais grande influence dans la communauté

<http://www.multicians.org/>

- En attendant la suite, Ken Thomson de BTL écrit un jeu *Space Travel* qu'il fait tourner sur un PDP-7 (machine pas trop chère)



- Problème : pas d'environnement de développement sur PDP-7 et nécessité de faire de l'assemblage croisé sur Honeywell 635 roulant GECOS
- Pour faciliter le développement du jeu, développement d'un système d'exploitation pour le PDP-7 : système de fichier simple (*s5fs*), système de gestion de processus, interpréteur de commande (*shell*)
- Le système devient auto-suffisant et est nommé *Unix* en 1969, jeu de mots en opposition à *Multics*
- Portage d'Unix sur PDP-11 et développement de l'éditeur de texte *ed* et du système de composition de texte *runoff*
- Développement de langage interprété *B* utilisé pour développer les outils



- Dennis Ritchie fait évoluer le langage en C dont le succès a largement dépassé le cadre d'Unix
- 1972 : 10 machines sous Unix...
- Unix réécrit en C en 1973 et la distribution version 4 contient elle-même cc
- L'université de Berkeley récupère une licence (gratuite à cause d'un procès antitrust de 1956 entre AT&T et Western Electric Company)
- La version 7 de 1979 est la première version réellement portable
- Beaucoup d'améliorations fournies par les utilisateurs eux-mêmes (de même que BSD & Linux maintenant) favorisé par le côté non commercial
- MicroSoft et Santa Cruz Operation collabore sur un portage pour



i8086 : Xenix

- Portage sur machine 32 bits (Vax-11) en 1978 : UNIX/32V qui est récupérée par Berkeley
- Rajout d'utilitaires (*cs* de Bill Joy) et d'un système de pagination
- La DARPA donne un contrat à Berkeley pour implémenter IP : BSD
 - ▶ Dernière version en 1993 : 4.4BSD. En tout : apport des *socket*, d'IP, d'un *fast file system*, des signaux robustes, la mémoire virtuelle
 - ▶ Société BSDI créée pour vendre 4.4BSD *lite* en 1994, débarrassé de tout code d'origine AT&T
- Le premier PC 1981
 - ▶ Grand retour en arrière (...pour les spécialistes) :



mono-programmation : MS-DOS 1.0, PC-DOS 1.0

- ▶ Système de fichier primitif (...mais robuste), pas de répertoire,...
- 1982 : loi antitrust qui éclate AT&T en baby-Bell dont le AT&T Bell Laboratories qui peut alors commercialiser Unix
 - ▶ 1982 : System III
 - ▶ 1983 : System V
 - ▶ 1984 : System V release 2 (SVR2)
 - ▶ 1987 : System V release 3 (SVR3) introduit les IPC (InterProcess Communications : mémoire partagée, sémaphores), les *STREAMS*, le *Remote File Sharing*, les bibliothèques partagées,...
 - ▶ Base de nombreux Unix commerciaux
- 1982 : Bill Joy quitte Berkeley pour fonder Sun Microsystems.



Adaptation de 4.2BSD en SunOS qui introduit le *Network File System*, interface de système de fichier générique, nouveau mécanisme de gestion mémoire

- Milieu des années 1980 à Carnegie-Mellon University développe Mach, un micro-noyau avec des serveurs implémentant une sémantique 4BSD. OSF/1 & NextStep sont basés sur Mach
- 1987 Andrew Tanenbaum publie « *MINIX: A UNIX Clone with Source Code for the IBM PC* »
<http://www.cs.vu.nl/~ast/minix.html>
- 1987 : AT&T achète 20% de Sun ~> prochaines version de SunOS basées sur System V : SunOS 5 (Solaris 2)
- 1989 : co-développement AT&T-Sun de SVR4 : inclut les fonctionnalités de SVR3, 4BSD, SunOS & Xenix. Création d'Unix Systems Laboratories pour développer et vendre Unix



- Novell achète une partie d'USL en 1991 pour développer UnixWare (Unix + Netware) et tout USL en 1993
- Arrivée de NT
- Linus Torwald récupère Minix sur PC i80386 et le développe en Linux en 1991
http://www.dina.dk/~abraham/Linux_vs_Tanenbaum.html
- Dernière version de SunOS version BSD : 4.1.4
- 1994 : première version publique de Linux : 1.0.
Développements pris en main par des programmeurs répartis sur Internet. Intégration des utilitaires GNU
- Arrivée de NT (*New Technology*) : mélange de MS-DOS, MacOS, VMS et Unix
- Novell cède la marque Unix au X/Open puis Sun rachète les



droits de SVR4 à Novell en 1994

- Chorus, société française, développe un micro-noyau
- Solaris 2.6 : NFSv3, version de Solaris 2... qui fonctionne ☺ et justifie l'abandon de SunOS4
- Rachat de Chorus par Sun. ↔ JavaOS ?
- Un système Unix \equiv programmes utilisateurs + bibliothèques + utilitaires + système d'exploitation qui fournit le support d'exécution et les services
- Unix tourne sur toutes les plates-formes depuis les systèmes embarqués jusqu'aux supercalculateurs massivement parallèles
- Solaris 7 : version 64 bits pour UltraSPARC et IA64
- 2000 : Windows 2000 puis Windows XP : plus stable, plus gros,...



- 2000 : Solaris 8, plus stable, plus gros,...
- 2002 : Solaris 9
- 2003 : Solaris 10 bêta, Gnome 2



Sources d'information

- Les manuels `man [-s section] nom`
- BigAdmin[sm] portal for system administrators
<http://www.sun.com/bigadmin/>
- L'« AnswerBook » local <http://machine:8888>
- <http://docs.sun.com> : site de référence de la documentation de Sun
- La documentation papier « *Solaris 9 System Administrator Collection* » <http://docs.sun.com/?p=coll/47.13> avec en particulier
 - ▶ Basic Administration (comptes utilisateurs, boot, logiciels, disques) <http://docs.sun.com/?p=/doc/806-4073>
 - ▶ Advanced Administration (imprimantes, disques, terminaux, contrôle du système) <http://docs.sun.com/?p=/doc/806-4074>



- ▶ System Administration Guide: Naming and Directory Services (DNS, NIS, and LDAP) <http://docs.sun.com/?p=/doc/806-4077>
- Sun[tm] SupportForum
<http://supportforum.sun.com/freesolaris/index.html>
- Revue Server/Workstation Expert <http://sun.expert.com/>
- Les FAQ (*Frequently Asked Questions* ou encore *Foire Aux Questions*)
 - ▶ <http://www.pasteur.fr/other/computer/FAQ/>
 - ▶ General FAQ about Solaris on the Intel Architecture platform
<http://www.sun.com/software/intel/faq.html>
 - ▶ Solaris on Intel - x86 FAQ <http://www.sun.drydog.com/faq/>
 - ▶ Technical FAQ about Solaris Operating Environment
<http://supportforum.sun.com/freesolaris/techfaqs.html>



- Les News
 - ▶ `news:comp.sys.sun`
 - ▶ `news:comp.unix.solaris`
 - ▶ `news:comp.sys.sun.admin`
 - ▶ `news:comp.sys.sun.announce`
 - ▶ `news:comp.sys.sun.apps`
 - ▶ `news:comp.sys.sun.hardware`
- Sources de Solaris <http://www.sun.com/software/solaris/source>
- La « hotline » de Sun
- Livre *Essential System Administration*, Aileen Frisch, 2nd Edition
September 1995 1-56592-127-5, Order Number: 1275 788
pages, O'Reilly, \$34.95 <http://www.oreilly.com/catalog/esa2/>
- Livre « Unix Internals —The New Frontiers », Uresh Vahalia,



Prentice Hall

- <http://ftp.univ-lyon1.fr/doc/francais/admin-cookbook/> : le livre « *Administration Système Unix* » de Thierry Besançon, Pierre David et Joël Marchand
- <http://www.freenix.fr/~dumas/linux/Guide/> : « *Le guide du ROOTard pour Linux* »



Distribution Solaris 9

- Système d'exploitation au choix
 - ▶ 1 DVD
 - ▶ Ou plein de CD
 - 1 CD *Solaris 9 Installation*
 - 2 CD *Solaris 9 Software 1 & Solaris 9 Languages 2*
 - 1 CD *Solaris 9 Documentation*
- 1 CD *Solaris 9 Software Supplement* (Java3D, SunForum, PCLauncher, PCfile viewer, ShowMeTV,...)
- 1 CD *Solaris Software Companion* (logiciels libres)
- 1 CD *Sun Management Center 3.0*
- 1 CD *Oracle*
- 1 CD *Gnome 2*



- 1 CD *StarSuite*
- 1 CD *StarOffice*
- 1 CD *iPlanet LDAP*
- 1 CD *Forte 6* (C, C++, Fortran, bibliothèques performantes, debug, TeamWare)
- 1 CD *Forte for Java*

Images aussi récupérables sur sun.com



Quoi de neuf ou mieux dans Solaris 9 ?

<http://docs.sun.com/db/doc/806-5202>

- Système :
 - ▶ *Resource Manager* : contrôle plus fin des ressources (pool de processus)
 - ▶ Nouvel ordonnanceur équitable
 - ▶ Nouvelle classe de priorité fixe,...
 - ▶ Solaris Management Console gère le RAID
 - ▶ Attributs (cachés) dans les systèmes de fichiers pour associer des choses arbitraires à un fichier (icône,...) (cf *runat* et *openat*)
 - ▶ Solaris Live Upgrade : prépare une nouvelle installation sur une autre partition
 - ▶ Web Start Flash : clone une installation, version graphique ou ligne de commandes



- Sécurité :
 - ▶ IKE pour les échanges de clés publiques IPsec
 - ▶ OpenSSH en standard
 - ▶ LDAP avec TLS pour remplacer NIS+
 - ▶ Cryptographie plus forte
 - ▶ inetd sécurisé à la tcpwrapper
- Réseau :
 - ▶ Netscape 6
 - ▶ IP network multipathing
 - ▶ RPC asynchrone
 - ▶ iPlanet LDAP
- Encore plus d'outils libres (GNU,...)
- GNOME en plus de CDE comme système de fenêtrage



- Compatibilité Linux au niveau des binaires



- Axiome unixien : fichiers de configuration au format texte
- Facile de les modifier avec un éditeur de texte
- Utilisation d'outils graphiques de Sun (admintool, solstice) ou d'autres plus génériques (<http://www.webmin.com/webmin/>). Jolis et pratiques, mais pas toujours portables
- Écriture de scripts de manipulation automatique des fichiers de configuration
- ∃ Outils d'automatisation libres tout faits (cfengine,...)
-  Certaines ressources sont centralisées (NIS, LDAP, FNS,...) et définies sur des serveurs : les administrer sur ces serveurs !



Admintool

23

The screenshot displays the Admintool graphical user interface with several windows open:

- Admintool: Users**: Shows a table of system users.

User Name	User ID	Comment
nobody	60001	Nobody
nobody4	65534	SunOS 4.x Nobody
nucap	0	ucap AdmLin
sys	3	ucap AdmLin
ucap	5	ucap AdmLin
- Admintool: Hosts**: Shows a table of network hosts.

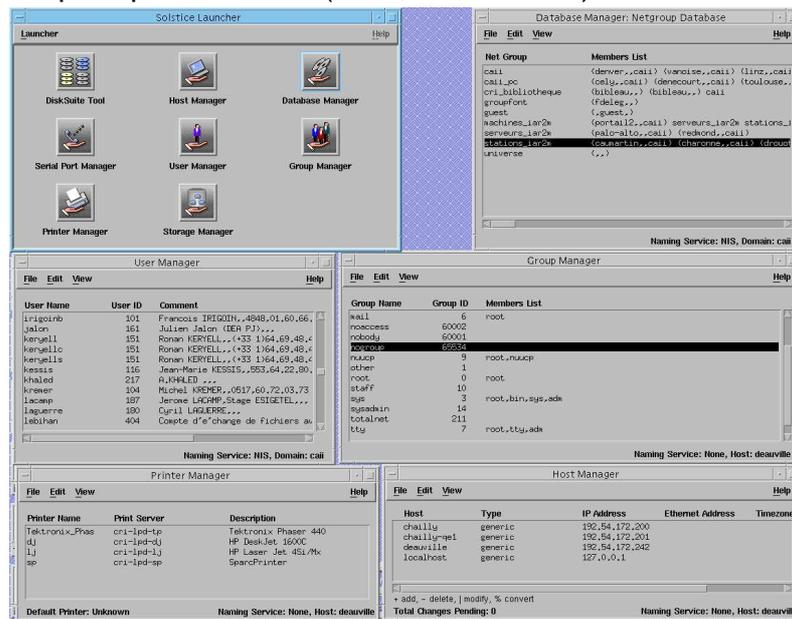
Host Name	IP Address
beaumont	10.2.16.11
channon	10.2.16.1
group1	10.2.16.9
franlin	10.2.16.14
havre	10.2.16.10
localhost	127.0.0.1
- Admintool: Modify User**: Shows configuration for the 'root' user, including User Name, User ID, Primary Group, Secondary Groups, Comment, Login Shell, and Account Security settings.
- Admintool: Printers**: Shows a table of installed printers.

Printer Name	Server	Description
Tektronix.Phase	cri-lpd-tp	Tektronix Phaser 440
dl	cri-lpd-dl	HP DeskJet 1600C
lj	cri-lpd-lj	HP Laser Jet
sp	cri-lpd-sp	SharpPrinter
telegraphe	nationalto	HP LJ 4000N
- Admintool: Add Local Printer**: A dialog box for adding a new printer, with fields for Printer Name, Printer Server, Printer Port, Printer Type, File Contents, and User Access List.
- Admintool: Software**: Shows a list of installed software packages.
- Admintool: Groups**: Shows a table of system groups.

Group Name	Group ID	Members
adm	4	root,adm,daemon
bin	2	root,bin,daemon
daemon	12	root,daemon
lp	9	root,lp,lpd



Plus complet qu'admintool (local, NIS & NIS+)



Administration Solaris
DÉPARTEMENT INFORMATIQUE — ENST BRETAGNE

—Outils—



Solaris Management Console

25

Le plus complet

- Gère machines et (sous-)réseaux
- Centralise les informations et états
- Gère l'usage par projets
- Traitements par lot
- Disques & RAID
- Ports de communication
- Gère les patches



Administration Solaris
DÉPARTEMENT INFORMATIQUE — ENST BRETAGNE

—Outils—



- √ Système d'exploitation, faire marcher une machine ne suffit pas... ☹
- Penser à gérer de manière globale le système informatique & réseau
- Déployer une infrastructure
 - ▶ Extensible, flexible, déploiement et changements rapides
 - ▶ Efficace, économique : retour sur investissement
 - ▶ Environnement sécurisé et robuste
 - ▶ Référence formant le « source » du système
 - ▶ Administration à distance
 - ▶ Communications sécurisées
 - ▶ Surveillance distribuée



- ▶ Installation automatique par le réseau
- ▶ Administration unifiée serveurs et machines de bureau
- ▶ Espace de nommage uniforme
- ▶ Authentification uniforme
- ▶ Gestion des machines par classes
- Exemples :
 - ▶ Infrastructures <http://www.infrastructures.org>
 - ▶ Arusha <http://ark.sourceforge.net>



- Politique
 - ▶ Décrit organisation système
 - ▶ Comment ça marche ?
 - ▶ Ce qui est géré ou pas ?
 - ▶ Qui fait quoi ?
- Outil(s) et fichiers de configuration
 - ▶ Fait le vrai travail
 - ▶ Outils utilisent des fichiers de configuration files (≈ programme) mais aussi des fichiers de référence (≈ données)
- Fichiers de référence
 - ▶ Fichiers spécifiques au système d'exploitation



- ▶ Fichiers spécifiques au matériel
- ▶ Fichiers spécifiques au site
- ▶ Fichiers spécifiques au sous-réseau
- ▶ Fichiers spécifiques au projet
- ▶ Fichiers personnels
- ▶ Fichiers invariant
- ▶ ...

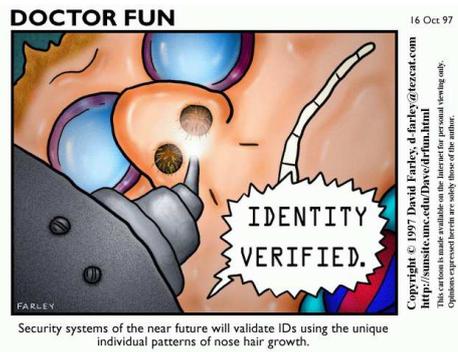


- Unix : fichiers de configuration textuels
- Possibilité de garder trace des évolutions de ces fichiers si utilisation d'outils de gestion de version
 - ▶ SCCS : ancien mais installé par défaut
 - ▶ RCS : plus récent, plus performant (notion d'« instantanés ») et base de CVS, mais à installer ou sur la distribution de *freeware* Solaris
- Gèrent le travail coopératif : utile de savoir qui a fait quoi
- Outils bien empaquetés dans `emacs` ~> avoir toujours un `emacs` dans un coin qui tourne sous `root`
 - ▶ `Menu Tools/Version Control`
 - ▶ `C-x v i` : entre un fichier dans la gestion de version



- ▶ `C-x v v` : passe un fichier en mode édition ou en lecture seule
- ▶ `C-x v l` : affiche l'histoire du fichier
- Possible de créer dans les répertoire un répertoire RCS qui recevra les fichiers « ,v » plutôt que de trop salir les répertoires courants
- Marche aussi avec le mode `tramp` d'accès transparent aux fichiers à distance :
`/root@machine:/un/fichier`
- GRAND est `emacs` ~> bon investissement... ☺





- Nécessité de rajouter des verrous pour protéger les systèmes et cerner les

responsabilités

- Besoin croissant avec les possibilités d'accès distants
- Possibilité de s'identifier par
 - ▶ Ce qu'on sait : mot de passe
 - ⚠ Espionnage...
 - ▶ Ce qu'on a : clé, carte à puce, ...
 - ⚠ Vol...
 - ▶ Ce qu'on est : empreintes digitales, fond de l'œil

Administration Solaris—Responsabilités utilisateur—



DÉPARTEMENT INFORMATIQUE — ENST BRETAGNE



⚠ Mutilations,...

Intérêt d'utiliser simultanément plusieurs techniques complémentaires

- Compte utilisateur de base sous Unix : association
 - ▶ Nom d'utilisateur (*login*)
 - ▶ Mot de passe
- Piratage : connaître les 2
- Nom d'utilisateur mnémotechnique (envoi/réception de courriel avec,...) et généralement en

rapport avec le nom de la personne : assez simple à deviner

- Reporter le blindage sur le mot de passe
- Généralement enregistrement des authentications ainsi que des échecs successifs
- Possible de verrouiller les comptes si trop d'échecs sur certains systèmes
 - ⚠ ~> possibilité de refus de service en générant

Administration Solaris—Responsabilités utilisateur—



DÉPARTEMENT INFORMATIQUE — ENST BRETAGNE



expressément plein d'authentifications invalides pour bloquer tous les comptes....

- Délai croissant rajouté après chaque échec pour limiter le nombre d'essais. Limite le refus de service
- Ne pas faire confiance aux courriels du style
 - ▶ « *Suite à un problème système, vous devez mettre trucmuche comme mot de passe.*

L'administrateur système. ». Le courriel peut être faux...

- ▶ « *Merci de nous envoyer votre mot de passe à telle adresse...* » Sans commentaire

 Cela marche parfois...

- L'administrateur système ne doit pas donner toujours le même mot de passe temporaire. Option pour forcer le changement de mot de passe lors de la première

Administration Solaris—Responsabilités utilisateur—



DÉPARTEMENT INFORMATIQUE — ENST BRETAGNE



identification

- L'administrateur doit vérifier l'identité de la personne venant demander un changement de mot de passe

- Lors d'un changement de mot de passe, tester le nouveau mot de passe *avant* de se déconnecter (via `su - mon-nom` ou `telnet localhost`). Surtout si `root...`



- Mauvais mot de passe : porte ouverte sur tout le système
- Éviter ce qui peut être deviné (et sera essayé par des programmes style crack !) :
 - ▶ Son propre mot de login ! Si la liste des utilisateurs est connue, attaque triviale...
 - ▶ Suites de touches clavier
 - ▶ Numéros de téléphones ou de plaques minéralogiques personnels
 - ▶ Noms/prénoms de l'environnement personnel
 - ▶ Mots de n'importe quelle langue à l'endroit ou à l'envers
 - ▶ Personnages/mots de romans, jeux,...
 - ▶ Des modifications simples des cas précédents : chiffre ou ponctuation avant ou après



- Bons mots de passes consistent typiquement en
 - ▶ Mélange lettres minuscules et majuscules
 - ▶ Chiffres et caractères de ponctuation en plus
 - ▶ Caractères spéciaux et espace
 - ▶ Longs (8 caractères par défaut sous Unix)
 - ▶ Faciles à retenir pour ne pas avoir besoin de les écrire
 - ▶ Utiliser l'espace des $4,3.10^{16}$ mots de passe Unix possibles
- Moyens mnémotechniques
 - ▶ Combinaison de mots avec caractères de ponctuation
 - ▶ Abréviation ou initiales d'une phrase
- Pour éviter d'avoir le même mot de passe sur différentes machines, construire mot de passe à partir d'un mot de passe de base + caractéristique du nom de la machine par exemple



- Ne pas écrire son mot de passe ou tout au moins pas sous forme triviale
- Ne pas envoyer de mot de passe en clair par réseau ou dans un fichier

```
find ... -exec egrep 'password|passwd|mot de passe' '{}'
```

 \;
- Ne pas utiliser le même mot de passe ailleurs, dans un serveur WWW,... Peut être espionné et pas forcément stocké crypté



Stockage des mots de passe

- Même si stockés dans un fichier lisible seulement par `root` (`/etc/shadow`) possibilité d'erreur de manipulation rendant publique le fichier ou piratage pour accéder aux mots de passe (pour corrompre d'autres machines)
- ↔ Stocker de manière cryptée les mots de passe
- La fonction de décryptage doit être très difficile (impossible)
- Usage : toute demande d'authentification par mot de passe, plutôt que de décrypter le mot de passe stocké, crypte le mot de passe et le compare à celui stocké
- Cryptage des mots de passe (de base) dans Unix :
 - ▶ Utilise DES pour encoder 1 bloc de 64 bits à 0 avec le mot de passe vu comme une clé de 56 bits
 - ▶ Réappliquer le DES sur le résultat avec toujours le mot de



passee comme clé

- ▶ En tout appliquer le DES 25 fois et stocker le résultat dans /etc/passwd ou /etc/shadow sous forme de 11 caractères ASCII codant chacun 6 bits
- ▶ Pas de technique connue d'inversion \rightsquigarrow craquage par dictionnaires pour éviter la recherche brute (2^{56} mais gros ordinateurs parallèles, spécialisés, ...)
- ▶ Tables du DES modifiées par rapport au DES classique pour éviter l'usage d'accélérateurs de DES classique
- ▶ Difficulté supplémentaire : le « sel » de 12 bits modifiant les tables du DES, choisi lors de l'établissement du mot de passe en fonction de l'heure et de la date et stocké aussi devant le mot de passe crypté sous forme de 2 caractères. Idée : oblige de pré-encoder 4096 fois le dictionnaire pour décrypter avec tous les sels possibles...



- Les ordinateurs font des progrès, FPGA programmables,... DES cassé en 22h en 1999 <http://www.eff.org/DESCracker>
- Algorithmes plus complexes utilisés sur certains Unix (via PAM,...)



- Souvent les mots de passe circulent en clair sur les réseaux si pas de cryptage des communications ~> programmes d'espionnage *sniffers*
- Pour éviter le vol de son mot de passe : ne plus utiliser de mot de passe réutilisable !
- ~> Utiliser des *One-Time Passwords* (OTP)
 - ▶ Liste imprimée de mots de passe qu'on raye après chaque usage
 - ▶ Calculatrice qui affiche le mot de passe à taper pour la minute courante
 - ▶ Code affiché lors de la connexion qui doit être tapé sur une calculatrice et qui donne un résultat à taper pour se connecter
 - ▶ Logiciel sur sa machine réalisant les fonctionnalités.  Si



des intrus ont la possibilité d'utiliser le logiciel sur la machine...

- Mais utilisation limitée par la nécessité de matériels spécifiques
-  Piles qui s'usent, oubliés à la maison...
- Remplacement du *shell* de login par un programme gérant le mot de passe jetable et validant ou pas la connexion



- Système de mots de passe jetables développé à Bellcore
- Stocker liste de mots de passe jetables de tous les utilisateurs : gros \rightsquigarrow Avoir un générateur de suites de mots de passe
- Utilisation d'une fonction cryptographique f qui donne une valeur à partir d'une entrée mais rend très difficile l'opération inverse
- L'itération de la fonction donne une suite de données difficilement prédictible
- Utilisation de chaque itération comme mot de passe jetable
- Le client et le serveur doivent partager la même fonction secrète
- Authentification :
 - ▶ Serveur demande au client le mot de passe p_n de l'itération n
 - ▶ Client doit répondre le mot de passe correspondant



- ▶ Forte authenticité si les mots de passes sont identiques
-  Si la fonction f et p_n sont piratés le système aussi :
 $p_{n+1} = f(p_n)$
- Comment changer son mot de passe ? Envoyer p_0 en clair ?
- Idée de Lamport en 1981 : regarder le problème à l'envers !
- Exploiter la difficulté de f^{-1} : trouver p_n en fonction de p_{n+1}
 - ▶ Serveur stocke mot de passe p_{n+1}
 - ▶ Client envoie $p_n = f^n(p_0)$ au serveur
 - ▶ Serveur calcule $p'_{n+1} = f(p_n)$
 - ▶ Si $p'_{n+1} = p_{n+1} \rightsquigarrow$ forte authenticité
 - ▶ Initialisation : client envoie $p_N = f^N(p_0)$. Espionnable sans danger
- RFC 1938



- Pour rajouter de la difficulté, rajout d'une graine intervenant dans le calcul qui est joint au n envoyé par le serveur comme challenge. Permet d'identifier sur différentes machines avec le même mot de passe mais différentes graines
- Utilisation des fonctions MD4 ou MD5
- Challenge : 64 bits. Exemple : E79F 3CA6 8C57 E381
- Moyen mnémotechnique si pas de couper-coller : encoder par groupe de 11 bits et définir une liste de 2048 mots standard.
Exemple : TANK WELT MOT HAL FATE MUSH



- *One-time Passwords In Everything*
- Implémentation de S/Key développée au United States Naval Research Laboratory (NRL)
- Remplace les commandes login, su et ftpd par leur homologue OPIE opielogin, opiesu et opieftpd
- Première connexion : utiliser opiepasswd pour mettre son mot de passe dans la base OPIE
- opieinfo donne le numéro de séquence et la graine pour un utilisateur
- opiekey *sequence_number seed* est une calculatrice à faire tourner sur le client : génère p_n à partir du mot de passe (secret) et de la graine et du numéro de séquence. Compiler opiekey sur toutes les machines depuis lesquelles on veut se connecter.



S/Key peut aussi tourner sur les calculettes programmables classiques (HP-48,...)

-  à ne pas rentrer le mot de passe dans `opiekey` ou `opiepasswd` à travers un médium espionnable ! Éviter faire tourner `opiekey` sur une autre machine via `rsh`, `rlogin` ou `X11`,... Problèmes pour les terminaux X !
- `opiepasswd` change son mot de passe en entrant la sortie de `opiekey`. 499 itérations par défaut
- `opieinfo` indique le numéro de séquence courant et la graine
- Si pas de calculette possible, imprimer une feuille ultra-secrète avec `opiekey -n 499 'opieinfo'` par exemple pour 499 secrets
- Bibliothèques pour rajouter OPIE dans d'autres programmes



- `/etc/opieaccess` pour faciliter les transitions en autorisant les connexions depuis certains réseaux/machines sans OPIE de façon normale.  trou de sécurité...
- `ftp://ftp.inner.net/pub/opie`
- `man opie`

Intégré aussi dans OpenSSH



Un compte utilisateur est défini par :

- Nom d'utilisateur utilisé à la connexion. 2 à 8 caractères, sans espace ni souligné, unique à travers tout le site administré (conflits sinon...)
- Mot de passe restreignant l'accès. 6 à 8 lettres. Éviter des mots de passe « trouvables ». Éviter tout ce qui peut être écrit quelque part ou combinaison triviale. Faire changer les mots de passe régulièrement. Ne pas réutiliser son mot de passe dans un autre contexte
- Répertoire personnel (\$HOME ou ~) qui contiendra les fichiers de l'utilisateur (qui pourra bien sûr avoir aussi d'autres répertoires à lui ailleurs)
- Des fichiers d'initialisation lus par le shell lors d'une connexion
- Un numéro d'utilisateur utilisé en interne pour faire référence à



l'utilisateur. ⚠ Éviter les conflits dans une grosse organisation.

Allocation des numéros

- ▶ 0 à 99 : comptes réservés au système, root (0), daemon, bin, sys,... pour cerner les droits
- ▶ 60001 : nobody
- ▶ 60002 : noaccess
- ▶ Le reste de 100 à 2147483647 : utilisateurs standards.
Utiliser ce grand espace de numérotation pour instaurer une hiérarchie (regrouper les utilisateurs par services, etc.
Exemple 100xy pour les utilisateur de l'équipe système, etc.).

⚠ Quelques problèmes de compatibilité avec de vieux Unix pour les gros numéros ≤ 60003 ...

Éviter de ré-allouer un ancien numéro : il peut rester des fichiers de l'ancien utilisateur qui seront alors accessibles au nouvel



utilisateur...

- Un numéro de groupe primaire d'utilisateurs partageant une unité de droit d'accès. Exemple : tous les membres d'une même équipe sont mis dans le groupe `equipe` et peuvent accéder aux fichiers réservés à l'équipe en lecture seulement pour le groupe `equipe`
- des groupes secondaires pour des projets communs
- `id -a` permet d'afficher son identificateur et ses groupes
- Par défaut, lorsqu'un utilisateur crée un fichier il lui appartient : il possède son numéro d'utilisateur et son numéro de groupe primaire



Numéros d'utilisateurs et de groupes Unix

- Numéro d'utilisateur utilisé en interne pour représenter un utilisateur selon la table `/etc/passwd`
-  `root` utilisateur spécial qui peut *tout* faire : administrateur
 - Trop puissant en Unix standard ?
 - Beaucoup de services ne peuvent tourner que sous `root`
 -  Si bug, possibilité de devenir `root`...
 - \rightsquigarrow Se défendre contre un `root`
 - ▶ Crypter les informations sensibles \rightsquigarrow `root` ne peut pas retrouver les mots de passe dans `/etc/shadow`
 - ▶ Utiliser des média à lecture seule (CD-ROM) non écrivable par `root`
 - ▶ Avoir des sauvegardes à jour !
 - ▶ Monter des disques en lecture seule (plus difficile pour



écrire : via le *device*)

- Autres systèmes d'exploitations avec subdivision de la puissance de `root`. ∃ Unix sécurisés. ⚠ Mais la subdivision n'empêche pas une certaine transivité et de toute manière dès qu'on accède aux devices...
- Autres pseudo-utilisateurs contrôlant des services : `lp`, `uucp`, `http`,...
- Plusieurs comptes utilisateurs avec le même numéro :
 - ▶ Partager la même identité mais avec plusieurs mots de passe : ↗ traçabilité
 - ▶ Comptes avec choix entre plusieurs shells
- Utilisateurs appartiennent à 1 ou plusieurs groupes en fonction de `/etc/passwd` et `/etc/group`



- ⚠ ↔ Bien vérifier qu'`/etc/passwd` et `/etc/group` ne peuvent être écrits que par `root`



-  Attention à la notion de comptes locaux ou de comptes dont la définition est centralisée par un service de nommage (NIS, NIS+, LDAP, FNS). Dans ce dernier cas il faut demander à aller modifier l'information directement à la base
- Modifier le fichier `/etc/passwd` à la main et éventuellement `/etc/group` (fichiers texte !).  Conflit si plusieurs personnes modifient ces fichiers simultanément (géré par exemple par `emacs` ou la version spécialisée `vipw` de `vi`). Penser à mettre à jour aussi `/etc/shadow` le cas échéant
- Outils graphiques : `admintool` (utilisateurs locaux) et `solstice` (utilisateurs locaux et centralisés)
- Outils textuels
 - ▶ Utilisateurs : `useradd/usermod/userdel`



- ▶ Groupes : `groupadd/groupmod/groupdel`
- ▶ Changement de mot de passe avec `passwd`
- ▶ Avec NIS+ utilisation de `nistbladm` et `niscat` pour visualiser une ressource
- ▶ Avec NIS visualiser la liste via `ypcat passwd` et `ypcat group`. Pour propager des changements depuis le serveur, penser à faire un `make` dans `/var/yp` sur le serveur



- `passwd [login]` : pour changer un mot de passe, utilisable par les utilisateurs
 - ▶ `passwd -e` permet de changer de shell (chsh d'autres systèmes)
 - ▶ `passwd -g` permet de changer les informations nominales (GECOS) (chfn d'autres systèmes)
- `useradd [-c comment] [-d dir] [-e expire] [-f inactive] [-g group] [-G optgroup[,optgroup...]] [-m [-k skel_dir]] [-u uid [-o]] [-s shell] [-A authorization[,authorization...]] [-P profile[,profile...]] [-R role[,role...]] login` : rajoute un utilisateur. Compte bloqué jusqu'au changement de mot de passe avec `passwd`
Exemple minimal créant le répertoire :



```
useradd -m untel
```

- `useradd -D ...` : modifie les valeurs par défaut de certains paramètres de `useradd`
- `userdel [-r] login` : efface un utilisateur
- `usermod [-u uid [-o]] [-g group] [-G group[,group...]] [-d dir [-m]] [-s shell] [-c comment] [-l new_name] [-f inactive] [-e expire] [-A authorization[,authorization]] [-P profile[,profile]] [-R role[,role]] login` : modifie un compte
- `groupadd [-g gid [-o]] group` : crée un groupe
- `groupdel group` : efface un groupe
- `groupmod [-g gid [-o]] [-n name] group` : modifie les caractéristiques d'un groupe



Dans le cas de systèmes de nommage compliqués, intérêt à bâtir l'infrastructure au dessus de ces outils.

```
useradd -m -k /usr/local/share/conf/etc/skel -s /usr/local/bin/tcsh \  
-d /users/enstb/info-invite/janiak -g info-invite janiak
```

 Pas mal de bugs dans ces commandes... ☹



Le fichier /etc/passwd

58

- Déclare un compte utilisateur
- Syntaxe (`man -s 4 passwd`) :
username:password:uid:gid:gecos-field:home-dir:login-shell
username : nom de login
passwd : mot de passe crypté de l'utilisateur (/etc/passwd est lisible par tout le monde...). Si
 - ▶ * : compte verrouillé
 - ▶ x : il faut accéder aux vrais mots de passe cryptés dans le fichier /etc/shadow lisible seulement par root
 - ▶ rien : pas de mot de passe 
Peut servir pour `sync`, affichage du menu de la cantine, des gens connectés, de la date,... sans se connecter explicitement
- uid* : numéro de l'utilisateur



gid : numéro de groupe primaire de l'utilisateur

gecos-field : nom réel de l'utilisateur (apparaît dans le champ From: des mails par exemple) (nom du temps où ces informations étaient utilisées pour identifier les commandes envoyées au General Electric Computer Operating System chez Bell Labs...)

- La commande `passwd` permet de changer son mot de passe ou le mot de passe de quelqu'un d'autre (si on est `root`...)
- `su` permet de prendre les droits d'un autre utilisateur, `su -` lance en plus un nouveau shell de login



Fichier /etc/shadow

- Utilisé en plus de `/etc/passwd` pour éviter que tout le monde puisse accéder aux mots de passes cryptés et fasse tourner un craqueur de mot de passe
- Utilisé localement ou via NIS+ (NIS n'a pas de protection de lecture sauf `glibc2` par exemple)
- Syntaxe (`man -s 4 shadow`) :
username:password:lastchg:min:max:warn:inactive:expire
username : nom de login
password : mot de passe crypté de l'utilisateur
 - ▶ **LK** : compte verrouillé
 - ▶ rien : pas de mot de passe *lastchg* : jour de dernière modification du mot de passe
min : nombre de jours minimum nécessaires entre des



changements de mots de passe

max : nombre de jours de validité du mot de passe. Après, un nouveau est demandé

inactive : nombre de jours d'inactivité tolérable avant blocage

expire : jour d'expiration du compte



Le fichier /etc/group

- Déclare les groupes
- Donne un nom aux groupes utilisés dans `/etc/passwd`
- Rajoute les utilisateurs dans d'autres groupes, les groupes secondaires. Utile pour déclarer des entités transversales (appartenance à des projets,...). Limite par défaut de 16 groupes secondaires par utilisateur
- `newgrp` permet de se connecter sous un nouveau groupe
- `groups` affiche la liste des groupes auxquels on appartient
- Syntaxe (`man -s 4 group`) :
groupname : *password* : *gid* : *user-list*
groupname : nom du groupe
password : mot de passe permettant à des utilisateurs de se



connecter avec `newgrp` sous un autre groupe dont il ne fait pas partie

gid : numéro de groupe avec les mêmes contraintes que pour le numéro d'utilisateur

user-list : liste des utilisateurs appartenant au groupe

- Problème du `NGROUPS_MAX` à configurer grand (16 par défaut)...
`ngroups_max` dans `/etc/system`
-  La modification des groupes d'un utilisateur nécessite une reconnexion ou un `newgrp` pour être prise en compte



Répertoires utilisateurs

Où les mettre ?

- Sun : `/home/utilisateur` (et donc sur le serveur dans `/export/home/utilisateur`). Simple mais peu hiérarchique
- Au CRI et LIT :
 - ▶ `/users/centre/utilisateur` est un lien symbolique ou un auto-montage vers `/home/...` pour cacher l'éclatement possible d'un centre sur plusieurs disques
 - ▶ `/home/machine/disque/centre/utilisateur` contient réellement le répertoire utilisateur
 - ▶ Intérêt à choisir de mettre dans `/etc/passwd` le dernier type ou le cacher avec l'auto-montage pour résister à une disparition catastrophique de `/users/...`



- 2 religions principales existent : les shells à syntaxe à la sh (sh, jsh, ksh, bash, zsh) et ceux à syntaxe à la « C » (csh, tcsh)
- Fichiers de configuration trouvés dans le répertoire principal de l'utilisateur
- 2 états de connexion possibles (shell de connexion principale (login) ou simplement interactif (autres fenêtres ou autre shell) + 1 état non interactif (shell utilisé pour exécuter un script)
- sh
 - ▶ /etc/profile (*a priori* non partagé...) puis .profile si shell de login
- bash : lecture dans l'ordre de
 - ▶ /etc/profile si shell de login



- ▶ premier fichier trouvé dans l'ordre .bash_profile, .bash_login et .profile si shell de login
- ▶ .bashrc si shell interactif mais pas de login
- ▶ le fichier spécifié par \$BASH_ENV si elle existe lors du lancement d'un shell non interactif
- ▶ .bash_logout lors de la déconnexion si shell de login
- csh et tcsh : lecture dans l'ordre de
 - ▶ /etc/csh.cshrc
 - ▶ /etc/csh.login pour un shell de login
 - ▶ .cshrc ou .tcshrc à la place (s'il existe et tcsh)
 - ▶ .history si shell de login (contient la liste des commandes précédemment tapées)
 - ▶ .login pour un shell de login



- ▶ `.cshdirs` si shell de login (contient la pile de répertoires courants)
- ▶ `.logout` lors de la déconnexion si shell de login
- ↔ Nécessité de fournir des fichiers tout configurés aux utilisateurs mais configurables et en plus avec un mécanisme d'évolution centralisée... Inclure par exemple un `source /usr/local/share/env/tcshrc`
`/etc/skel` contient des fichiers standard de configuration



Variables d'environnement

Quelques unes des variables d'environnement changeant le comportement

`DISPLAY` : nom de l'écran X11 à utiliser pour l'affichage

`HOME` : nom du répertoire personnel

`LANG` : langage et codage des messages du système. Variables `LC_...` pour changer seulement des parties (cf. `/etc/default/init` pour exemple)

`LD_LIBRARY_PATH` : liste des répertoires où trouver les bibliothèques dynamiques

`LOGNAME` : nom d'utilisateur (*gecos*)

`LPDEST` : imprimante par défaut

`MAIL` : nom du fichier boîte aux lettres



MANPATH : liste des répertoires où trouver les manuels

MANSECTS : liste des sections de manuels possibles

PATH : liste des répertoires où trouver les commandes

SHELL : shell à utiliser si besoin est par certains programmes

TERM : modèle de terminal à utiliser

TZ : fuseau horaire



Role-Based Access Control (RBAC)

- `root` trop puissant ?
- Utilisateurs standard pas assez puissants pour gérer leurs propres problèmes
- Tous les utilisateurs ne peuvent pas être `root` à chaque fois qu'ils ont une tâche un peu spéciale à faire...
- Éclatement de la puissance de `root` en rôles plus petits
 - ▶ Autorisation : permet d'utiliser une fonction protégée
 - ▶ Profil d'exécution : paquet d'autorisations avec des attributs spéciaux, typiquement associé à un utilisateur ou un groupe
 - ▶ Rôle : type de compte utilisateur destiné à faire des tâches administratives
- À partir de Solaris 8 (pas encore très standardisé dans Unix)



- Exemple d'un rôle *imprim* pour résoudre les problèmes d'impression
 - ▶ `su imprim` puis mot de passe le cas échéant
 - ▶ `lpadmin -p lw-d109 options`
 - ▶ `lprm -p lw-d109 ...`
 - ▶ Les commandes ainsi que leur droits sont spécifiées par le rôle *imprim*
- On ne peut pas se connecter directement comme rôle, il faut être déjà connecté comme utilisateur
- Actions notées comme venant de l'utilisateur et non du rôle
- Améliore la traçabilité



- `auths [user]...` affiche les autorisations d'utilisateur(s)
- `profiles [-l] [user]...` affiche les profils d'utilisateur(s)
- `roles [user]...` affiche les rôles accessibles à un/des utilisateur(s)



- `roleadd [-c comment] [-d dir] [-e expire] [-f inactive] [-g group] [-G group[,group...]] [-m [-k skel_dir]] [-u uid [-o]] [-s shell] [-A authorization[,authorization...]]` *role* crée un nouvel utilisateur correspondant au rôle
- `roledel [-r] role` efface le compte du rôle
- `rolemod [-u uid [-o]] [-g group] [-G group[,group...]] [-d dir [-m]] [-s shell] [-c comment] [-l new_name] [-f inactive] [-e expire] [-A authorization[,authorization]] [-P profile[,profile]]` *role* modifie les paramètres d'un rôle



- `/etc/user_attr` complète `passwd` et `shadow`
- `root:::type=normal;auths=solaris.*,solaris.grant;profiles=All`
`imprim:::type=role;profiles=Printer Management,All`
`keryell:::type=normal;auths=solaris.system.date;roles=imprim;\`
`profiles=All`



- /etc/security/auth_attr
- solaris.system.date:::Set Date & Time::help=SysDate.html
solaris.admin.printer:::Printer Information::
solaris.admin.printer.read:::View Printer Information::\
 help=AuthPrinterRead.html
solaris.admin.printer.modify:::Update Printer Information::\
 help=AuthPrinterModify.html
solaris.admin.printer.delete:::Delete Printer Information::\
 help=AuthPrinterDelete.html



- /etc/security/prof_attr
- All:::Execute any command as the user or role:help=RtAll.html
Basic Solaris User:::Automatically assigned rights:\
 auths=solaris.profmgr.read,solaris.jobs.users,\
 solaris.admin.usermgr.read,solaris.admin.logsvc.read,\
 solaris.admin.fsmgr.read,solaris.admin.serialmgr.read,\
 solaris.admin.diskmgr.read,solaris.admin.procmgr.user,\
 solaris.compsys.read,solaris.admin.printer.read,\
 solaris.admin.prodreg.read,solaris.admin.dcmgr.read;\
 profiles=All;help=RtDefault.html
Printer Management:::Manage printers, daemons, spooling:\
 help=RtPrntAdmin.html;auths=solaris.admin.printer.read,\
 solaris.admin.printer.modify,solaris.admin.printer.delete



- /etc/security/exec_attr
- Maintenance and Repair:suser:cmd:::/usr/bin/date:euid=0
- Printer Management:suser:cmd:::/usr/sbin/accept:euid=lp
- Printer Management:suser:cmd:::/usr/ucb/lpq:euid=0
- Printer Management:suser:cmd:::/etc/init.d/lp:euid=0
- Printer Management:suser:cmd:::/usr/bin/lpstat:euid=0
- Printer Management:suser:cmd:::/usr/lib/lp/lpsched:uid=0
- Printer Management:suser:cmd:::/usr/sbin/lpfilter:euid=lp
- Printer Management:suser:cmd:::/usr/bin/lpset:egid=14
- Printer Management:suser:cmd:::/usr/sbin/lpadmin:egid=14
- Printer Management:suser:cmd:::/usr/sbin/lpsystem:uid=0
- Printer Management:suser:cmd:::/usr/sbin/lpmove:euid=lp
- Printer Management:suser:cmd:::/usr/sbin/lpshut:euid=lp
- Printer Management:suser:cmd:::/usr/bin/cancel:euid=0



Printer Management:suser:cmd:::/usr/bin/disable:euid=lp

Printer Management:suser:cmd:::/usr/sbin/reject:euid=lp

Printer Management:suser:cmd:::/usr/sbin/lpforms:euid=lp

Printer Management:suser:cmd:::/usr/ucb/lprm:euid=0

Printer Management:suser:cmd:::/usr/bin/enable:euid=lp

Printer Management:suser:cmd:::/usr/sbin/lpusers:euid=lp



- Appartiennent à un utilisateur propriétaire (*u*, *user*) et à un groupe (*g*, *group*)
- En Unix de base, droits spécifiés pour le propriétaire (*u*), le groupe (*g*) et les autres (*o*, *other*) représentés dans classiquement dans l'ordre *ugo*
- Droits en lecture (*r*) et écriture (*w*) pour utilisateur, groupe et autres
- Droit en exécution (*x*) idem mais
 - ▶ Sur 1 fichier : programme exécutable. Possible d'exécuter un programme sans être capable d'en voir le contenu
 - ▶ Sur 1 répertoire : droit de traversée. *d--x--x--x* : on peut accéder à des fichiers dans le répertoire mais on ne peut pas en voir le contenu



- *suid* bit : change l'identificateur d'utilisateur effectif du processus à celui du propriétaire lors de l'exécution
 - *sgid* bit
 - ▶ Sur exécutable : change l'identificateur de groupe effectif du processus à celui du fichier lors de l'exécution
 - ▶ Sur répertoire : les fichiers créés dans le répertoire héritent du groupe du répertoire au lieu du groupe effectif du processus
- ```
drwxrwsr-x 13 pips pips_grp 1024 Sep 18 09:17 /projects/Pips
```
- *Sticky bit* *t*
    - ▶ Sur fichier exécutable : reste collé en mémoire physique si possible. Anachronisme avec les Unix modernes
    - ▶ Sur répertoire : interdit à un utilisateur d'effacer un répertoire qui n'est pas à lui même s'il a le droit en écriture sur le



répertoire

```
drwxrwxrwt 7 sys sys 318 Feb 2 11:04 /tmp
```

- Généralisation des concepts à des listes d'utilisateurs : *Access Control List (ACL)*
- `chmod`, `chown`, `chgrp`, `touch` pour changer les différentes informations d'un fichier
- `umask` contrôle les droits par défaut des fichiers créés en spécifiant les bits à mettre à 0 par rapport à ce qui est demandé. En octal (`rxw`) dans l'ordre `ugo` :  
`umask 022` : met à 0 le bit d'écriture pour le groupe et les autres  
↔ fichiers en lecture-écriture pour soi et en lecture seule pour le reste du monde
- Bien contrôler les droits des fichiers pour éviter des trous de sécurité. Éviter `umask 0` avec `root`...



- Besoins temporaires de changer d'identité
  - ▶ Pour changer son mot de passe il faut que la commande `passwd` tourne sous `root` pour modifier `/etc/shadow`
  - ▶ Envoyer du mail : `sendmail` doit écrire le mail dans la file d'attente sans qu'elle soit écrivable par tout le monde
- 2 types d'identificateurs d'utilisateurs et de groupes
  - ▶ Réels : représente l'identité sous laquelle on s'est connecté. Contrôle la permission d'envoi de signaux
  - ▶ Effectifs : représente l'identité sous laquelle on effectue des actions à l'instant : création et accès aux fichiersPar défaut identificateurs effectifs = identificateurs réels
- Seule possibilité de changement pour un utilisateur normal : remettre les identificateurs effectifs à la valeur de leurs



identificateurs réels

- root peut faire toute les manipulations d'identificateur. Exemple :  
login
- Pour acquérir d'autres droits sous contrôle : programmes exécutables avec les bits *suid* et *sgid*. Le système positionne les identificateurs effectifs d'utilisateur et/ou de groupe en conséquence
  - ▶ `-r-sr-sr-x 3 root sys 85760 Oct 6 09:57 /bin/passwd`
  - ▶ `-r-xr-sr-x 1 bin tty 11592 Oct 6 10:10 write`
  - ▶ `--s-x-x 1 root bin 50652 Feb 1 23:05 /bin/su`  
permet de changer d'identificateurs effectifs en tapant un mot de passe
  - ▶ `-rwsr-xr-x 1 root sys 7628 Oct 6 09:48 /bin/newgrp`  
permet de se connecter sous un autre groupe (équivalent de



su pour les groupes). Laisse des traces dans  
`/var/adm/sulog`

- ▶  Vérifier régulièrement que des programmes *suid/sgid* n'apparaissent pas...
- ▶  Ne pas faire de *suid shell script* : exécution en général non atomique  $\rightsquigarrow$  attaque avec liens symboliques
- ▶  Interdire les *suid/sgid* depuis des médias temporaires (disquettes, CD-ROM) et via `/net` de l'automonteur : un programme *suid* distant est vu localement aussi comme un *suid* local...
- ▶  Cheval de Troie si « . » au début du PATH de root. Faire par exemple une commande `ls` chez soi qui crée un *shell suid* root avant d'appeler le vrai `ls`
-  Si bugs dans des programmes *suid* ou *sgid* (root,...)



- SVR4 : notion en plus de numéros d'utilisateurs et de groupes effectifs *sauvegardés* passés à travers un `exec()`. Possibilité de refaire un changement d'identificateurs vers ces identificateurs sauvegardés
-  Si `uid` réel ou effectif de l'expéditeur vaut `uid` réel ou sauvegardé du récepteur : envoi de *signal* (`kill()`) possible  $\rightsquigarrow$  appel de gestionnaire de signal hors contexte potentiel...



## Devices — Fichiers conducteurs de périphériques

- Contrôle du matériel par des fichiers de type
  - ▶ Mode caractère : la majorité. Mode brut, disques par bloc de 512 octets

```
crw-r----- 1 root sys 13, 1 Jan 7 08:21 mm@0:kmem
```
  - ▶ Mode bloc : cachage des données par bloc pour permettre une E/S plus souple

```
brw-r----- 1 root sys 32, 50 Dec 11 1997 sd@6,0:c
```
- Fonctionnalité dans le système associée au numéro majeur et mineur et pas au nom
-  Ne pas autoriser les devices à travers média transportables ou via le réseau hostile (`/net` automonteur) : écran, clavier, `/dev/kmem` ou disques locaux en lecture écriture pour tout le monde localement...



-  Vérifier qu'il n'y a pas des conducteurs louches qui traînent...



- Un pirate peut avoir mis des fichiers dans des répertoires discrets : « ... », « »,...
- Les noms de fichier peuvent contenir des caractères de contrôle perturbant (cachant) l'affichage des noms
- Recherche des programmes louches *suid/sgid* en affichant les en octal les caractères non-ASCII

```
find / \(-perm -004000 -o -perm -002000 \) -type f -print | cat -ve
```

- Un pirate pourrait aussi modifier les appels système pour cacher ses fichiers...



- Certains systèmes (4.4BSD) ont des attributs d'immuabilité (fichiers de configuration,...) et d'ajout seulement (fichiers de log,...)
- 4.4BSD a 4 niveaux de sécurité fonctionnement. `root` peut augmenter le niveau mais seul `init` peut le baisser
  - ▶ Niveau le plus bas : Unix standard
  - ▶ Niveau le plus haut : drapeaux d'immuabilité et d'ajout seulement non modifiables, `/dev/mem` et `/dev/kmem` non écrivables même par `root`, disques non écrivables en mode brut même par `root`,...
- Média en lecture seulement : commutateur sur un disque dur, CD-ROM
- Systèmes de fichiers en lecture seulement. `root` peut modifier



ce statut. Si une partition est en lecture seule sauf quelques fichiers les remplacer par des liens symboliques vers une partition écrivable

- Exportation de disques en lecture seulement si possible
- N'empêche pas un accès physique à la machine ou aux disques...



- Besoin de définir les accès aux fichiers plus finement que (utilisateur,groupe,autre)
- Groupes informels sans avoir besoin d'être `root` pour les créer
- Pouvoir rajouter des accès supplémentaires pour d'autres utilisateurs ou groupes
- Pas encore très standardisé dans Unix. Lié à UFS (Solaris) et NFSv3

[http://docs.sun.com:80/ab2/coll.47.11/SYSADV2/@Ab2PageView/idmatch\(SECFILES-37\)?DwebQ](http://docs.sun.com:80/ab2/coll.47.11/SYSADV2/@Ab2PageView/idmatch(SECFILES-37)?DwebQ)



## Représentation des types de droits

Utilisée en affichage ou pour établissement

- Droits « classiques » aussi représentable en ACL :
  - ▶ `u[ser]::perms`
  - ▶ `g[roup]::perms`
  - ▶ `o[ther]::perms`
- Droits supplémentaires sur les fichiers :
  - ▶ `u[ser]:uid:perms` : rajoute l'accès à un utilisateur
  - ▶ `g[roup]:gid:perms` : rajoute l'accès à un groupe
  - ▶ `m[ask]:perms` : « et » logique à appliquer aux 2 types d'accès précédent et aux droit du groupe propriétaire : `m:r-` implique que les utilisateurs ou les groupes supplémentaires ne pourront pas avoir mieux que la lecture du fichier. Facilite la restriction



- Droits par défauts sur les répertoires pour les fichiers qui y seront créés. Étend la sémantique BSD (+s) :
  - ▶ `d[efault]:u[ser]::perms`
  - ▶ `d[efault]:g[roup]::perms`
  - ▶ `d[efault]:o[ther]:perms`
  - ▶ `d[efault]:u[ser]:uid:perms`
  - ▶ `d[efault]:g[roup]:gid:perms`
  - ▶ `d[efault]:m[ask]:perms`



## Manipulation des ACL

- Gestionnaire graphique des fichiers (⚠ bugs sous Solaris... ☹)
- Modification
  - ▶ `setfacl -r -m user:delafune:rw-,user:jimenez:r-unix.ps` modifie les droits. `-r` pour recalculer un masque minimal autorisant les accès demandés (évite de l'expliciter)
  - ▶ `setfacl -s droits fichiers` met des droits
  - ▶ `setfacl -d droits fichiers` efface des droits
- Lecture
  - ▶ `ls -l`

```

-rw-r--r-- 1 keryell ensrec 220955 Sep 6 12:35 trans.tex
-rw-r-----+ 1 keryell ensrec 511269 Sep 6 13:51 unix.ps

```
  - ▶ `getfacl unix.ps`

```

file: unix.ps

```



```
owner: keryell
group: ensrec
user::rw-
user:delafune:rw- #effective:rw-
user:jimenez:r-- #effective:r--
group::r-- #effective:r--
mask:rw-
other:---
```

- ▶ `getfacl -d` permet d'avoir les droits par défaut d'un répertoire
- Copie
  - ▶ `getfacl fichier1 | setfacl -f - fichier2`
  - ▶ Les outils d'archivage gèrent les ACL



Processus de *boot* : démarre le noyau système (ou tout autre programme autonome : programme de test,...)

- *Bootstrap loader* : petit programme suffisant pour charger un gros programme (lorsque les ordinateurs se démarraient en programmant explicitement les instructions de démarrage...)
- `man boot`
- Différences entre un Sun et un PC...
- Sun :
  - ▶ Programme (*firmware*) de démarrage écrit en PROM
  - ▶ Exécution d'un programme de test à l'allumage
  - ▶ Chargement du système automatique si indiqué dans la mémoire non volatile du système (`man eeprom`)
  - ▶ Boot primaire depuis



- Disque : bloc de boot primaire (`ufsboot`) dans les blocs 1 à 15 du disque
  - Réseau : requête RARP pour récupérer le numéro IP à partir du numéro Ethernet de la machine puis diffusion d'une requête TFTP pour récupérer `inetboot` depuis le réseau qui lui-même fait une requête RARP puis diffuse une requête en protocole `bootparams` pour savoir où trouver son noyau qu'il monte alors par NFS
- PC :
    - ▶ Boot primaire implémenté dans la ROM du BIOS et dans les ROMs des cartes d'extension
    - ▶ Contrôle possible des périphériques et E/S via interruptions
    - ▶ Charge le premier secteur physique du disque ou de la disquette (*Master Boot Record*) et l'exécute en mode réel



- ▶ Le boot secondaire (`boot.bin`) est chargé par le boot primaire et s'exécute en mode protégé et paginé 32 bits en appelant du BIOS en mode réel et est capable de charger un programme depuis un disque UFS, un CD-ROM ou le réseau
  - ▶ Le boot secondaire peut lancer le *Configuration Assistant* pour changer la configuration matérielle si l'utilisateur tape ESC rapidement
  - ▶ Ensuite, si la variable `auto-boot` de `eeprom` l'indique, le boot secondaire commence et interprète `/etc/bootrc` qui contrôle le processus de démarrage et charge le noyau par défaut. `b` permet alors de spécifier un autre fichier de démarrage et `i` fait entrer en mode interactif
- Enfin démarrage du noyau par défaut
  - Chargement des modules nécessaires au fonctionnement du



noyau

- Montage des disques nécessaires décrits dans `/etc/vfstab`
- Démarrage de `/sbin/init` qui amène le système dans l'état par défaut
- Comme système d'enregistrement des messages non fonctionnel dès le début, regarder messages initiaux avec `dmesg`



## Niveaux de fonctionnement

`man init`

- Système V divise l'état du système en plusieurs niveau pour clarifier la situation
- 0 : retourne sous moniteur ou BIOS
- 1 : met le système en mode administrateur. Les disques locaux sont montés
- 2 : mode multi-utilisateur
- 3 : mode multi-utilisateur et exportation de services vers le monde extérieurs (disques,...)
- 4 : libre pour tout usage
- 5 : arrête le système et la machine (si possible par le matériel)
- 6 : arrête le système et reboot dans l'état par défaut



- a, b et c : pseudo-état pour lancer des commandes correspondant à ces niveaux sans changer l'état
- S ou s : mode *single-user* pour la maintenance (ne monte pas les disques locaux autres que ceux nécessaire au système de base)

Le comportement des niveaux est décrit dans le fichier `/etc/inittab` qui est relu en tapant `init q`

```
ap::sysinit:/sbin/autopush -f /etc/iu.ap
ap::sysinit:/sbin/soconfig -f /etc/sock2path
fs::sysinit:/sbin/rcS sysinit >/dev/msglog 2<>/dev/msglog </dev/console
is:3:initdefault:
p3:s1234:powerfail:/usr/sbin/shutdown -y -i5 -g0 >/dev/msglog 2<>/dev/msglog
sS:s:wait:/sbin/rcS >/dev/msglog 2<>/dev/msglog </dev/console
s0:0:wait:/sbin/rc0 >/dev/msglog 2<>/dev/msglog </dev/console
s1:1:respawn:/sbin/rc1 >/dev/msglog 2<>/dev/msglog </dev/console
s2:23:wait:/sbin/rc2 >/dev/msglog 2<>/dev/msglog </dev/console
s3:3:wait:/sbin/rc3 >/dev/msglog 2<>/dev/msglog </dev/console
s5:5:wait:/sbin/rc5 >/dev/msglog 2<>/dev/msglog </dev/console
s6:6:wait:/sbin/rc6 >/dev/msglog 2<>/dev/msglog </dev/console
```



```
fw:0:wait:/sbin/uadmin 2 0 >/dev/msglog 2<>/dev/msglog </dev/console
of:5:wait:/sbin/uadmin 2 6 >/dev/msglog 2<>/dev/msglog </dev/console
rb:6:wait:/sbin/uadmin 2 1 >/dev/msglog 2<>/dev/msglog </dev/console
sc:234:respawn:/usr/lib/saf/sac -t 300
co:234:respawn:/usr/lib/saf/ttymon -g -h -p "'uname -n' console login: "\
-T sun -d /dev/console -l console -m ldterm,ttcompat
```

Le premier champ est juste mnémotechnique. `initdefault` indique l'état normal

`/etc/default/init` positionne les variables d'environnement de tous les processus

```
TZ=MET
LC_COLLATE=fr.ISO8859-15
LC_CTYPE=fr.ISO8859-15
LC_MESSAGES=fr
LC_MONETARY=fr.ISO8859-15
LC_NUMERIC=fr.ISO8859-15
```



LC\_TIME=fr.ISO8859-15

qui précise le fuseau horaire et le fait qu'on veut du français et un encodage avec Euro



## Démarrage en mode mono-utilisateur

- Permet de ne pas être dérangé par les utilisateurs lors de tâches d'administration lourdes (mise à jour du système,...)
- Niveau de démarrage par défaut en cas de gros problèmes (disques immontables car systèmes de fichiers corrompus) et attente d'une correction par l'administrateur
- Peut passer en mode mono-utilisateur depuis un état mono-utilisateur
- Boot avec `boot -s` (Sun) ou `b -s` (PC)
- Mode qui pose plus de questions avec `boot -as` et `b -as`
- Si on sort du shell mono-utilisateur (`^D`) on lance le mode multi-utilisateur



- D'abord savoir où on en est... `who -a` (pas la version GNU)
- Le plus violent (ne prévient pas les utilisateurs...) : `init niveau`
- Des équivalents rapides
  - ▶ `reboot` (BSD) pour `init 6`
  - ▶ `halt` (BSD) pour `init 5`
  - ▶ `poweroff` pour `init 0`
- Plus conciliant avec les utilisateurs : `shutdown`. Exemple :  
`shutdown -y -g secondes -i niveau message`  
 envoie un message régulier à tous les utilisateurs annonçant la fin prochaine et la raison. Messages envoyés aussi aux utilisateurs des machines utilisant des disques de la machine locale



## Lancer des choses au démarrage

- Chaque niveau  $n$  lance l'exécution d'un script `/sbin/rcn` (*Run Control script*)
- Grossièrement, chaque script exécute les programmes du répertoire `/etc/rcn.d` commençant par la lettre  $K$  par ordre alphabétique et avec `stop` comme argument puis les scripts commençant par la lettre  $S$  par ordre alphabétique et avec `start` comme argument

Exemple de `/etc/rc2.d` :

|               |                |              |                   |
|---------------|----------------|--------------|-------------------|
| K06mipagent   | S47asppp       | S74syslog    | S90wbem           |
| K07dmi        | S47pppd        | S74xntpd     | S91afbinit        |
| K07snmpdx     | S69inet        | S75cron      | S91ifbinit        |
| K16apache     | S70sckm        | S75flashprom | S92volmgt         |
| K21dhcp       | S70uucp        | S75savecore  | S93cacheos.finish |
| K28nfs.server | S71ldap.client | S76nsd       | S94ncalogd        |
| README        | S71rpc         | S77sf880dr   | S95ncad           |
| S01MOUNTFSYS  | S71sysid.sys   | S80PRESERVE  | S98efcode         |
| S05RMTMPFILES | S72autoinstall | S80lp        | S99audit          |



|              |                   |             |            |
|--------------|-------------------|-------------|------------|
| S20syssetup  | S72inetsvc        | S80spc      | S99dtlogin |
| S21perf      | S72slpd           | S85power    | S99openssh |
| S22acct      | S73cachefs.daemon | S88sendmail |            |
| S30sysid.net | S73nfs.client     | S88utmpd    |            |
| S40llc2      | S74autofs         | S89bdconfig |            |

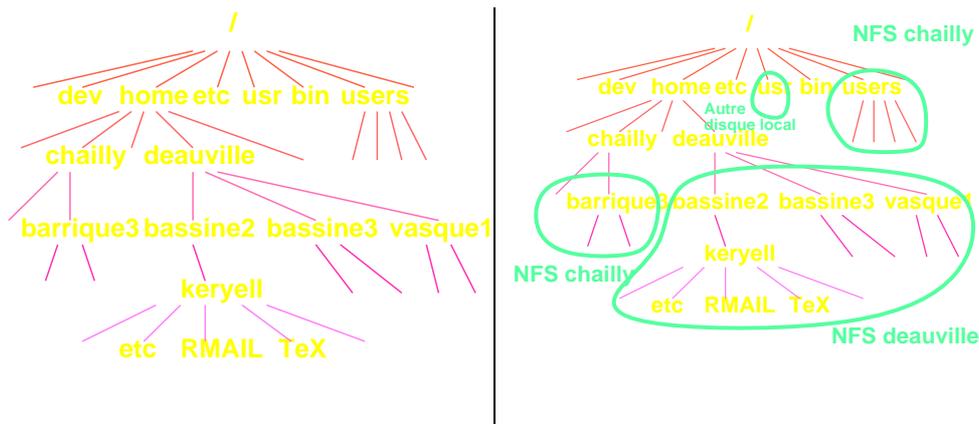
- Pour simplifier, comme un script peut être utilisé par plusieurs niveaux, ils sont tous mis dans `/etc/init.d` et on utilise en fait des liens vers eux depuis les `/etc/rcn.d`. En plus les scripts `K` et `S` pointent vers les mêmes fichiers
- Si un script se termine par `.sh` il est « sourcé » par le `sh` courant du `/sbin/rcn` au lieu d'être exécuté dans un autre `sh` afin de propager des effets de bords (variables de `sh`,...)
- Les comportements varient en fonction des niveaux et des niveaux précédents
- ↪ Simple de rajouter quelque chose de nouveau en gérant les



dépendances. Exemple de `pppd` : faire un `/etc/init.d/ppp`

```
#!/bin/sh
state=$1
case $state in
'start')
 /etc/ppp/ppp-on &
 ;;
'stop')
 /etc/ppp/ppp-off
 ;;
esac
```





- Organisation hiérarchique simplifiant l'administration
- « Montage » ≡ accrocher une hiérarchie à un répertoire de la hiérarchie déjà existante



- Morceaux de hiérarchie privés (configuration et fichiers de la machine elle-même)
- Morceaux de hiérarchie partagés (partage de fichiers)
- Pour avoir un système fonctionnel il faut au moins
  - / : La racine de la hiérarchie Unix. Partition montée par le processus de démarrage
  - /usr : Hiérarchie partageable montée par un script lors du démarrage
- La majorité des choses sont configurables à partir de fichiers texte (pas de Registry...)



`man filesystem`

`/boot` : version PC : contient la configuration de boot et le système d'aide à la configuration matérielle. Émulation de l'EEPROM des Sun

`/dev` : fichiers spéciaux permettant d'accéder au matériel. Les fichiers *devices* sont typiquement construits pour refléter la configuration logicielle et matérielle de la machine

`/dev/dsk` : disques accédés en mode bloc (pas de système de fichier)

`/dev/pts` : pseudo-terminaux

`/dev/rdisk` : disques accédés en mode brut caractère (pas de système de fichier) mais de manière brute (par blocs de 512 octets forcément)



`/dev/rmt` : bandes en mode brut

`/dev/sad` : gestion des STREAMS (programmes qu'on peut insérer dans des flots de données)

`/dev/term` : pour accéder aux terminaux et modems

`/dev/cua` : pour accéder aux terminaux et modems en outrepassant la détection de porteuse

`/devices` : idem `/dev` mais non plus classé par fonction mais par contrôleur (pseudo/, sbus@1f,0/). En général les fichiers de `/dev` sont des liens vers des fichiers de `/devices`

`/etc` : fichiers et bases de données de configuration spécifiques à la machine. Définit l'identité de la machine

`/etc/acct` : configuration du système de comptabilité système

`/etc/cron.d` : configuration du système de lancement de processus



à instants déterminés `cron`

`/etc/default` : configuration par défaut pour divers programmes

`/etc/dfs` : information de configuration sur les systèmes de fichiers exportés vers (utilisables par) d'autres machines

`/etc/fs` : binaires permettant de monter différents types de systèmes de fichier (CD-ROM, disque, NFS) lors du démarrage avant le montage de `/usr`

`/etc/inet` : fichiers de configuration d'Internet et de ses services

`/etc/init.d` : scripts utilisés pour passer d'un niveau d'exécution du système à un autre

`/etc/lib` : bibliothèques partagées utilisées lors du démarrage

`/etc/lp` : fichiers de configuration du système d'impression



`/etc/mail` : configuration du courrier électronique

`/etc/net` : configuration de certains services réseaux indépendamment du niveau transport

`/etc/opt` : informations de configuration sur les packages optionnels

`/etc/rc0.d` : scripts pour entrer ou sortir du niveau d'exécution 0 du système (moniteur/BIOS de la machine)

`/etc/rc1.d` : scripts pour entrer ou sortir du niveau 1 (administration système)

`/etc/rc2.d` : scripts pour entrer ou sortir du niveau 2 (multi-utilisateur)

`/etc/rc3.d` : scripts pour entrer ou sortir du niveau 3 (multi-utilisateur et exporte les ressources)



- /etc/rcS.d : scripts pour entrer ou sortir du niveau S (*single-user*)
- /etc/saf : configuration de l'utilitaire d'accès à des services (login,...)
- /etc/skel : fichiers de configuration par défaut pour le rajout d'utilisateur via useradd
- /etc/tm : fichiers Trademark affichés lors du démarrage du système
- /etc/uucp : information de configuration d'UUCP
- /export : racine par défaut des disques qui seront exportés
- /floppy : pour accéder au lecteur de disquette
- /home : racine par défaut des répertoires des utilisateurs
- /mnt : répertoire vide utilisable pour des montages temporaires de systèmes de fichiers



- /opt : répertoire contenant des applications additionnelles sous forme de package
- /proc : contient le système de fichier des processus
- /sbin : exécutables nécessaire lors de la phase de démarrage ou d'une phase de récupération système après catastrophe
- /tmp : répertoire des fichiers temporaires :  effacé pendant la phase de démarrage
- /var : répertoire contenant des fichiers propres à la machine mais qui varient (grandissent...) au cours du temps
- /var/adm : contient des fichiers de log et de comptabilité
- /var/log : contient des fichiers de log et de comptabilité
- /var/cron : fichiers de log de cron



- /var/mail : boîtes aux lettres des utilisateurs
- /var/nis : base de donnée NIS+
- /var/opt : fichiers variants associés à des logiciels optionnels
- /var/preserve : contient les fichiers de sauvegarde pour vi et ex
- /var/sadm : base de données utilisée par le système de gestion des paquets de logiciel
- /var/saf : messages et comptabilité de saf
- /var/spool : répertoire de *spool* (*Simultaneous Peripheral Operation On Line*)
- /var/spool/cron : contient les bases de données cron et at par utilisateur
- /var/spool/locks : contient des fichiers de verrouillage



- /var/spool/lp : zone de stockage des fichiers à imprimer
- /var/spool/mqueue : zone de stockage du courrier électronique en cours de traitement
- /var/spool/pkg : zone de stockage de paquets de logiciels
- /var/spool/uucp : opérations uucp en cours de traitement
- /var/spool/uucppublic : fichiers déposés par uucp
- /var/tmp : fichiers temporaires. Répertoire non effacé lors du démarrage du système
- /var/uucp : fichiers de messages et statut d'uucp
- /var/yp : base de donnée NIS
- /vol : répertoire de périphériques gérés par le gestionnaire de volume



- /kernel : hiérarchie contenant la partie du noyau qui est indépendante des spécificités de la machine
- /kernel/drv : contrôleurs du matériel 32 bits
- /kernel/drv/sparcv9 : contrôleurs du matériel 64 bits pour SPARCv9
- /kernel/genunix : le noyau lui-même, indépendant de l'architecture de la machine
- /platform : répertoire contenant les objets spécifiques à la plate-forme. Structure calquée sur celle de /
- /platform/*platform-name*/kernel : contient les parties du noyau spécifiques au matériel
- /platform/*platform-name*/kernel/unix : noyau 32 bits
- /platform/*platform-name*/kernel/sparcv9/unix : noyau 64 bits



pour SPARCv9

- /platform/*platform-name*/lib : bibliothèques spécifiques
- /platform/*platform-name*/sbin : exécutables spécifiques



- Garder / petit
- ↪ gros disques montés dans /export, /usr, /opt, /var,...
- /usr contient des fichiers partageables indépendants de l'architecture (/usr/share) ou pas
- Possible de faire des serveurs de /usr (en lecture seulement)

/usr/4lib : bibliothèques de compatibilité Solaris 1 (SunOS4)

/usr/bin : contient les utilitaires systèmes standard

/usr/ccs : système de compilation C

/usr/demo : programmes de démonstration avec leur données

/usr/dt : racine de CDE Motif.

/usr/dt/bin : utilitaires CDE de base



/usr/dt/include : entêtes CDE

/usr/dt/lib : bibliothèques CDE

/usr/dt/man : manuels CDE

/usr/games : des jeux ! Mais vide...

/usr/include : les entêtes pour la compilation C

/usr/java : compilateur et machine virtuelle Java

/usr/kernel : morceau du noyau indépendant de la plate-forme  
non indispensable au démarrage

/usr/lib : bibliothèques ainsi que certains exécutable tels que des  
démons

/usr/lib/64 : bibliothèques 64 bits

/usr/lib/acct : scripts et binaires utilisés pour la comptabilité



- /usr/lib/class : classes d'ordonnement
- /usr/lib/font : fontes pour troff
- /usr/lib/fs : modules et commandes spécifiques à des systèmes de fichier
- /usr/lib/iconv : tables de codage de jeux de caractères pour iconv
- /usr/lib/libp : bibliothèques système utilisée lors du *profiling*
- /usr/lib/locale : bases de données de localisation en fonction de la langue
- /usr/lib/lp : bases de données et exécutables utilisés par le processus d'impression
- /usr/lib/mail : fichiers utilisés pour la génération de la configuration de sendmail V8



- /usr/lib/netsvc : programmes et démons utilisés par les services réseau Internet
- /usr/lib/nfs : démons NFS
- /usr/lib/sa : commandes utilisées par le système d'analyse d'activité sar
- /usr/lib/saf : programmes et démons pour saf
- /usr/lib/sparcv9 : bibliothèques 64 bits SPARCv9
- /usr/lib/spell : bases de données du correcteur orthographique
- /usr/lib/uucp : programmes auxiliaires pour uucp
- /usr/local : tout ce qui est local à un site. Reproduit l'organisation de /usr pour mettre le logiciels du domaine publique, etc. dans /usr/local/bin, /usr/local/man, /usr/local/src, etc. ou dans des sous-répertoire comme /usr/local/corba contenant



lui-même un bin, man, include, etc. si l'ensemble est gros

/usr/old : des vieilleries

/usr/openwin : contient le logiciel de fenêtrage OpenWindows basé sur X11

/usr/platform : toute une arborescence de type /platform contenant des choses spécifique à une plate-forme mais non nécessaire au démarrage

/usr/sadm : fichiers et répertoires pour l'administration système

/usr/sadm/install : programmes de gestion des packages

/usr/sbin : commandes d'administration

/usr/sbin/static : version statiquement compilée de certaines commandes en cas de problème avec les bibliothèques dynamiques



/usr/share : fichiers partageables et indépendants de l'architecture

/usr/share/lib : bases de données indépendantes de l'architecture

/usr/share/lib/terminfo : fichiers de description de terminaux pour terminfo

/usr/share/lib/zoneinfo : fichiers décrivant les fuseaux horaires

/usr/share/man : manuels en ligne

/usr/share/src : les sources si disponibles

/usr/ucb : commandes compatible Berkeley (BSD)

/usr/ucbinclude : entêtes à la Berkeley

/usr/ucb/lib : bibliothèques Berkeley

/usr/xpg4 : commandes POSIX.2



- Une machine peut exporter des disques systèmes vers des machines sans disque ou avec peu de disque
- Définition d'une hiérarchie d'exportation standard

`/export/exec/architecture-name` : /usr pour l'architecture donnée et la version courante du système d'exploitation

`/export/exec/architecture-name.release-name` : /usr pour l'architecture donnée et une version précise du système d'exploitation

`/export/exec/share` : version exportée de /usr/share

`/export/exec/share.release-name` : pour une autre version du système

`/export/root/hostname` : le / de la machine *hostname*



`/export/swap/hostname` : le fichier de *swap* de la machine *hostname*

`/export/var/hostname` : /var de la machine *hostname*



- Besoin de gérer l'ajout et l'enlèvement de logiciels complets déjà compilés
- Système V possède la notion de paquets (*packages*)
- Fichier contenant le logiciel lui-même ainsi que ses fichiers de configuration mais aussi scripts d'installation et de dés-installation, de gestion de dépendance
- Solaris est lui-même découpé en paquets. Permet de spécialiser l'installation du système
- Site de logiciels libre <http://sunfreeware.com>
- Gestion par `admintool` ou `pkgadd` et `pkgrm`
- Gestion d'une base de donnée des paquets et fichiers installés. Affichage des logiciels installés par `pkginfo [-1]`



- Convention de nommage des paquets : commence par un identificateur du producteur. Paquets de Sun commencent par SUNW



- Installations peuvent poser des questions
- Faire un fichier de réponse avec `pkgask`
-  Interaction entre installation et partage de ressources entre machines
- Faire le tri entre les paquets pour savoir si l'installation est locale ou centralisée
- Installer les parties centralisées sur les serveurs (typiquement dans `/opt`, `/usr/opt` si partagé et `/usr/local`) et les parties non partagées sur toutes les machines (`/etc`, `/usr/opt` si non partagé)
- Problème si installation centralisée pour une machine dont les fichiers ne sont pas au même endroit sur le serveur (`/export/root/machine` au lieu de `/`) : nécessité de rajouter



une option pour changer la position de la racine  
`pkgadd -R /export/root/machine SUNWntpr`



- Contient le système en plus de l'installation de base
- Beaucoup d'outils « standard » et utiles ne font pas partie de la distribution Solaris néanmoins livré avec un CD de *freeware*
- Gros travail pour avoir les logiciels à jour  
↔ Essayer de mutualiser les efforts si on a confiance
- Comment installer des logiciels ?

- ▶ Compiler soi-même. Avec de la chance :

```
gtar zxvf zsh-3.1.6.tar.gz
cd zsh-3.1.6
./configure
make
et en temps que root :
make install
```



```
make install.info
```

- ▶ Récupérer des paquets tous faits

```
http://www.sunfreeware.com/
```

```
pkgadd -d openssh-2.9p2
```

- ▶ Utiliser l'installation de paquet automatique avec `pkg-get`

- Garder une trace des installations et de la méthode, par exemple

```
/usr/local/src/Systeme/zsh.README
```



- Gère à la fois le téléchargement et l'installation
- Système du style apt-get de Debian mais ne gère pas les dépendances
- Installer d'abord <http://www.bolthole.com/solaris/pkg-get.html>

```
pkgadd -d BOLTpget.pkg
```

Choisir dans `/etc/pkg-get.conf` un serveur plus proche :

```
url=ftp://ftp.worldonline.fr/pub/SunFreeware
```

- ▶ `pkg-get update` récupère la liste des paquets disponibles
- ▶ `pkg-get available` affiche la liste des paquets disponibles
- ▶ `pkg-get compare` compare ce qui est installé avec ce qui est disponible
- ▶ `pkg-get install paquet` récupère et installe le logiciel



## Exemple d'organisation de /usr/local

**BerkeleyDB.3.2** : base de donnée de Berkeley

SunOS4 : pointe vers un `/usr/local` SunOS 4.1.4 qui contient des programmes antiques mais néanmoins utiles exécutés en mode compatibilité binaire le cas échéant

X : tout ce qui concerne le système de fenêtrage X11R6 et des applications ;

apache : serveur WWW Apache <http://www.apache.org>

bind : serveur de nom de domaine Internet BIND/named

daVinci\_V2.1 : logiciel d'affichage de graphe daVinci

<http://www.informatik.uni-bremen.de/~davinci> utilisé dans le projet PIPS <http://www.cri.ensmp.fr/pips>

doc : contient des documentations diverses



`etc` : conserve des informations de configuration pour des logiciels locaux tel que le système de sauvegarde Amanda ou des démons

`include` : conserve les fichiers de type `#include` venant avec les logiciels GNU par exemple

`info` : centralise les documentations au format `info`

`java` : contient tous les rajouts concernant le langage Java

`lib` : stocke les bibliothèques statiques ou dynamiques ainsi que certaines bases de données ou dictionnaires associées aux logiciels de `/usr/local`

`libexec` : contient des programmes utilisés de manière interne par les divers logiciels

`man` : centralise tous les manuels locaux



`netscape` : contient la dernière version du visionneur WWW de Netscape

`pure` : logiciels de mise au point et d'analyse de l'usage de la mémoire Purify

`samba` : système Samba gérant le protocole SMB de communication avec les logiciels MicroSoft

`sbin` : programmes utilisés plus particulièrement lors des tâches d'administration (sauvegardes, configuration,...) ; ces programmes ne sont pas toujours liés statiquement

`share` : regroupe ce qui est indépendant de l'architecture et du système de la machine utilisant cet `/usr/local`. Il y a en particulier :

`a2ps` : fichiers utilisés par l'utilitaire d'impression vers PostScript  
`a2ps`



- cfengine : fichiers de configuration du système cfengine utilisé pour mettre en place tous les systèmes de toutes les machines
- conf : tout ce qui est lié à l'installation et à la maintenance automatique du système au CRI. Contient des fichiers de références et des utilitaires
- emacs : fichiers de configuration de l'éditeur de texte qui fait tout et même plus, Emacs. En particulier tous les rajouts locaux vont dans /usr/local/share/emacs/site-lisp et le fichier emacsrc.el que les utilisateurs locaux incluent dans leur .emacs
- ghostscript : principalement les fontes liées au moteur GhostScript compatible PostScript
- lib : bibliothèques indépendantes de l'architecture



- locale : les messages utilisés par les différents logiciels traduits en différentes langues
- sparc-sun-solaris2.8 : spécifiques à cette version du système. En particulier si d'autres versions du système se partagent /usr/local il y aura d'autres répertoires de ce type
- spirit : le logiciel d'indexation en texte intégral et à analyse sémantique
- src : regroupe tous les sources (éventuellement seulement des packages) des logiciels installés en plus du système de base sur les machines. Rangement en :

|        |       |            |        |         |       |
|--------|-------|------------|--------|---------|-------|
| Compil | Java  | Maths      | PC     | Sun     | XML   |
| Corba  | Knuth | Multimedia | Perl   | Systeme | intel |
| Crypto | Linux | Musique    | Reseau | TeX     |       |
| Emacs  | Mail  | Office     | SQL    | Web     |       |



GNU            Makefile      Outils          Security      X11

Chaque *programme* est accompagné d'un *programme*.README décrivant la source et l'installation. Sous RCS

ssl : une implémentation du protocole SSL de *socket* sécurisées utilisé par exemple pour les connexions WWW par le protocole HTTPS

teTeX : le système de composition de texte basé sur T<sub>E</sub>X dans sa distribution teTeX <http://www.tug.org/teTeX> pour Unix

var : le répertoire des choses variant potentiellement beaucoup liées aux logiciels de /usr/local tels que les bases de données utilisées par le système de sauvegarde



- Mises à jour de morceaux du système d'exploitation (extensions, corrections de bugs,...)
- Extension du système de paquets. Regroupe plusieurs paquets
- patchadd/patchrm
- showrev -p affiche la liste de tous les patches installés
- Nommage : *numéro-de-patch.numéro-de-version*
- Récupération des patches depuis le site de Sun
- Les plus importants <http://www.sun.com/bigadmin/patches>



- Trop d'informations spécifiques à 1 site pour être dupliquées sur chaque machine
- Besoin de pouvoir remettre à jour les informations instantanément sur toutes les machines
- Paramétrage de la source des information pour les `get.X by Y ()` (`gethostbyname()`, `getpwnam()`,...)
- Différents systèmes possibles sur Solaris sélectionnés par type de ressources selon `/etc/nsswitch.conf`
  - ▶ Fichiers
  - ▶ NIS
  - ▶ NIS+
  - ▶ DNS : système de nommage des machines sur Internet. Plutôt réservé à la ressource `hosts`



- ▶ LDAP sur TLS
- ▶ Federated Naming Service (FNS) : méthode de nommage fédérant NIS+, NIS, fichiers, DNS, et X.500/LDAP. API XFN dépassant les `get.X by Y ()`



- Pas un système de nommage réseau
- À dupliquer sur toutes les machines... mais robuste
- Nommage en réseau « à la main » : mettre en place un système de synchronisation des fichiers depuis une base centrale
- /etc/nsswitch.conf typique :

```
passwd: files | netmasks: files
group: files | bootparams: files
hosts: dns files | publickey: files
ipnodes: files | netgroup: files
networks: files | automount: files
protocols: files | aliases: files
rpc: files | services: files
ethers: files | sendmailvars: files
```

Administration Solaris—Systèmes de nommage—



DÉPARTEMENT INFORMATIQUE — ENST BRETAGNE



```
printers: user files | prof_attr: files
auth_attr: files | project: files
```



- Distribue des tableaux associatifs
- Pas très sécurisé en local car les mots de passe cryptés sont accessibles pour essais de craquage
- Système très répandu
- Choisir un nom de domaine NIS unique sur son réseau. Par exemple nom de domaine Internet
- Vérifier que `/etc/nodename` contient bien le nom de la machine
- Mettre dans `/etc/defaultdomain` le nom du domaine
  - ▶ Est utilisé par le système pour la commande `domainname`
  - ▶ `domainname` permet de connaître ou modifier le nom de domaine NIS
- Choisir le serveur NIS maître et les serveurs esclaves



- Faire `/usr/sbin/ypinit -m` sur le serveur maître qui demande la liste de tous les serveurs
- Si `/etc/resolv.conf` existe sur le serveur les NIS font suivre les requêtes `hosts` au DNS
- Faire `/usr/sbin/ypinit -c` sur le serveur esclave qui demande la liste de tous les serveurs. Donner son propre nom en premier, puis le serveur maître puis les autres
- Sur les clients faire un `ypinit -c` et donner la liste des serveurs, les plus proches en premier
- Remplacer les `/etc/nsswitch.conf` avec le contenu de `/etc/nsswitch.nis`
- `/var/yp/Makefile` contrôle les tables NIS exportées. Éditer pour rajouter ses propres ressources. Faire `make` dans `/var/yp`



pour propager les ressources lorsque les fichiers de référence sont modifiés

- `/var/yp/securenets` restreint l'accès des NIS
- `ypcat [-k] table` permet d'afficher le contenu d'une table NIS
- `ypcat -k ypservers` affiche liste des serveurs NIS
- `ypwhich` donne le serveur actuellement utilisé
- `ypwhich -m table` indique le serveur maître d'une table
- `ypwhich -x` donne la liste des alias



- Organisation hiérarchique de l'espace de nommage
- Distribue tout type d'information (binaires...)
- Serveurs secondaires pour répartition de la charge et secours
- Mécanisme d'authentification et d'autorisation tables par tables
- Mots de passes chiffrés pas lisibles par les utilisateurs
- Système plutôt Sun



- Interfaces
- Routage
- Nommage
- Services



## Interface

- Attribution des noms IPv4 par fichiers  
`/etc/hostname.interface`
- Attribution des noms IPv6 par fichiers  
`/etc/hostname6.interface`
- Attribution des masques réseaux définie réseau par réseau dans  
`/etc/inet/netmasks`  
`192.44.75.0 255.255.255.0`
- Nom principal de la machine dans `/etc/nodename`
  - ▶ Utilisé par le système pour appeler `hostname`
  - ▶ Commande `hostname` utilisée pour connaître (comme avec `uname -n`) ou établir le nom principal de la machine
- Manipulation des paramètres de l'interface avec `ifconfig`



- Affichage des caractéristiques des interfaces avec :

```
gavotte.enst-bretagne.fr-keryell > ifconfig -a
lo0: flags=1000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4> mtu 8232 index 1
 inet 127.0.0.1 netmask ff000000
hme0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 2
 inet 192.44.75.87 netmask ffffffff broadcast 192.44.75.255
lo0: flags=2000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv6> mtu 8252 index 1
 inet6 ::1/128
hme0: flags=2000841<UP,RUNNING,MULTICAST,IPv6> mtu 1500 index 2
 inet6 fe80::a00:20ff:fec6:bc6b/10
hme0:1: flags=2080841<UP,RUNNING,MULTICAST,ADDRCONF,IPv6> mtu 1500 index 2
 inet6 2001:660:283:2:a00:20ff:fec6:bc6b/64
```



- Routeur(s) par défaut dans `/etc/defaultrouter`  
192.44.75.1
- Si routage RIP, `routed` utilise la définition des passerelles dans `/etc/gateways`
- Affichage des paramètres de routage avec :

```
gavotte.enst-bretagne.fr-keryell > netstat -r
```

```
Routing Table: IPv4
```

| Destination                   | Gateway    | Flags | Ref        | Use      | Interface |
|-------------------------------|------------|-------|------------|----------|-----------|
| galaxie-net0.enst-bretagne.fr | gavotte    |       | U          | 1 120827 | hme0      |
| BASE-ADDRESS.MCAST.NET        | gavotte    | U     | 1          | 0        | hme0      |
| default                       | galaxie-75 | UG    | 1          | 6809     |           |
| localhost                     | localhost  | UH    | 2616389330 |          | lo0       |



Routing Table: IPv6

| Destination/Mask    | Gateway                           | Flags | Ref | Use  | If   |
|---------------------|-----------------------------------|-------|-----|------|------|
| 2001:660:283:2::/64 | 2001:660:283:2:a00:20ff:fec6:bc6b | U     |     | 1    | 0 hm |
| fe80::/10           | fe80::a00:20ff:fec6:bc6b          | U     | 1   | 0    | hme0 |
| ff00::/8            | fe80::a00:20ff:fec6:bc6b          | U     | 1   | 0    | hme0 |
| default             | fe80::280:c8ff:fee7:63bc          | UG    | 1   | 0    | hme0 |
| localhost           | localhost                         | UH    | 1   | 1490 | lo0  |

Option `-n` pour supprimer la traduction adresse vers nom (quand le DNS ne marche pas...)

- Manipulation des tables avec la commande `route`
- `route monitor` pour surveiller l'état des routes
- `route get destination` pour voir par où cela passerait



- `/etc/inet/hosts` associe numéro IP et noms
 

```
192.44.75.8 antares.enst-bretagne.fr antares timehost
192.44.75.87 gavotte.enst-bretagne.fr gavotte loghost
192.44.75.98 peyresourde.enst-bretagne.fr peyresourde
```
- `/etc/inet/ipnodes` version IPv4 et IPv6 de `/etc/inet/hosts`
- Requêtes DNS envoyées suivant `/etc/resolv.conf`

```
nameserver 192.44.75.10
nameserver 192.108.115.2
nameserver 192.44.77.1
domain enst-bretagne.fr
search enst-bretagne.fr enstb.org
```
- `/etc/netgroup` définit des groupes de (*machines, personnes, domaines*) utilisés principalement pour



des autorisations dans `.rhosts` ou NFS

```
ens-rec (,brouty,) (,leroy,) (,thepaut,) (,lhostis,) \
(,beugnard,) (,bourget,) (,cousin,) (,piriou,) (,derrien,) \
(,ouvradou,) (,floch,) (,mercel,) (,laisne,) (,lasquel,) \
(,ledrezen,) (,geffroy,) (,prou,) (,ogor,) (,rannou,) \
(,retif,) (,kermarre,) (,duault,) (,berre,) (,legleau,) \
(,madec,) (,choukair,) (,keryell,) (,gravey,) (,mallet,) (,segarra,)
invites (,sun,) (,bellot,) (,guerin,) (,pucci,)
sun-chercheurs (gavotte.enst-bretagne.fr,,) (xatard.enst-bretagne.fr,,) \
(fambetou.enst-bretagne.fr,,) (cardonille.enst-bretagne.fr,,) \
(puymorens.enst-bretagne.fr,,) (aspin.enst-bretagne.fr,,) \
(palmarella.enst-bretagne.fr,,) (rodoumouls.enst-bretagne.fr,,) \
(uglas.enst-bretagne.fr,,) (betelgeuse.enst-bretagne.fr,,) \
(peyresourde.enst-bretagne.fr,,)
```



- Association nom et protocole/port : `/etc/inet/services`

```
telnet 23/tcp
smtp 25/tcp mail
domain 53/udp
domain 53/tcp
```

- Super-démon `inetd` qui écoute économiquement plusieurs ports et lance des clients en conséquence en fonction de `/etc/inet/inetd.conf`

```
ftp stream tcp6 nowait root /usr/sbin/in.ftpd in.ftpd
telnet stream tcp6 nowait root /usr/sbin/in.telnetd in.telnetd
shell stream tcp nowait root /usr/sbin/in.rshd in.rshd
shell stream tcp6 nowait root /usr/sbin/in.rshd in.rshd
login stream tcp6 nowait root /usr/sbin/in.rlogind in.rlogind
```



- Services lancés directement au démarrage depuis `/etc/init.d` dans les `/etc/rcX.d`



- Géré par l'outil `sendmail` <http://www.sendmail.org>
- <http://docs.sun.com:80/ab2/coll.47.11/SYSADV3/@Ab2PageView/44211?Ab2Lang=C&Ab2Enc>
- Configuration ultra-simplifiée par défaut :
  - ▶ Les machines standard se contentent d'envoyer le courrier à une machine centrale
  - ▶ Machine centrale `mailhost.mon.domaine` joue le rôle de facteur avec Internet



- ▶ Monter `/var/mail` depuis le serveur central via `vfstab` ou `auto.direct`, avec les options `rw,hard,actimeo=0`
- ▶ Chaque outil de lecture de courrier ouvre `/var/mail/utilisateur`



## Le sendmail central routeur de courrier

- Mettre le fichier de configuration de `sendmail` pour un facteur :  
`cp /etc/mail/main.cf /etc/mail/sendmail.cf`
- Créer un alias `mailhost.mon.domaine` dans le système de nommage pour cette machine
- Mettre en place l'utilisation du DNS dans `/etc/nsswitch.conf` au niveau de `hosts`:
- Créer les alias de courrier nécessaire dans `/etc/mail/aliases`, au moins pour `postmaster`, `root`, `adm`,...
- Faire pointer au niveau du DNS tous les MX de toutes les machines vers ce serveur de courrier
- Relancer le tout :  
`/etc/init.d/sendmail stop; /etc/init.d/sendmail start`



- Les fichiers `~/ .forward` permettent aux utilisateurs de rediriger leur courrier
- `mailq` permet d'afficher les courriers en attente de partance

Pour plus d'information voir mon cours sur `sendmail`



- Système de reconfiguration dynamique du matériel (rajout de cartes sans arrêter le système) sur Sun haut de gamme
- Beaucoup de matériels (PC...)  $\rightsquigarrow$  beaucoup de *device drivers*
- Néanmoins la majorité sont déjà disponibles sur un système mais ne sont chargés qu'à la demande lors d'une auto-reconfiguration (gain de mémoire, évite de reconstruire le noyau à chaque fois)
- Conducteurs de matériels placés dans `/kernel/drv` et `/platform/`uname -m`/kernel/drv` et fichiers de configuration associés en extension `.conf` en cas de besoin spécifique
- Rajout de conducteurs via `pkgadd` par exemple



- Vérifier que le conducteur existe pour le périphérique et le rajouter si besoin est
- Arrêter en demandant une reconfiguration du noyau :  
redémarrage après création d'un fichier `/reconfigure` ou bien redémarrage avec option `-r`
- Éteindre la machine puis les périphériques
- Rajouter le périphérique
- Rallumer les périphériques puis la machine
- Redémarrer le système
- Vérifier que le nouveau périphérique est pris en compte et apparaît par exemple dans `/dev`



- Bien préparer tout le matériel
- Geler le système avec `L1-A`
- Rajouter et allumer le périphérique
- Continuer le système avec `go` ↔ interruption du service de quelques secondes
- Reconfigurer le système à chaud (en s'inspirant du fichier de reconfiguration du reboot)

```
/usr/sbin/drvconfig
/usr/sbin/devlinks
/usr/sbin/disks
/usr/sbin/ports
/usr/sbin/tapes
/usr/sbin/audlinks
```



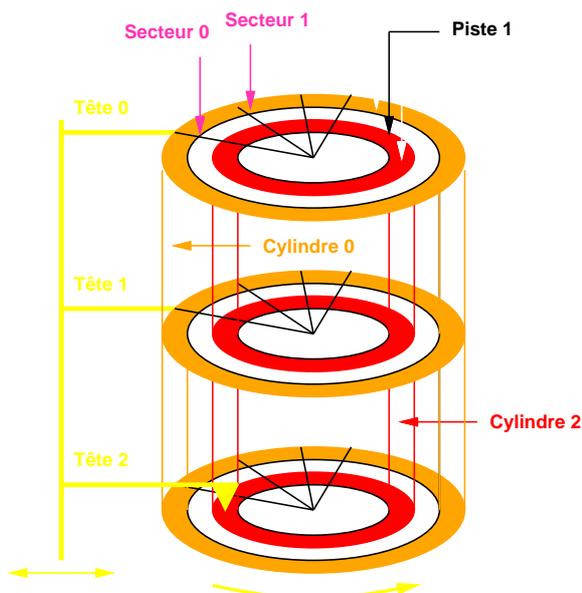
- dmesg affiche messages début du boot. Contient entre autre liste des périphériques reconnus

```
Jan 17 18:18
SunOS Release 5.7 Version Generic [UNIX(R) System V Release 4.0]
Copyright (c) 1983-1998, Sun Microsystems, Inc.
mem = 65136K (0x3f9c000)
avail mem = 51163136
root nexus = i86pc
isa0 at root
pci0 at root: space 0 offset 0
 IDE device at targ 0, lun 0 lastlun 0x0
 model ST36531A, stat 50, err 0
[...]
```

- sysdef affiche information sur paramètres du noyau : device drivers, paramètres du noyau, modules chargeables, mémoire
- prtconf affiche la liste des périphériques



( Disques magnétiques )



FFS optimisé pour les disques :



- Partition (ou tranche) : ensemble de cylindres consécutifs ↔ ↗  
localité
- Allocation dans des cylindres consécutifs
- Allocation dans des secteurs consécutifs avec un saut (temps de rotation)
- Laisse des cylindres vides régulièrement pour allouer plus rapidement de nouveaux secteurs

Problèmes des caches dans les contrôleurs disque qui éloignent de la réalité...



- Découpe des disques pour des usages différents
- Augmente la localité (et donc performances) des accès au sein de chaque partition
- Limites infranchissables (contre certains utilisateurs expansifs)
- Fournit des zones brutes pouvant avoir chacune leur système de fichier (indépendant et même de type différent) voir sans (swap, base de donnée)
- Peuvent avoir des politiques d'exportation différentes sous NFS
- ⚠ Éviter d'avoir 2 partitions qui se recouvrent sans raison...
- Partitionnement fait automatiquement et graphiquement par la procédure d'installation
- Mais en cas de problème sur un disque, de remplacement, de



changement du partitionnement : connaissance utile

- Chaque système d'exploitation a sa convention de partitionnement (n'est pas déterminé au niveau du disque lui-même)
- Solaris découpe en 8 ou 10 partitions avec comme convention l'usage courant
  - 0 : contient /
  - 1 : du swap
  - 2 : tout le disque (déborde sur les autres...)
  - 3 : /export sur un serveur
  - 4 : /export/swap sur un serveur
  - 5 : /opt
  - 6 : /usr



- 7 : /home ou /export/home
- 8 : sur PC contient le système de boot et pointe au début du disque
- 9 : sur PC contient les blocs alternatifs utilisés à la place d'autres en panne et pointe après la partition 8
- Sur PC nécessité d'une « convention collective des OS » pour faire du multi-OS. Convention de partitionner un disque jusqu'en 4 partitions via `fdisk`. Solaris prend une de ces partitions et la repartitionne avec son propre système
-  La loi de Murphy veut que le partitionnement choisi n'est jamais le bon... En général, / et le swap sont trop petits
-  Loi de Murphy numéro 2 : difficile de changer le partitionnement dynamiquement...



- /dev/dsk pour accéder aux disques en mode bloc
- /dev/rdisk pour accéder aux disques en mode bloc et brut de fonderie (que par blocs de 512 octets, moins pratique mais plus rapide)
- Contrôleur avec un disque relié directement dessus (IDE, Xylogics)
  - ▶ SPARC : /dev/[r]dsk/cx dy sz  
x : numéro logique contrôleur  
y : numéro disque  
z : numéro partition Solaris (0 à 7)
  - ▶ Intel : /dev/[r]dsk/cw dx pz  
w : numéro logique contrôleur  
x : numéro disque



- z : numéro partition PC (fdisk) (1 à 4, partition 0 représente ≡ tout le disque)
- /dev/[r]dsk/cw dx sy est le partitionnement Solaris à l'intérieur de la partition PC Solaris
- y : numéro partition Solaris (0 à 9)
- Contrôleur orienté bus (SCSI, IPI) : plusieurs systèmes indépendants connectés au bus, chaque système peut avoir des disques
    - ▶ SPARC : /dev/[r]dsk/cwt x dy sz  
w : numéro logique contrôleur  
x : numéro cible sur le bus (numéro unité logique SCSI typiquement)  
y : numéro disque  
z : numéro partition Solaris (0 à 7)



- ▶ Intel : `/dev/[r]dsk/cwtxdypt`
    - `w` : numéro logique contrôleur
    - `x` : numéro cible sur le bus (numéro unité logique SCSI typiquement)
    - `y` : numéro disque
    - `t` : numéro partition PC (`fdisk`) (1 à 4, partition 0 représente  $\equiv$  tout le disque)
- `/dev/[r]dsk/cwtxdysz` est le partitionnement Solaris à l'intérieur de la partition PC Solaris
- `z` : numéro partition Solaris (0 à 7)
- Par convention, partition numéro 2 de Solaris contient tout le disque (ou toute la partition PC `fdisk` réservée à Solaris)
  - CD-ROM : comme un disque standard. Généralement caché par le volume manager



Utilisation :

- Installation d'un nouveau disque :
  - ▶ Formattage
  - ▶ Partitionnement
- Affiche les disques reconnus sur le système
- Affiche des informations et leur partitionnement
- Test d'un disque
- Réparation d'un disque
- Destruction du contenu (sensible...) avant renvoi



Les commandes sont abrégées

- partition gère et affiche le partitionnement (prtvtoc donne aussi l'information)

```
partition> p
Current partition table (original):
Total disk cylinders available: 253 + 2 (reserved cylinders)
```

| Part | Tag        | Flag | Cylinders | Size     | Blocks            |
|------|------------|------|-----------|----------|-------------------|
| 0    | root       | wm   | 3 - 28    | 203.95MB | (26/0/0) 417690   |
| 1    | swap       | wu   | 29 - 170  | 1.09GB   | (142/0/0) 2281230 |
| 2    | backup     | wm   | 0 - 252   | 1.94GB   | (253/0/0) 4064445 |
| 3    | unassigned | wm   | 0         | 0        | (0/0/0) 0         |
| 4    | unassigned | wm   | 0         | 0        | (0/0/0) 0         |
| 5    | unassigned | wm   | 0         | 0        | (0/0/0) 0         |
| 6    | usr        | wm   | 171 - 252 | 643.23MB | (82/0/0) 1317330  |
| 7    | unassigned | wm   | 0         | 0        | (0/0/0) 0         |
| 8    | boot       | wu   | 0 - 0     | 7.84MB   | (1/0/0) 16065     |
| 9    | alternates | wu   | 1 - 2     | 15.69MB  | (2/0/0) 32130     |



Adresses aussi en *cylindre/tête/bloc*

Format propose un partitionnement par défaut

- current décrit le disque courant

```
format> cu
Current Disk = c0t4d0: bassine
<SEAGATE-ST39102LW-0004 cyl 6922 alt 2 hd 12 sec 214>
/sbus@1f,0/espdma@e,8400000/esp@e,8800000/sd@4,0
```

- format reformate le disque
- backup récupère un label (la table des matières du disque) de secours en cas de perte du label principal
- analyse permet de tester le disque avec un effet plus ou moins destructeur
- repair répare 1 bloc du disque en le rajoutant dans la liste des



défectueux et en remet un autre à la place. En cas de problème matériel

- `defect` permet de gérer la liste des défauts (1 gros disque est rarement parfait...)
- `volname` donne un nom au disque. Au CRI on donne des noms de conteneurs pour s'y retrouver. `goutte` aura moins d'octets que `bassine`. Même nom qu'on retrouve monté
- `label` entérine les modifications

`/etc/format.dat` contient les paramètres de formatage (géométrie, etc) des disques connus

- Disque récent (SCSI-2) en bon état : informe directement `format`
- Sinon, lire la documentation ou récupérer un `format.dat` récent (ou le contraire !)



## Étape fdisk sur PC

- Partage du disque entre plusieurs OS

```
Total disk size is 788 cylinders
Cylinder size is 16065 (512 byte) blocks
```

| Partition | Status | Type        | Cylinders |     | Length | %  |
|-----------|--------|-------------|-----------|-----|--------|----|
|           |        |             | Start     | End |        |    |
| 1         |        | IFS: NTFS   | 0         | 50  | 51     | 6  |
| 2         |        | DOS-BIG     | 51        | 101 | 51     | 6  |
| 3         | Active | Solaris     | 102       | 356 | 255    | 32 |
| 4         |        | UNIX System | 357       | 592 | 236    | 30 |

SELECT ONE OF THE FOLLOWING:

1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Exit (update disk configuration and exit)
5. Cancel (exit without updating disk configuration)

Enter Selection:



- 1 seule partition Solaris par disque
- Partition Solaris alignée sur 1 cylindre
- Épargner le *Master Boot Record* sur le cylindre 0
- Subtilité
  1. On formate le disque avec `format`
  2. On partitionne globalement avec `fdisk` appelable directement depuis `format`
  3. On partitionne la partition Solaris générée avec `format` à nouveau...
- Mode non interactif pour extraire des configurations et configurer de manière précise un disque brute style `/dev/rdisk/c0t0d0p0`. Intérêt pour faire des installations automatiques multi-système d'exploitation



- Utilise le Virtual File System : définit une interface permettant de rajouter assez simplement un nouveau type de système de fichiers
- Masque les détails : possibilité de lire, écrire, consulter, etc. quel que soit le type de système de fichiers (local, distant,...)
- Systemes de fichiers de type disque local
  - UFS** : Unix File System, basé sur le FFS 4.3BSD, Système de fichier par défaut
  - HSFS** : High Sierra et ISO-9660 (version officielle de la précédente) pour CD-ROM. Lecture seule. Extension Rock Ridge fournissant la sémantique UFS (sauf les liens durs et... l'écriture !)
  - PCFS** : lecture et écriture sur des disques au format MS-DOS



(typiquement disquettes)

**S5FS** : lecture et écriture sur des disques au format System V sur PC

- Système de fichiers de type accès distant

**NFS** : Network File System pour accéder à des fichiers distants comme s'ils étaient locaux (modulo des différences de performance)

- Système de fichiers virtuels

**CacheFS** : Cache File System pour stocker localement une copie rapide. CD-ROM, Intranet distant,...

**TMPFS** : Temporary File System pour stocker en mémoire pour aller très vite. Configuration par défaut de `/tmp` (accélération des compilations...) qui est doublement volatil

**LOFS** : Loopback System pour faire apparaître à un autre



endroit une partie de la hiérarchie (y compris montages NFS)

LOFI *Loopback file driver* permet de générer un pilote brut à partir d'une image fichier

- ▶ Montage de l'image d'un CD-ROM

```
lofiadm -a $CD/sol-8-u5-sparc-v1.iso
mount -F hsfs -o ro /dev/lofi/1 /mnt
```

- ▶ Montage de l'image d'une disquette

**PROCFS** : Process System montre la liste des processus en train de tourner sous forme de répertoires. Utilisé par des outils de debug et d'analyse

5 autres systèmes de fichiers à usage interne sans administration particulière

- Tâche de l'administrateur ?
  - ▶ Créer de nouveaux systèmes de fichiers



- ▶ Rendre les ressources locales et distantes accessibles aux utilisateurs
- ▶ Connexion et ajout de nouveaux disques
- ▶ Mise en place d'une *excellente* politique de sauvegarde
- ▶ Vérification et correction des fichiers endommagés  
Pour les hackers : `fsdb` un débogueur de système de fichiers pour récupérer un accident, `grep` du device,...
- Commandes générique : `mount`, `umount`, `mkfs`, `fsck`,... acceptent l'option `-F fs-type` et appellent en fait `mount`, `umount_fs-type`, `mkfs_fs-type`, `fsck_fs-type`,... Voir les documentations de ces dernières commandes pour les détails intrinsèques



## Montage/démontage d'un système de fichiers

- Pour accéder à un système de fichier : montage pour attacher le système à un répertoire de la hiérarchie préexistante
- / est toujours monté (lancement du noyau) et indémontable
- Montage cache les fichiers préexistants dans le répertoire
- Démontage du système de fichier fait réapparaître d'éventuels fichiers préexistants
- Démontage possible seulement si plus aucun process n'utilise le système de fichier
- Démontage utile pour faire une sauvegarde d'une partition en étant sûr que personne ne la modifie
- Arrêt du système utilise une procédure de démontage



/etc/mnttab fichier en écriture seulement

```

/proc /proc proc rw,suid,dev=2940000 915634700
/dev/dsk/c0d0s0 / ufs rw,suid,dev=1980000,largefiles 915634700
/dev/dsk/c0d0s6 /usr ufs rw,suid,dev=1980006,largefiles 915634700
fd /dev/fd fd rw,suid,dev=2a00000 915634700
swap /tmp tmpfs rw,dev=1 915634702
palo-alto:/opt /opt nfs ro,dev=2b40001 915634709
palo-alto:/usr/local /usr/local nfs ro,dev=2b40002 915634710
auto.home /home autofs ignore,indirect,hard,bg,intr,dev=2b80001 915634711
-xfn /xfn autofs ignore,indirect,dev=2b80002 915634711
-hosts /net autofs ignore,indirect,nosuid,nobrowse,dev=2b80003 915634711
auto.direct /var/mail autofs ignore,direct,dev=2b80004 915634711
auto.direct /users autofs ignore,direct,dev=2b80005 915634711
voltaire.ensmp.fr:vold(pid246) /vol nfs ignore,noquota,dev=2b40003 915634726
palo-alto:/export/verre1 /users nfs rw,hard,intr,dev=2b400ca 916307121
palo-alto:/export/verre2/var/mail /var/mail nfs rw,hard,intr,actimeo=0,dev=2b402aa 916663055

```

df affiche aussi la liste des systèmes de fichiers montés avec de l'information sur la place disponible

| Filesystem      | 1024-blocks | Used   | Available | Capacity | Mounted on |
|-----------------|-------------|--------|-----------|----------|------------|
| /dev/dsk/c0d0s0 | 195580      | 34359  | 141663    | 20%      | /          |
| /dev/dsk/c0d0s6 | 618904      | 481994 | 81209     | 86%      | /usr       |



|                                   |         |         |         |     |            |
|-----------------------------------|---------|---------|---------|-----|------------|
| swap                              | 1083936 | 616     | 1083320 | 0%  | /tmp       |
| palo-alto:/opt                    | 2733423 | 2312742 | 407014  | 85% | /opt       |
| palo-alto:/usr/local              | 2733423 | 2312742 | 407014  | 85% | /usr/local |
| auto.direct                       | 2061938 | 20124   | 2034941 | 1%  | /var/mail  |
| auto.direct                       | 2061938 | 1892023 | 163042  | 92% | /users     |
| palo-alto:/export/verre1          |         |         |         |     |            |
|                                   | 2061938 | 1892023 | 163042  | 92% | /users     |
| palo-alto:/export/verre2/var/mail |         |         |         |     |            |
|                                   | 2061938 | 20124   | 2034941 | 1%  | /var/mail  |



- Déclaration dans la table `/etc/vfstab` pour montage au démarrage

| #device              | device            | mount      | FS    | fsck | mount   | mount   |
|----------------------|-------------------|------------|-------|------|---------|---------|
| #to mount            | to fsck           | point      | type  | pass | at boot | options |
| fd                   | -                 | /dev/fd    | fd    | -    | no      | -       |
| /proc                | -                 | /proc      | proc  | -    | no      | -       |
| /dev/dsk/c0d0s1      | -                 | -          | swap  | -    | no      | -       |
| /dev/dsk/c0d0s0      | /dev/rdisk/c0d0s0 | /          | ufs   | 1    | no      | -       |
| /dev/dsk/c0d0s6      | /dev/rdisk/c0d0s6 | /usr       | ufs   | 1    | no      | -       |
| palo-alto:/opt       | -                 | /opt       | nfs   | -    | yes     | ro      |
| palo-alto:/usr/local | -                 | /usr/local | nfs   | -    | yes     | ro      |
| swap                 | -                 | /tmp       | tmpfs | -    | yes     | -       |

- ▶ `mount -a` permet de monter tout ce qui est dans la table
- ▶ `mountall` idem mais seulement ce qui est marqué comme montable lors du boot et lance avant un `fsck` si marqué
- ▶ `mount mount-point` monte le système de fichier correspondant décrit dans la table



- ▶ `mount -p` affiche la table de montage courante en format `vfstab` pour ré-injection facile
- Montages infréquents : à la main avec  
`mount [-F FSType] [-o options] [-O] special`  
`mount-point` : -O pour monter au dessus d'un point de montage  
`mount -F pcfs /dev/dsk/c0d0p0:c /mnt` pour la partition  
« C: »
- Auto-monteur : montage à la demande lors de l'utilisation d'un système de fichier  

```
miromesnil-keryell > ls /net
miromesnil-keryell > ls /net/cri.ensmp.fr
export/ usr/
```
- Volume manager : monte automatiquement des média temporaires



- Détache un système de fichier lorsque inutile
- Met à jour (finalise) les structures de données du système de fichier
- Lors de l'arrêt du système (seul moyen de démonter /...)
- `umount mount-point`
- `umountall` essaye de démonte tout sauf le nécessaire vital (`/`, `/proc`, `/var` et `/usr`)
- `umountall -k` tue les processus qui utilisent les partitions à démonter (cf `fuser`)
- Par l'auto-monteur lorsqu'un système de fichier n'est pas utilisé pendant un certain temps



- Unix File System est celui utilisé par défaut sous Solaris. Extension du FFS 4.3BSD. La partition est divisée en groupes de cylindres
- Boot Block** 8 Ko permettant le démarrage. Existe même si pas partition de boot
- Superblock** contient les informations sur le système de fichier : taille, statut, label, taille des blocs, date de dernière modification, nom du dernier répertoire de montage, etc. Contient des drapeaux précisant le fonctionnement
- État** *clean*, *stable*, *active*, *logging* et *unknown*. Permet de savoir où en est le disque lors d'un accident. *clean*, *stable* ou *logging* ne nécessite pas de `fsck`
- Extended Fundamental Type (EFT)** pour avoir des numéros d'utilisateurs, de groupes et de devices sur 32 bits



**Large file systems** système de fichiers de 1 To en tout.

Pratique si stripping/RAIDs à la DiskSuite

**Large files** pour fichiers dépassant les 2 Go. Par défaut

Comme l'information des superblocs est critique, elle est répliquée dans tous les groupes de cylindres et décalée de telle manière qu'elle soit répartie en plus sur tous les plateaux

**Inodes** contiennent toutes les informations sur un fichier sauf son nom : type (normal, répertoire, device,...), mode, propriétaire et groupe, taille, dates,... et tableau de 15 adresses de blocs de données. L'adresse 13 pointe vers un bloc d'adresses, l'adresse 14 pointe vers un bloc d'adresses de blocs d'adresses et l'adresse 15 encore un niveau de plus pour les très gros fichiers

**Blocs de données** stockent le contenu des fichiers et des répertoires (fichiers de noms et d'adresses d'inodes). Blocs



de taille 8 Ko ou 1 Ko (fragments) par défaut

**Blocs libres** blocs non utilisés (ni inodes, ni données, ni blocs d'adresse) par groupe de cylindre. Garde trace de la fragmentation pour limiter sa propagation

Pour des raisons de performance, on arrête le remplissage du disque à 90 % de la capacité pour ne pas perdre trop de temps à chercher de la place

- Journalisation
  - ▶ Penser les modifications aux fichiers sous forme de transactions
  - ▶ Stocker les transactions dans un journal
  - ▶ Appliquer (plus tard) les transactions au système de fichier
  - ▶ Après accident, lors du redémarrage les transactions incomplètes sont éliminées mais les transactions complètes



sont prises en compte ~> cohérence maintenue

- ▶ Plus besoin de faire tourner de longs `fsck` au démarrage
- ▶ Démarré par option `-o logging` au montage
- ▶ Le journal est alloué dans la liste de blocs vides
- `mkfs -F ufs` permet de créer un système de fichier en spécifiant tous les paramètres
- `newfs` crée un système de fichier standard en appelant `mkfs -F ufs` avec des paramètres par défaut

```
deauville-root > newfs -v /dev/rdisk/c0t4d0s5
newfs: construct a new file system /dev/rdisk/c0t4d0s5: (y/n)? y
mkfs -F ufs /dev/rdisk/c0t4d0s5 5926944 214 12 8192 1024 64 2 167 6144 t 0 -1 8 128
/dev/rdisk/c0t4d0s5: 5926944 sectors in 2308 cylinders of 12 tracks, 214 sectors
 2894.0MB in 61 cyl groups (38 c/g, 47.65MB/g, 7936 i/g)
super-block backups (for fsck -F ufs -o b=#) at:
 32, 97840, 195648, 293456, 391264, 489072, 586880, 684688, 782496, 880304,
978112, 1075920, 1173728, 1271536, 1369344, 1467152, 1561376, 1659184,
1756992, 1854800, 1952608, 2050416, 2148224, 2246032, 2343840, 2441648,
```



```
2539456, 2637264, 2735072, 2832880, 2930688, 3028496, 3122720, 3220528,
3318336, 3416144, 3513952, 3611760, 3709568, 3807376, 3905184, 4002992,
4100800, 4198608, 4296416, 4394224, 4492032, 4589840, 4684064, 4781872,
4879680, 4977488, 5075296, 5173104, 5270912, 5368720, 5466528, 5564336,
5662144, 5759952, 5857760,
```

Il peut être utile de stocker cette information pour avoir l'adresse des superblocs en cas de coup dur.

- `tunefs` permet de figurer les paramètres après coup. Moins utile avec tous les caches des disques



- Beaucoup de choses sont faites de manière asynchrone pour accélérer un système de fichiers. `fsflush` effectue les écritures en tâche de fond
- `sync` re-synchronise les disques avec ce que pense l'utilisateur (utile si obligé d'arrêter salement une machine)
- Suite à un reboot intempestif ou une panne matérielle, structures de données incohérentes dans le système de fichier : fichiers à moitiés effacés, superbloc endommagé,...
- Lancement d'un `fsck` au démarrage si un système de fichier n'est pas marqué *clean* (démonté proprement à l'arrêt), *stable* (non démonté proprement à l'arrêt mais non modifié après le dernier `sync` ou `fsflush` avant l'arrêt) ou `log` (système de fichier journalisé). Parcours de toute la structure du disque : long !



Mais analyse de plusieurs disques en parallèle

- ▶ Corrige le superbloc (taille, nombre d'inodes, nombre de blocs et d'inodes libres)
  - ▶ Peut récupérer un superbloc de secours. Si le système est trop HS pour savoir où le trouver, chercher si vous n'avez pas la sortie de `newfs` quelque part sinon faire un `newfs -N` du disque pour faire un système de fichiers pour de faux
  - ▶ Vérification des inodes (nombre de liens vers l'inode, taille, blocs de données référencés 2 fois)
  - ▶ Correction des répertoires « . » et « .. » dans les répertoires
  - ▶ ...
  - ▶ Si fichiers et répertoires (inodes) non référencés dans un répertoire : reliés à `lost+found`
- Certains problèmes sont insolubles automatiquement : choix ~>



questions à l'utilisateur. Possibilité de faire un `fsck` à la main (sur un système de fichier inactif !) avec

```
fsck /dev/rdisk/device-name
```

-  `fsck` n'a aucun moyen de réparer le *contenu* des fichiers...
-  Ne pas monter a priori de disque local via `/etc/vfstab` sans préciser que le `fsck` doit être fait au démarrage. Un `-` dans `/etc/vfstab` indique pas de `fsck`, 1 pour `fsck` séquentiel dans l'ordre du `/etc/vfstab` et plus que 1 pour dire que les `fsck` sont ensuite faits en parallèle sur les disques
-  `fsck` ne remplace pas les RAID et encore moins les sauvegardes ! Évite juste les restaurations en cas de problèmes mineurs
- Pour hackers et pompiers le débogueur de système de fichiers : `fsdb`, `fsdb_ufs`,...



## Système de fichiers cache (CacheFS)

- Utiliser un cache dans disque local pour servir de mémoire cache de fichiers distants
- Application diminution de trafic
  - ▶ Réseau local pour cacher du trafic NFS
  - ▶ Distant (PPP) et Intranet. Possibilité d'exporter le cache en local pour le factoriser entre plusieurs clients locaux
  - ▶ Accélérer la lecture d'un CD-ROM sur un serveur de CD-ROMs (prévoir au moins 650 Mo de cache par CD-ROM...)
- Extension du concept avec *AutoClient* pour cacher même `/` et `/usr` et faire des *network computer* : machine sans système de fichiers local mais un disque qui sert de cache à ses fichiers servis par une machine distante. Utilisation des fichiers du



cache aussi pour le reboot



## Mise en œuvre CacheFS

201

- Créer un répertoire qui contiendra la base de données du cache avec

```
cfsadmin -c répertoire-du-cache
cfsadmin -c /cache
```

- Créer un montage de type CacheFS avec

```
mount -F cachefs -o
backfstype=fstype,cachedir=répertoire-du-cache
back-filesystem mount-point

mkdir /b
mount -F cachefs -o backfstype=nfs,cachedir=/cache palo-alto:/export /b
```

- Contrôler l'efficacité du cache sur un système de fichiers avec

```
root@miromesnil /: cachefsstat /b
/b
taux de succès : 80% (663 sélectionné, 161 manquant)
contrôles de cohérences : 163 (163 réussite, 0 échec)
modifie : 0
défragmentation de la mémoire : 0
```



- Statistiques sur une base de données de cache avec  
`cfsadmin -l repertoire-du-cache`
- Pour améliorer le fonctionnement on peut conseiller au cache de précharger/garder certains fichiers dans le cache
  - ▶ `cachefspack -p liste-de-fichiers`
  - ▶ `cachefspack -f[r] fichier-de-listes` pouvant contenir des expressions régulières, des résultats de commandes exécutées au vol,...
  - ▶ `cachefspack -u liste-de-fichiers`  
`cachefspack -uf fichier-de-listes` pour déverrouiller  
`cachefspack -U repertoire-du-cache` pour déverrouiller tous les fichiers du cache
  - ▶ `cachefspack -i fichier` donne le statut d'un fichier
- `cachefslog` permet de demander au CacheFS de mettre des



messages d'information dans un fichier journal de fonctionnement



- Permet d'avoir plus de mémoire apparente (virtuelle...) que de mémoire physique dans l'ordinateur : moins cher
- Opérations de traduction d'adresse (mémoire virtuelle vers mémoire physique) et d'échange de pages entre la mémoire physique et les fichiers de *swap* (échange)
-  pour avoir plus de 4 Go par processus nécessité de fonctionner en mode 64 bits et d'avoir un processeur 64 bits (SPARCv9, pas x86)
- Solaris introduit la notion de swap virtuel (SWAPFS) permettant de bénéficier entre autres d'un espace presque égal à la mémoire physique + les fichiers de swap
- Possibilité de rajouter en cours de route du swap mais aussi d'en *enlever* (changements à chaud de matériel, etc)



- Partition (sans système de fichier pour la vitesse) de swap allouée sur disque lors de l'installation d'une machine

```
/dev/dsk/c0d0s1 - - swap - no -
```

Possibilité d'un nombre arbitraire de partitions de swap dans `/etc/vfstab`.  $\rightsquigarrow$   $\nearrow$  débit en parallélisant sur plusieurs disques

- Surveillance de la mémoire

- ▶ En tout (y compris mémoire physique) avec `swap -s`

```
deauville-keryell > swap -s
total: 329232k bytes allocated + 27936k reserved = 357168k used, 3080200k available
```

- ▶ Par partition avec `swap -l`

```
deauville-keryell > swap -l
swapfile dev swaplo blocks free
/dev/dsk/c0t0d0s1 32,1 16 2307344 2017344
/dev/dsk/c0t3d0s1 32,25 16 4194272 3895568
```

- Rajout de fichiers de swap avec `swap -a fichier` qui peut être une partition brute ou un fichier standard (machine sans disque,



pas de partition disque disponible, besoin temporaire). Fichier standard créé avec `mkfile` :

```
deauville-root > mkfile 1g /export/vasque2/gros-swap
deauville-root > swap -a /export/vasque2/gros-swap
deauville-keryell > swap -s
total: 329304k bytes allocated + 28168k reserved = 357472k used, 4128408k available
deauville-keryell > swap -l
swapfile dev swaplo blocks free
/dev/dsk/c0t0d0s1 32,1 16 2307344 2017344
/dev/dsk/c0t3d0s1 32,25 16 4194272 3895568
/export/vasque2/gros-swap - 16 2097136 2097136
```

- Suppression d'une zone de swap avec `swap -d fichier`

```
deauville-root > swap -d /export/vasque2/gros-swap
```



## Fichiers temporaires en mémoire — TMPFS

- Beaucoup d'applications créent de nombreux fichiers temporaires : compilateurs, bases de données, etc
- Idée : utiliser un système de fichiers en mémoire (TMPFS) pour les répertoires stockant ces fichiers temporaires

- Par défaut `/tmp` est un TMPFS

```
swap - /tmp tmpfs - yes -
```

-  Si un TMPFS est plein c'est la mémoire virtuelle qui est pleine... Possibilité de limiter la taille d'un TMPFS avec `-o size=taille`
- Inconvénient : si la machine est éteinte le contenu du TMPFS est perdu. En accord avec la sémantique de `/tmp`
- Sinon autre possibilité pour `/var/tmp` : utiliser un CacheFS d'un `/var/tmp` sur disque dont le fichier de cache serait lui même en



TMPFS...



## Nécessité des sauvegardes

209

-  Les fichiers représentent la propriété intellectuelle. Toute la vie d'une entreprise sous forme de fichiers...
- Les pannes existent, les utilisateurs (`rm -rf *`) et les bugs aussi...
- La majorité des entreprises ne survivent pas à moyen terme à la perte de leur informatique
- Nécessité de faire des sauvegardes régulières (1 fois par jour)
- Et sur le long terme (années) pour retrouver des fichiers perdus par erreur et dont on ne s'aperçoit pas tout de suite
- Stocker les média de sauvegarde à différents endroits le plus loin possible du système de sauvegarde (autres bâtiments, villes)
- Penser à crypter les sauvegardes en milieu sensible



- Ne pas oublier que les média ne sont pas éternels ( $\approx$  10 ans pour les bandes magnétiques)
- Que faire de média dont plus aucun lecteur n'existe ? Légende disant qu'il existe un service de l'État qui possède tous les types de lecteurs possibles au cas où...



## Organisation de sauvegardes

- Choisir un système matériel : DAT (12 Go et 1 Mo/s), 8mm, AIT, DLT, DTF, SD-3, D-1, D-2 (330 Go et 15 Mo/s)
- RAIDs de bandes ( $\nearrow$  tolérance,  $\nearrow$  débit,  $\nearrow$  capacité), automates de plein de bandes
- Choisir une logiciel de sauvegarde
  - ▶ Outils bruts par partition et par machine :  
ufsdump/ufsrestore,...
  - ▶ Outils automatisant la sauvegarde d'un réseau et utilisant des outils bruts
    - *Solstice Backup*
    - Amanda ([www.amanda.org](http://www.amanda.org)) gratuit et utilisé au ENSMP/CRI & IAR2M : 80 partitions (1 centaine de Go) sauvegardées chaque nuit sur 1 cassette 8 mm de 7 Go. Au



ENSTBr/RIRE : DAT DDS4 de 20 Go pour des centaines de Go

Utiliser un outil de ce type !

<http://www.lit.enstb.org/~keryell/publications/conf/2001/JRES2001/amanda>

- Choisir une politique de sauvegarde :
  - ▶ Faire un état des lieux des machines et des disques
  - ▶ Quoi sauvegarder ? Tout si possible ! En priorité les partitions utilisateurs avant celles du système (récupérables en gros sur CD)
  - ▶ Quand sauvegarder ?
  - ▶ Durée nécessaire aux sauvegardes
  - ▶ Centralisation du système de sauvegarde
  - ▶ Débit du réseau : ne pas écrouler le réseau pendant le fonctionnement normal



- ▶ Éviter de faire des sauvegardes sur des systèmes de fichiers très actifs  $\rightsquigarrow$  sauvegardes la nuit. En théorie, devrait être en mode mono-utilisateur...
- ▶  Problème de cohérence avec des systèmes de base de donnée : sauvegarder plutôt une image instantanée de la base plutôt que la base de donnée
- ▶ Utiliser astucieusement les sauvegardes incrémentales : plutôt que de sauvegarder toute une partition, on la sauve une fois complètement et ensuite on ne sauve que les différences par rapport à la sauvegarde complète. Récursion possible sur le concept. Compromis à trouver entre sécurité, facilité de restauration et place occupée sur bande. De toute manière, besoin supérieur à l'incrément quotidien.  
Exemple à allocation statique pour une partition disque
  - Vendredi soir : niveau 0



- Lundi soir : niveau 1
- Mardi soir : niveau 2
- Mercredi soir : niveau 3
- Jeudi soir : niveau 4
- Si on perd 1 niveau 0 on perd la semaine
- Si on veut récupérer 1 jeudi il faut lire 5 bandes

Rotation d'un jour sur toutes les partitions de manière cyclique pour répartir la quantité (niveaux 0) sur les bandes de sauvegarde ~> rapidement compliqué...

~> Outils comme AMANDA



## Dériveurs de bande

- Pas d'accès direct à l'information
  - ▶ Fichiers séparés par des marques de fin de fichier
  - ▶ Possible de sauter plus rapidement d'une marque à l'autre
  - ▶ Concaténation possible de fichiers si pas de rembobinage

Utilisation de la commande `mt` pour contrôler la bande

- Conducteur matériel : `/dev/rmt/x y [n]` *Raw Magnetic Tape* (blocs de 512 octets)
  - `x` : numéro dérouleur
  - `y` : lettre de densité (l, m, h, u) plus de compression `c` optionnelle
  - `n` : optionnel indiquant qu'on ne veut pas de rembobinage automatique



- Outils de sauvegarde de base de UFS (gestion via inodes...)
  - Sauvegarde d'abord la table des matières ~→  si gros changements pendant la sauvegarde...
  - Niveaux incrémentaux de 0 à 9
  - ufsdump pour la sauvegarde d'une liste de fichiers ou 1 système de fichier. Il faut préciser la taille de la bande (même si sauvegarde dans un fichier...) car possibilité d'utiliser plusieurs bandes
    - ▶ ufsdump 0uf /dev/rmt/0 / : niveau 0
    - ▶ ufsdump 1uf deauville:/dev/rmt/0 /export/home : niveau 1 sur bande distante
- « u » a pour effet de mettre à jour /etc/dumpdate qui note la date des sauvegardes pour chaque niveau d'incrément. Ne sont



sauvegardés pour un niveau donné que les fichiers modifiés après la date des sauvegardes de niveau inférieur

- ufsrestore pour la restauration de fichiers
  - ▶ ufsrestore rf /dev/rmt/0 : récupère le contenu de la partition sur la bande
  - ▶ ufsrestore if moret.c0t1d0s7.19981201.1 : récupération en mode interactif
 

```
deauville-keryell > ufsrestore if moret.c0t1d0s7.19981201.1
ufsrestore > ls
.:
 WWW/ users/
ufsrestore > cd WWW
ufsrestore > cd conf
ufsrestore > ls
./WWW/conf:
 srm.conf
ufsrestore > add srm.conf
ufsrestore > extract
You have not read any volumes yet.
```



```
Unless you know which volume your file(s) are on you should start
with the last volume and work towards the first.
```

```
Specify next volume #: 1
set owner/mode for '.'? [yn] n
ufsrestore > quit
```

En général ne pas changer les droits/modes de « . » si  
restauration ailleurs que dans le répertoire d'origine

- Lors d'une restauration complète, il y a ré-allocation des inodes  
et il ne faut pas commencer la sauvegarde suivante par une  
incrémentale mais par une de niveau 0



## Création de snapshot

- Sauvegarde loin d'être atomique...
- ↔ Incohérences possibles dans les fichiers entre le début et la  
fin de la sauvegarde
- Problème typique avec des bases de données (format binaire  
interne opaque)
- Faire une photo instantanée du système de fichier juste avant de  
lancer la sauvegarde
- Les données ayant changé depuis l'instantané sont conservées  
dans un fichier (*backing store*)

```
fssnap -F ufs -o maxsize=5m,bs=/usr/burette.snapshot.file,unlink /export/burette
mkdir /usr/burette.snapshot
mount -F ufs -o ro /dev/fssnap/0 /usr/burette.snapshot
cd /usr/burette.snapshot
```



```
tar cvf /dev/rmt/0 .
umount /usr/burette.snapshot
fssnap -d /export/burette
```

<http://docs.sun.com:80/ab2/coll.786.2/S8ADMINSUPP/@Ab2PageView/2977?Ab2Lang=C&Ab2Enc=>



- Problème : des morceaux du noyau risquent d'être écrasés lors de la restauration...
- Démarrer sans utiliser ces disques : depuis le CD-ROM, depuis un serveur d'installation (toujours en avoir un sous le coude), en tant que *diskless* depuis le réseau ou depuis un autre disque le cas échéant
- Monter le disque à restaurer dans `/mnt` et faire la restauration dedans puis démonter
- Comme la sauvegarde ne gère pas le boot, réinstaller la zone de démarrage sur le disque /

```
installboot /usr/platform/'uname -i'/lib/fs/ufs/bootblk /dev/rdisk/ dev
```



- `ufsdump/ufsrestore` efficace pour UFS Solaris mais non portable
- Outils portables pour transporter une arborescence
- `tar` + GNU
- `cpio` plus portable et multivolume + GNU
- `pax` gère les formats POSIX `cpio` et `ustar`
- Utile pour recopier des arborescences sous Unix  

```
tar cf - ici | (cd là-bas ; tar xf -)
```

 ou un `cp -ap` de GNU
-  si sauvegarde avec des noms absolus, restauration avec des noms absolu depuis `/`. Pour rattraper la sauce : changer la racine du processus de restauration via `chroot...`



## Clonage de système de fichiers

- Capture de toute l'information au niveau de `/dev/rdisk`
- Peut être utile pour une restauration de `/` avec la zone de démarrage
- Duplication rapide de disquette  

```
volcheck
```

```
dd if=/vol/dev/aliases/floppy0 of=devcfg-2.7-disquette bs=1440k
```

```
eject
```

 puis avec une nouvelle disquette  

```
volcheck
```

```
fdformat -d -U
```

```
dd if=devcfg-2.7-disquette of=/vol/dev/aliases/floppy0 bs=1440k
```

```
eject
```
- Clonage d'une installation de Windows NT multi-OS via Solaris  

```
cd /export/verre4
```



```
rsh roosevelt "dd if=/dev/rdisk/c0d0p1 bs=512 | /usr/local/bin/gzip -9" > nt-4.0.server.roosevelt.gz
```

puis installation depuis une autre machine lors de la procédure JumpStart

```
mkdir /tmp/verre4
mount palo-alto:/export/verre4 /tmp/verre4
/usr/local/bin/gunzip -c /tmp/verre4/nt-4.0.server.roosevelt.gz | dd bs=512 of=/dev/rdisk/c0d0p1
```



## Quotas : contrôle de l'usage des disques

- Fixe des limites à l'usage de chaque disque par utilisateur
  - ▶ Limite *soft* : déclenche un avertissement et laisse un délai de correction
  - ▶ Puis limite *hard* : empêche toute écriture
- Mettre en place les quotas
  - ▶ Au niveau du répertoire racine du système de fichier à contrôler avec les droits de `root`

```
touch quotas
chmod 600 quotas
```
  - ▶ Rajouter `rq` dans l'option du système de fichier dans `/etc/vfstab` pour la demande au démarrage
  - ▶ `quotaon disque` : démarre sans attendre le prochain reboot
- `edquota utilisateur` modifie les quotas



- `quota -v [utilisateur]` affiche l'état de ses quotas
- `repquota` Affiche tous les quotas
- Arrêt des quotas avec `quotaoff`



- Partage de fichiers en réseau par machines hétérogènes de fichiers ou de hiérarchies
- Aide à rendre les machines uniformes par centralisation de l'information
- Protocole portable (Unix, Windows, VMS,...)
- Devices non partageables
- Exportation contrôlée des ressources sur la base d'une hiérarchie et d'un groupe de machine
- Les hiérarchies exportées ne peuvent pas se recouvrir
- Sous Unix exportation d'un système de fichier ou d'une partie
- Version
  - 2 : Version la plus commune



- 3 : À partir de Solaris 2.5, nécessite clients et serveurs V3
  - ▶ Autorise les écritures asynchrones sur disque (ne bloque pas le client)
  - ▶ Macro-requêtes pour diminuer le trafic
  - ▶ Vérification des droits améliorés
  - ▶ Dépasse 8 Ko/paquets
  - ▶ Support des ACL
  - ▶ NFS au dessus de TCP en plus d'UDP (pas spécifique version 3)
  - ▶ Gestion des gros fichiers (plus de 2 Go)
  - ▶ Utilisation dynamique de plusieurs serveurs en cas de panne sur des systèmes de fichier en lecture seule (/usr/local,...)
  - ▶ WebNFS RFC 2054. RFC 2225 : meilleur débit que HTTP, pas de sur-coût à la FTP



- ▶ Sécurité Kerberos V5 et RPCSEC\_GSS (API)



- À la main avec

```
share [-F nfs] [-o options] [-d description] chemin
```

Quelques *options*

- ▶ `index=file` Renvoie *file* au lieu de la liste des fichiers du répertoire lors d'un accès via URL `nfs://...`
- ▶ `nosuid` Empêche la création de programme `setuid` et `setgid`
- ▶ `public` Définit ce système de fichier comme étant celui racine consultable par un butineur avec `nfs://`
- ▶ `ro=[liste-machines]` Exporte en lecture seulement à une liste de machine ou à  tout l'univers
- ▶ `root=liste-machines` Les machines spécifiées ont les droits de  `root`
- ▶ `rw=[liste-machines]` Exporte en lecture et écriture à une



liste de machine ou à  tout l'univers

- ▶ `sec=liste-modes` système d'authentification utilisé
  - ▶ `liste-machines` consiste en des noms de machine, des `netgroup`, des suffixe DNS (à condition d'avoir `dns` en tête de `hosts` dans `/etc/nsswitch.conf`, des (sous-)réseaux. « - » est utilisé comme privatif dans les accès
- Automatiquement au démarrage en mettant les lignes précédentes dans le fichier `/etc/dfs/dfstab`. Prise en compte des modifications avec `shareall`
  - Vérifier que le serveur NFS est lancé avec `/etc/init.d/nfs.server start`  
Normalement lancé au démarrage si existence de `/etc/dfs/dfstab`



- Avec `mount -F nfs, /etc/vfstab` ou l'auto-monteur
- Syntaxe `mount [-F nfs] [options_génériques] [-o options_spécifiques] [-O ] ressource mount_point`
  - *ressource* à monter
    - ▶ `host:pathname`
    - ▶ `nfs://host[:port]/pathname`
    - ▶ Une liste de ressources séparées par des virgules : système de tolérance aux pannes (mais en lecture seulement)
  - Quelques options
    - ▶ `fg` Bloque jusqu'à ce que le montage soit fait (défaut)
    - ▶ `bg` Réessaye en tâche de fond si le montage échoue
    - ▶ `hard` Réessaye l'entrée-sortie jusqu'à ce que le serveur réponde



- ▶ `soft` Renvoie un code d'erreur si le serveur ne répond pas.  Si un programme ne teste pas les retours d'erreurs...
  - ▶ `intr` Accepte d'interrompre les un accès `hard` bloquant en tapant `~C` au clavier
  - ▶ `nointr` Le contraire
  - ▶ `public` Passe par le répertoire exporté comme étant public
  - ▶ `ro` Monte en lecture seule
  - ▶ `rw` Monte en lecture/écriture
  - ▶ `suid` Autorise l'exécution de programmes en `setuid`. Comportement par défaut...
  - ▶ `nosuid` L'inverse
- `nfsstat` permet d'avoir diverses statistiques sur le fonctionnement de NFS en fonction des options
  - `showmount` permet d'avoir des informations sur les clients ou les partitions exportées d'un serveur donné



- Trop de montages saturent le système
- ↷ Montage et démontage à la demande à partir de tables centralisées
- Tout ne peut pas être connu dès le départ
- Uniformisation des machines
- Éviter de devoir être `root` pour monter des répertoires (distants)
- Généralisation de l'espace de nommage  
*/net/machine/fichier*
- Changement de tous les montages de toutes les machines de manière centralisée
- ↷ AutoFS : probablement l'automonteur d'Unix le plus avancé



- Une *carte* peut être un fichier */etc/carte* ou une *carte* NIS, NIS+, FNS en fonction de */etc/nsswitch.conf*
- Master map : */etc/auto\_master* associe des répertoires avec des cartes

```
keryell@voltaire ~: more /etc/auto_master
Master map for automounter
#
+auto.master
keryell@voltaire ~: ypcat -k auto.master
/home auto.home -hard,bg,intr
/xfn -xfn
/net -hosts -nosuid,nobrowse
/- auto.direct
```

- ▶ `-hosts` signifie de monter les fichiers (exportés !) de



n'importe quelle machine demandée dans le répertoire

- ▶ -xfr signifie de monter les ressources spécifiées par leur nom FNS
- ▶ /- évite d'associer une carte avec un répertoire
- Direct map : dirige AutoFS directement vers des systèmes de fichiers à partir d'un nom de répertoire

```
keryell@voltaire ~: ypcat -k auto.direct
/var/mail -rw,hard,intr,actimeo=0 palo-alto:/export/verre2/var/mail
/users -rw,hard,intr palo-alto:/export/verre1
```

- Indirect map : monte des systèmes de fichier à partir d'une clé

```
keryell@voltaire ~: ypcat -k auto.home
palo-alto / &:/export /verre1 &:/export/verre1 \
/verre2 &:/export/verre2 /verre3 &:/export/verre3 \
/verre4 &:/export/verre4
```



- ▶ & rappelle le nom de la clé
- ▶ \* accepte n'importe quelle clé
- \* &:/export
- Possibilité d'utiliser des variables pour changer localement des montages : \$ARCH, \$CPU, \$HOST, \$OSNAME, \$OSREL, \$OSVERS
- Possibilité de mettre des poids dans des montages redondants
-  Ne pas mettre au même endroit exportation et montage local mais plutôt par exemple /export/... et /home/.... AutoFS utilise directement un LOFS
- automount contrôle le comportement d'automountd (relecture d'auto\_master, changement du temps de rafraichissement)



- Développement des
  - ▶ Disquettes
  - ▶ Cartes PCMCIA (PC-card)
  - ▶ CD-ROMs surtout
- Monter un système de fichier sous Unix est *a priori* compliqué : insérer le médium, passer super-utilisateur, créer un point de montage, monter le contrôleur du médium sur le point de montage, quitter le mode super-utilisateur. Idem pour enlever le médium.
- Pénible et nécessité d'être super-utilisateur
- ↔ outil automatique : *Volume Manager*



- CD-ROM
  - ▶ Insérer
  - ▶ Patienter
  - ▶ Accéder au système de fichiers dans `/cdrom/cdrom0`,  
`/cdrom/cdrom1` (second lecteur),...
  - ▶ Accéder aux données brutes dans  
`/vol/dev/aliases/cdrom0`,...
- Disquette
  - ▶ Insérer
  - ▶ Taper `volcheck` (pas de détection d'insertion)
  - ▶ Accéder au système de fichiers dans `/floppy/floppy0`,  
`/floppy/floppy1` (second lecteur),...
  - ▶ Accéder aux données brutes dans



```
/vol/dev/aliases/floppy0,...
```

- Possibilité d'en faire bénéficier d'autres machines en exportant via NFS le système de fichier comme un autre en rajoutant `share cdrom*` dans `/etc/rmmount.conf` pour que ce soit automatique



## Éjection médium amovible

- Vérifier que plus aucun processus n'utilise le médium
- `eject cdrom` ou `eject floppy` (même si cela ne peut pas éjecter physiquement la disquette, cela en avertit le système)
-  Si médium encore utilisé, impossible d'éjecter... Trouver la liste des processus l'utilisant avec `fuser -u /cdrom/cdrom0` par exemple et employer des mesures correctives



Peut stocker de 360 Ko à 2,88 Mo de données

- Format MS-DOS et NEC-DOS
  - ▶ Formattage physique  
`fdformat -v -U + options de densité, nom,...`

- Format UFS
  - ▶ SPARC
  - ▶ Intel

Incompatibilités liées au stockage grand indien/petit indien des octets

- ▶ Formattage physique  
`fdformat -v -U + options de densité, nom,...`
- ▶ Rajout d'un système de fichier UFS  
`newfs -v /vol/dev/aliases/floppy0`



- Avertir le volume manager qu'il peut monter la disquette dans  
`/floppy`  
`volrmmount -i floppy0`



- Système de *spool* permettant d'entasser les impressions en différé
- Imprimantes locales : impressions sont mises en attente de libération de l'imprimante dans une zone de spool  
/var/spool/lp
- Imprimantes distantes : mise en attente dans /var/spool/lp et envoi par protocole BSD ou TCP *pass-through* sur le serveur de l'imprimante distante, voire l'imprimante elle-même (mais implémentation parfois fantaisiste des protocoles réseau...)
- ↪ Penser à avoir de gros /var/spool/lp...
- Installation par outils graphiques (solstice AdminSuite ou admintool) ou textuels (lpadmin)
- ∃ Alternative plus moderne libre : CUPS



## Rajouter une imprimante — client

Base de donnée dans

- \$HOME/.printers
  - /etc/printers.conf
- ```
# The default printer:
_default:\
    :use=lw-d3-etage

# All the printers:
_all:\
    :all=copieur-d3-rdc,lw-d109,lw-d3-etage,lw-rdc,lw-rdc-hall,DJ1120C,xeroxcol

lw-d3-etage:\
    :bsdaddr=lw-d3-etage.priv.enst-bretagne.fr,lp:\
    :description=Imprimante HP double face de l'étage:
modifiable aussi par outils graphiques et lpadmin)
```



- Ressource NIS `printers.conf.byname`
- Ressource NIS+ et FNS
- Accès direct via
`lpr -P serveur:imprimante fichier`

Choix de l'imprimante par défaut

- `$PRINTER` et `$LPDEST`
- Recherche d'une imprimante `_default` dans les fichiers précédents



Contrôle imprimante

- `lpstat -p imprimante` donne des informations sur l'imprimante
- `accept imprimante` accepte les demandes d'impression
- `reject -r une-raison imprimante` n'accepte plus les demandes d'impression
- `enable imprimante` permet l'impression
- `disable -r une-raison imprimante` arrête l'impression
- `cancel -u user | request-id-list | imprimante` annule des travaux d'impression
- `lpmove impr-src impr-dst` transfère les impressions depuis une imprimante vers une autre en faisant un `reject impr-src`
- `lpadmin -A` contrôle la marche à suivre en cas d'alerte



- `lpadmin -u allow:user-list` autorise des utilisateurs
- `lpadmin -u deny:user-list` interdit à des utilisateurs
- `/usr/lib/lp/lpshut` arrête le système d'impression
- `/usr/lib/lp/lpsched` démarre le système d'impression
- Messages de log dans `/var/lp/logs/`



- Besoin de lancer des tâches à un instant précis : sauvegardes, nettoyage nocturne, miroir de site WWW,...
- Géré par le système `cron`
- Le résultat de la tâche (`stdout` et `stderr`) est renvoyé par mail
- Répétitions : `crontab`

► Format fichier

*minutes secondes jour-du-mois mois
jour-de-la-semaine commande*

« * » pour ignorer un champ

Commande exécutée par un shell (`sh` par défaut).  Un

« % » représente un caractère fin de ligne

► `crontab -l [utilisateur]` Affiche la liste des travaux

```
22 03 * * * crontab -l > $HOME/bib/crontab.'hostname'
25 01 * * * rsh deauville $HOME/bin/gen_calendar
33 03 * * * cp /var/spool/mail/$USER $HOME/lettres/arrivee/mail.{'usr/bin/date +%w'
```



- ▶ `crontab -e` pour éditer le fichier
- ▶ Contrôle des autorisations par utilisateur via `/etc/cron.d/cron.allow` et `/etc/cron.d/cron.deny`
- Une seule fois : `at`
 - ▶ `at -m heure-date` demande un script et envoie un mail après exécution
 - ▶ `at -l` affiche ses travaux en attente
 - ▶ `atq` affiche les travaux en attente
 - ▶ `at -r` efface une demande de tâche
 - ▶ Variante `batch` : empile la tâche sur la pile des travaux à exécuter
 - ▶ Contrôle des autorisations par utilisateur via `/etc/cron.d/at.allow` et `/etc/cron.d/at.deny`



- Administrer un domaine : faire des actions à distance
- `rlogin machine` permet de se connecter sur une machine distante. Aucun mot de passe demandé si autorisé par fichier (distant...) `$HOME/.rhost` ou `/etc/hosts.equiv`
- `rsh machine commande` exécute une commande à distance dans son `$HOME`. Autorisation par les mêmes fichiers que `rlogin`
- `rcp [-rp] [machine-src:]fichiers [machine-dst:]rep` utilise le protocole `rsh` pour faire un `cp` de fichiers. Méta-caractères acceptés mais les protéger pour action à distance
-  Ne pas laisser des `/etc/hosts.equiv` ou `.rhost` trop permissifs. Vérifier leur contenu
- Récupérer une copie d'écran à distance (`xwd -root` si X11)



-  Toutes ces commandes sont peu sécurisées ~> utiliser plutôt ssh et scp
- ping *machine* teste si les paquets réseau font bien l'aller-retour
- rusers [-l] demande et affiche la liste des utilisateurs connectés aux machines du réseau
- rup demande et affiche la charge des machines du réseau
- Ouvrir plein de `perfmeter` graphiques



- *File Transfer Protocol*
- ftp [-i] *machine* ouvre une connexion
- Le nom de login `anonymous` signifie une connexion anonyme (on donne son adresse de mail comme mot de passe par convention) si un compte anonyme existe
- Commandes classiques `ls`, `cd`, `mkdir`
- `lcd` change de répertoire localement
- `get/put` pour transférer un fichier
- `mget/mput` pour transférer une liste de fichier (sans demander confirmation si `ftp -i`)
- `bin` demande) des transferts en binaire (pas de traduction de format de fin de ligne dans fichiers textuels, etc)



- quit ou bye pour arrêter
- Emballé maintenant dans les URL `ftp://machine/` (anonymous)
- Protocole vieux et compliqué (ports dynamiques, pas terrible pour les pare-feu,...)
-  Toutes ces commandes sont peu sécurisées ~> utiliser plutôt `sftp` ou SCP d'OpenSSH



- Problème des protocoles comme `telnet` ou `rlogin` :
 - ▶  Font confiance aux traductions IP ↔ noms ou font passer les mots de passe en clair sur le réseau... ☹
 - ▶  Une connexion peut être détournée en cours de route (interception/injection, changement des tables de routage,...) : l'authentification sécurisée (OTP) ne suffit pas
- ~> Besoin logiciel sécurisé par chiffrement de type
 - ▶ Connexion à distance (style `rlogin`) : `ssh`
 - ▶ Exécution de commande à distance (style `rsh`) : `ssh`
 - ▶ Copie de fichier entre machines style `rcp` : `scp` ou style `ftp` : `sftp`
- Généalogie
 - ▶ Entreprise finlandaise : version 1. Protocole peu sécurisé



(taille des paquets non chiffrée, somme de vérification non chiffrée,...). Utilisation non commerciable libre

- ▶ Version 2 plus sécurisée. Utilisation non commerciable libre
- ▶ OpenSSH sous produit libre d'OpenBSD ; version 1 et 2 du protocole
- Authentification forte
 - ▶ RSA ou autres. Clés publiques des autres dans son `~/.ssh/authorized_keys`
 - Serveur génère un nombre aléatoire de 256 bits
 - Chiffré par serveur avec la clé publique du client demandant la connexion
 - Client déchiffre le nombre aléatoire avec sa clé secrète et renvoie son hachage MD5 (pour éviter une attaque de RSA à texte connu)



- Serveur calcule aussi le hachage MD5 et le compare à celui reçu
- ▶ `.rhosts` et `/etc/hosts.equiv` basée sur adresses IP (comme `rlogin`,...) mais avec protection par clé RSA par machine (`/etc/ssh_known_host` et `~/.ssh/known_hosts`) pour éviter les attaques IP et reroutage et une partie des mensonges de DNS
- ▶ Mélange des 2
- Confidentialité : toutes communications chiffrées automatiquement
 - ▶ Utilisation de RSA pour échanger les clés de l'algorithme symétrique
 - ▶ Algorithmes symétriques disponibles : IDEA, Blowfish, Triple-DES



- ▶ Authentification démarrée après le chiffrement : pas de mots de passe en clair sur le réseau même si pas d'authentification forte
- ▶ Possibilité de protéger clé secrète par une phrase secrète hachée par MD5 pour déchiffrer la clé via 3DES. Sinon : `root` local peut voler trivialement la clé d'un utilisateur local pour connexion à distance
- Encapsulation chiffrée du protocole X11 et gestion automatique Xauthority & `$DISPLAY`
- Redirection de n'importe quel port TCP/IP (transaction commerciale et monétaire, accès Intranet, serveur de mail, de News,...)
- Pas de confiance *a priori* au réseau
- Remplace les commandes `rlogin`, `rsh` (`ssh2`), `rcp` (`scp2`) et `ftp`



(`sftp2`)

- Éventuelle compression des données (marche mieux *avant* le chiffrement ☺)
- Couplage possible avec des caulettes d'authentification et S/Key
- Compatibilité avec l'authentification pare-feu TIS
-  Ne pas oublier de supprimer l'usage de `rlogin`, `rsh`,...
- Essaye d'être facile à utiliser pour ne pas dégoûter de la sécurité !
- Distributions
 - ▶ L'original : <http://www.ssh.fi>, <ftp://ftp.cs.hut.fi/pub/ssh>
 - ▶ `ssh2` gratuit en utilisation non commerciale
 - ▶ `1sh` aussi v2 GNU en cours de développement



- ▶ Version libre OpenSSH basée sur `ssh1.27` mais aussi v2
www.openssh.org
- ▶ Introduction : <http://www-lns.mit.edu/compfac/ssh.html>



Utilisation de ssh

- `ssh`, `slogin`, `scp` : commandes de base. Peuvent être installées sous les noms de `rsh`, `rlogin` et `rcp`
 - ▶ Utilise les clés publiques RSA des machines distantes de `/etc/ssh_known_hosts` ou `~/.ssh/known_hosts` pour vérifier que la machine cible est bien la bonne
 - ▶ `~/.ssh/identity` contient sa clé secrète RSA et `~/.ssh/identity.pub` la clé publique correspondante
 - ▶ `~/.ssh/identity.pub` doit être présent dans le `/.ssh/authorized_keys` distant pour autorisation
- `sshd` serveur à lancer en attente de connexion
 - ▶ `/etc/ssh_host_key` contient la clé secrète du serveur créée à l'installation
 - ▶ `/etc/ssh_host_key.pub` contient la clé publique du serveur



créée à l'installation. Récupérée par `make-ssh-known-hosts` pour permettre une authentification à la connexion

- ▶ `/etc/ssh_known_hosts` et `~/.ssh/known_hosts` permettent l'autorisation par machine via mécanisme `rhosts`
- ▶ `~/.ssh/authorized_keys` contient les clés publiques RSA pour se connecter chez un utilisateur
- `ssh-keygen` crée sa double clé RSA personnelle protégée par une phrase. Stockage d'un commentaire pour aider la mémoire
- `make-ssh-known-hosts` interroge le DNS d'un domaine pour construire la liste des machines. Interroge ensuite tous les serveurs `ssh` pour récupérer leur clé publique et construit le fichier `/etc/ssh_known_host`
- Attention : comme la confiance est basée aussi sur les clés publiques, être sûr qu'on a les bonnes clés publiques. Problème



de démarrage du processus...



Accéder à l'intranet de l'ENSMP depuis l'ENST Bretagne :

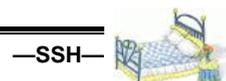
- ~/.ssh2/ssh2_config:


```

*:
    VerboseMode    yes
    Compression    yes
    KeepAlive      no
# Intranet des Mines :
    LocalForward   "7777:news.ensmp.fr:119"
    LocalForward   "8080:fontainebleau.ensmp.fr:80"
      
```
- Accéder aux News et Forum sous Emacs : ~/.gnus.el


```

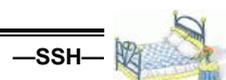
(setq
 ;; Plus de connexion directe aux Mines :
 gnus-nntp-server nil
 ;; Mes serveurs
      
```



```

rk-serveur-news-enstbr '(nntp "news.enst-bretagne.fr")
rk-serveur-news-mines '(nntp
    "Mines"
    (nntp-address "localhost.enst-bretagne.fr")
    ;; Tunnel ssh :
    (nntp-port-number 7777)
    )
rk-serveur-forum-enstb '(nntp
    "Forum ENST Bretagne"
    (nntp-address "melimelo.enst-bretagne.fr")
    (nntp-port-number 7777)
    )

;; Où lis-je :
gnus-select-method rk-serveur-news-enstbr
gnus-secondary-select-methods (list
    rk-serveur-news-mines
      
```



```
rk-serveur-forum-enstb)
;; Différents serveurs pour poster les News :
gnus-post-method (list
  rk-serveur-news-enstbr
  rk-serveur-news-mines
  rk-serveur-forum-enstb
)
```



Détection des changements et mise à jour

267

- Avoir des copies pour comparaison
 - ▶ Nécessite une capacité double
 - ▶ Problèmes de licence interdisant la copie de certains logiciels...
 - ▶ Permet une remise à jour facile du système après compromission
 - ▶ Copies locales (cachées sur un disque et/ou cryptées) ou distantes (via NFS en lecture seulement)
 - ▶  Les commandes de comparaison peuvent être compromises...
 - ▶ Automatisation des comparaisons et des installations depuis des maîtres : rdist, rsync, cfengine
- Méta-données : liste des fichiers et répertoires, leur droits et leur



dates de modification

- ▶ Prend moins de place et moins lourd que tout en double
- ▶  Dates peuvent être modifiées

- Signatures

- ▶ Utilisation de fonctions de hachage cryptographiques
- ▶ Comparaison des hachages avec une base
- ▶ Outil Tripwire : permet de vérifier contenu et/ou droit, dates, etc. configurable par fichier



Audit et journaux de fonctionnement

- Une fois système en place, nécessité de garder trace de l'activité et des problèmes
- Faire confiance mais « vérifier tout de même »...
- Permet de trouver trace de
 - ▶ Bug
 - ▶ Intrusion
 - ▶ Dommages causés
- Utile pour reconstruire le système, justice, compagnies d'assurance,...
-  Les journaux eux-mêmes peuvent être compromis
- Utilisation d'une vieille machine pour stocker les informations depuis une liaison série avec protocole minimal



- Fichiers généralement contenus dans `/var/adm` et `/var/log`
 - ▶ `access_log` indique les fichiers accédés par HTTP
 - ▶ `aculog` contient les numéros appelés par modems
 - ▶ `lastlog` contient la date de dernière connexion et éventuellement de dernier échec
 - ▶ `loginlog` stocke les échecs de connexion
 - ▶ `messages` stocke les messages systèmes envoyés à la console par `syslog`
 - ▶ `pacct` enregistre les commandes lancées par tous les utilisateurs. Visualisé via `lastcomm`. Démarrage de l'enregistrement des commandes par `/usr/lib/acct/startup` référencé par `/etc/init.d/acct start`
 - ▶ `suolog` contient les usages de la commande `su`



- ▶ `utmp` possède la liste des utilisateurs connectés (utilisé par `w`)
 - ▶ `utmpx` version étendu d'`utmp` (contient d'où on est connecté,...) utilisé par `finger`, `who`
 - ▶ `vold.log` informations du *volume manager*
 - ▶ `wtmp` stocke les dates de connexion et de déconnexion ainsi que de reboot
 - ▶ `wtmpx` version étendue à la `utmpx`. Usage par `last`
 - ▶ `xferlog` contient les accès FTP
- Les connexions par un processus de login sont enregistrées.  Pas les exécutions de commandes à distance via `rsh`
 - Les fichiers de log peuvent devenir très gros ~> refus de service
 - Penser à allouer suffisamment de place pour les journaux
 - ~> Scripts de nettoyage et de résumés statistiques dans la



crontab

-  Ne pas effacer simplement un fichier de log mais plutôt

```
cp /dev/null pacct
```

par exemple *file descriptors* ouverts possibles (noyau, syslog, ...)



- Centralisation du système de journalisation des messages d'information
- Permet de modifier le type de journalisation sans avoir à modifier toutes les applications
- Enregistrement
 - ▶ Nom du programme
 - ▶ Type (noyau, utilisateur, mail, autorisation,...)
 - ▶ Importance (critique, urgence, alerte, information, debug,...)
 - ▶ Message
- Configuration par fichier `/etc/syslog.conf` (attention à l'importance des tabulations)
- Possibilité d'envoyer



- ▶ Dans un fichier
- ▶ Sur la console
- ▶ Sur une imprimante, une liaison série (bastion)
- ▶ Au `syslog` d'une autre machine ou plusieurs. Par défaut envoie à la machine `loghost` si elle existe
- Commande `logger` pour créer des messages `syslog` depuis un *shell*
-  Peut générer de faux messages `syslog`, avec des caractères spéciaux,...
- `syslog` utilisé par d'autres systèmes qu'Unix pour centraliser de l'information (routeurs,...)
-  Attaques de refus de service sur `syslog` (UDP port 514)...
-  UDP peu sûr



- Facilement submergé par tous les journaux de log
- \rightsquigarrow Outil libre écrit en `perl`
- Extrait les informations anormales par expressions régulières à la `perl`
- Résumés possibles par intervalle de temps (tel message a été vu tant de fois)
- Actions configurables



- Si compte corrompu, chercher dans les fichiers d'historique de commande (`~/.history`)
- Ces fichiers peuvent être détruits ou faux...
- Rajouter avant un lien *hard* sur les `.history` ou essayer carrément les *pipe* nommés
- Des configurations stockant le courriel envoyé dans des fichiers
- Fichier d'autorisation de connexion (`.rhosts`, `.netrc`)
- En arrêtant le système on peut geler l'information en cours de piratage
- Aller voir dans les blocs libres du disque démonté s'il reste des traces de fichiers effacés
-  Éthique sur la confidentialité de la vie privée !



- Voir les travaux pratiques...
- Visualisation des processus : `ps auxww` (BSD) ou `ps -ef` (SVR4), `top`
- Tracer les appels systèmes et appels de fonction d'un processus : `truss -f -p pid` ou `strace`
Pratique pour voir quels sont les fichiers cherchés par une application
- Debugger un processus : `gdb -p pid`
- Avoir des infos sur un processus : regarder dans `/proc/pid`
- Regarder des paquets sur le réseau : `snoop` ou `tcpdump`
- Associer des fichiers ou sockets à des processus : `fuser`
- Lister les connexions réseaux, des statistiques,... : `netstat`



<http://docs.sun.com/?p=/doc/806-5205>

- Démarrage d'un Unix simplifié depuis
 - ▶ Réseau
 - ▶ DVD
 - ▶ CD-ROM
 - ▶ Disque
 - ▶ Bande

Besoin d'une disquette de démarrage sur PC pour récupérer un minimum de drivers

- Installation de type
 - ▶ WebStart : à partir d'un butineur WWW pour installation assez standard
 - ▶ Installation interactive : permet de faire une installation plus



précise

- ▶ `suninstall` en mode texte
- ▶ Installation personnalisée JumpStart : automatique !
- ▶ Web Start Flash
- ▶ Live upgrade : installation d'une nouvelle version dans une partition inutilisée et redémarrage sur celle-ci. Retour arrière trivial sur l'ancien système possible
- Possibilité d'installation depuis rien ou mise à niveau version Solaris (*upgrade*)
- Peut préserver le contenu des disques utilisateurs. Ne dispense pas des sauvegardes...
- Intérêt installation JumpStart par réseau : tout automatique sans avoir même à se déplacer. Sur Sun :

```
ssh machine 'reboot "net - install"'
```



- Pour la mise au point de la méthode un ordinateur portable avec réseau sans fil aide énormément !



Installation manuelle avec Web Start

- Au choix
 - ▶ `boot cdrom` depuis le DVD/CD Solaris 9 Installation CD
 - ▶ `boot net` depuis un serveur d'installation correctement configuré
- Installation graphique (ou pas avec l'option - `nowin`)
- Lance un navigateur WWW
- Répondre aux questions
- Possibilité de faire une installation initiale ou une mise à niveau



- Installation textuelle (console)
- Au choix
 - ▶ boot cdrom depuis le DVD/CD Solaris 9 Software 1
 - ▶ boot net depuis un serveur d'installation
- Lance un serveur X et t_{wm} ou en texte brut
- Répondre aux questions
- Possibilité de faire une installation initiale ou une mise à niveau



Création d'un serveur d'installation

- Contient le système et de quoi faire démarrer les machines via le réseau
- Création dans /export/calice2/Solaris après insertion du CD-ROM Solaris 9 Software 1

```
mkdir -p /home/gavotte/calice2/Solaris
setenv INSTALL_DIR /home/gavotte/calice2/Solaris/Solaris-9
setenv JUMPSTART /usr/local/share/conf/Solaris/jumpstart
cd /cdrom/sol_9_sparc/s0/Solaris_9/Tools
./setup_install_server $INSTALL_DIR
eject cdrom
```

Compléter avec le CD Solaris 9 Software 2

```
cd /cdrom/sol_9_sparc_2/Solaris_9/Tools
./add_to_install_server $INSTALL_DIR
eject cdrom
```



- Si on veut en plus faire du WebStart par le réseau, rajouter le CD Solaris 9 Installation

```
cd /cdrom/multi_icd_sol_9_sparc_cs/s0
./modify_install_server -p $INSTALL_DIR \
    /cdrom/multi_icd_sol_9_sparc_cs/s1
eject cdrom
```

Hum... Ne marche pas avec moi ☹

- Rajoute le support linguistique avec le CD Solaris 9 Languages

```
cd /cdrom/sol_9_lang_sparc/Tools
./add_to_install_server $INSTALL_DIR
eject cdrom
```

- Crée un répertoire de description des machines pour JumpStart

```
mkdir /usr/local/share/conf/Solaris/jumpstart
# Copie des fichiers d'exemples :
```



```
cp -pr /export/calice2/Solaris/Solaris-9/Solaris_9/Misc/jumpstart_sample \
    /usr/local/share/conf/Solaris/jumpstart
```



- Télécharger les logiciels depuis sun.com : 1836577 Ko

```
setenv CD /home/gavotte/calice2/Solaris/Solaris-8-7_01/CD
setenv INSTALL_DIR /home/gavotte/calice2/Solaris/Solaris-8-7_01/install_server
setenv JUMPSTART /usr/local/share/conf/SunOS5/jumpstart
mkdir -p $CD
Download Solaris 8 7/01 Installation CD ( 305.38 MB )
Download Solaris 8 7/01 Software 1 of 2 CD ( 359.19 MB )
Download Solaris 8 7/01 Software 2 of 2 CD ( 158.52 MB )
Download Labels for first 2 CDs ( .04 MB )
Download Labels for second pair of CDs ( .05 MB )
Download Labels for first 2 CDs (to print on A4) ( .04 MB )
Download Labels for second pair of CDs (to print on A4) ( .05 MB )
Download Solaris 8 7/01 Languages CD (optional) ( 390.14 MB )
Download Solaris 8 7/01 Documentation CD, European(optional) ( 380.00 MB )
Download Solaris 8 7/01 Documentation CD, Asian (optional) ( 286.12 MB )
Download Labels for Documentation CDs (optional) ( .04 MB )
```



Serveur d'installation depuis images réseau sans média287

Download Labels for Documentation CDs (to print on A4) (.04 MB)

Télécharger

<http://www.sun.com/software/solaris/binaries/download.html>

dans \$CD

- Décompactage et création du serveur

```
unzip sol-8-u5-sparc-v1.zip
lofiadm -a $CD/sol-8-u5-sparc-v1.iso
mount -F hsfs -o ro /dev/lofi/1 /mnt
cd /mnt/Solaris_8/Tools/
./setup_install_server $INSTALL_DIR
cd $CD
umount /mnt
lofiadm -d /dev/lofi/1
rm sol-8-u5-sparc-v1.iso

unzip sol-8-u5-sparc-v2.zip
lofiadm -a $CD/sol-8-u5-sparc-v2.iso
mount -F hsfs -o ro /dev/lofi/1 /mnt
cd /mnt/Solaris_8/Tools/
./add_to_install_server $INSTALL_DIR
cd $CD

unzip sol-8-u5-lang-sparc.zip
lofiadm -a $CD/sol-8-u5-lang-sparc.iso
mount -F hsfs -o ro /dev/lofi/1 /mnt
cd /mnt/Tools/
./add_to_install_server $INSTALL_DIR
cd $CD
```



```
umount /mnt | rm sol-8-u5-lang-sparc.iso
lofiadm -d /dev/lofi/1
```

- Construction du répertoire JumpStart

```
setenv CD /home/gavotte/calice2/Solaris/Solaris-8-7_01/CD
setenv INSTALL_DIR /home/gavotte/calice2/Solaris/Solaris-8-7_01/install_server
setenv JUMPSTART /usr/local/share/conf/SunOS5/jumpstart
cd $INSTALL_DIR/Solaris_8/Misc/
# Think to preserve any existing $JUMPSTART/rules before...
mkdir -p $JUMPSTART
cp -pr jumpstart_sample/* $JUMPSTART

http://docs.sun.com/?p=/doc/806-5205/6je7vd5tb&a=view
```



Les forces en présence :

- Ordinateur à installer
- Système d'exploitation sur média ou serveur d'installation
- Un fichier de préconfiguration sur média ou serveur d'installation
- Des règles d'installation sur média ou serveur d'installation
- Un serveur RARP pour donner l'identité à la machine
- Un serveur bootparams pour donner les paramètres de démarrage

Pour des raisons de confort tout peut-être mis sur le réseau



- Mise en place d'un service d'installation

```
cd repertoire-de-solaris/Solaris_9/Tools
./add_install_client -c serveur-jumpstart:repertoire-de-jumpstart
-s serveur-solaris:repertoire-de-solaris
-p serveur-préconfiguration:repertoire-préconfiguration
-e adresse-ethernet nom arch
```

- Configure automatiquement sur le serveur :

- ▶ /etc/dfs/dfstab
- ▶ /etc/ethers
- ▶ /etc/bootparams

- Autres options pour déclarer un service de nommage, utiliser DHCP, préciser une adresse IP...



- sysidcfg
- Contrairement à ce que dit le man c'est le répertoire contenant se fichier qu'il faut préciser... ☺
- Permet de spécifier entre autres
 - ▶ Système et paramètre de nommage
 - ▶ IPv6
 - ▶ Mot de passe de root  visible chiffré dans ce fichier...
 - ▶ Langage
 - ▶ Fuseau horaire
 - ▶ Serveur de temps



- Permet de choisir profil d'installation et scripts de pré- et post-installation en fonction des caractéristiques de la machine
- Fichier `rules.ok` dans le répertoire JumpStart généré à partir de `rules` avec `check` du répertoire
- Structure de `rules` en lignes :

prédicat pré-script profil post-script



- Précise les disques et leur formatage
- Type d'installation (initiale ou mise à jour)
- Paquets logiciels à installer
- Langue



- Résultats dans `/var/sadm/system/logs`
- Fichiers mal exportés
- `root` de la machine pas `root` sur le serveur via NFS
- Un autre serveur (vieux,...) qui répond de mauvaises informations
- Tracer le trafic réseau avec `snoop/tcpdump/ethereal`
- Pendant l'installation on a Unix qui fonctionne : possibilité d'avoir une fenêtre de shell



Création d'un profil client

- Pré-configuration du système au LIT via `$JUMPSTART/LIT/sysidcfg` :

```
# $Header: /usr/local/share/conf/Solaris/jumpstart/LIT/RCS/sysidcfg,v
# 1.3 2002/09/23 10:20:14 keryell Exp $

name_service=NONE
#name_service=DNS {domain_name=enst-bretagne.fr
# name_server=192.44.75.10,192.108.115.2,192.44.77.1
# search=enst-bretagne.fr}
network_interface=PRIMARY {
                                netmask=255.255.255.0
                                default_route=192.44.75.1
                                protocol_ipv6=yes
                                }
security_policy=NONE
```



```

root_password=RXDXVgPB7xEsA
system_locale=fr
#system_locale=en_US
terminal=sun
timezone=MET
# Where to pick the time at install time :
#timeserver=192.44.75.8
timeserver=localhost

http://docs.sun.com/?p=/doc/806-5205/6je7vd5r6&a=view

```

- Création d'un profil JumpStart pour la machine rodomouls

```
$JUMPSTART/LIT/profile/rodomouls :
```

```

# $Header:
# /usr/local/share/conf/Solaris/jumpstart/LIT/profile/RCS/rodomouls,v
# 1.4 2002/09/23 10:16:08 keryell Exp $

```

```
# Install from scratch:
```



```

install_type initial_install

# All the system software :
cluster SUNWCXall

# Disk layout:

# I want the swap at the beginning for speed (cylinder 0)
# (be careful, the size is in this case in cylinder that are 760 KB
# here for <SUN2.1G cyl 2733 alt 2 hd 19 sec 80>):
# 1GB :
fileys c0t0d0s0 0:1347 swap
fileys c0t0d0s3 free /
fileys c0t2d0s0 free /usr

# All the other disks are data:
## For initial install, format the disk:

```



```
#filesystems c0t1d0s0 all /export/burette
##filesystems c0t2d0s0 all /export/dinette
#filesystems c0t3d0s0 all /export/pipette
# or choose to ignore them if there are useful data on them:
dontuse c0t1d0
##dontuse c0t2d0
dontuse c0t3d0

# All the remote file systems are auto-mounted, so that's all.

# We are in France with Euro :
locale fr_FR.ISO8859-15

# But think about other people...
# Not enough room on disk for all... :- (
#geo N_Africa
```



```
#geo C_America
#geo N_America
#geo S_America
#geo Asia
#geo Ausi
#geo C_Europe
#geo E_Europe
#geo N_Europe
geo S_Europe
geo W_Europe
#geo M_East

http://docs.sun.com/?p=/doc/806-5205/6je7vd5tn&a=view
```

- Sélection du profil par des règles \$JUMPSTART/rules :

```
hostname rodomouls.enst-bretagne.fr - LIT/profile/rodomouls LIT/finish_install
network 192.44.75.0 && disksize c0t0d0 1800-2200 - LIT/profile/disk_2GB LIT/finis
hostname sample_host - host_class set_root_pw
```



```

network 924.222.43.0 && \
    karch sun4c - net924_sun4c -
arch sparc && \
    disksize c0t3d0 400-600 && \
    installed c0t3d0s0 solaris_2.1 - upgrade -
arch i386 x86-begin x86-class -
any - - any_machine -

http://docs.sun.com/?p=/doc/806-5205/6je7vd5tl&a=view
Validation et compilation via :

cd $JUMPSTART
./check

http://docs.sun.com/?p=/doc/806-5205/6je7vd5tp&a=view

```

- Création d'un script de fignoloage
\$JUMPSTART/LIT/finish_install
#! /bin/sh -vx



```

set -v -x

# At the end of the install, the future / is indeed in /a,
# and / is only temporary during the installation.
UsrLocal=/usr/local
TempUsrLocal=/a$UsrLocal

# Since there is no running automount yet, no NIS, no DNS,...:
/bin/mkdir -p $TempUsrLocal
/usr/sbin/mount -F nfs 192.44.75.87:/export/calice1/local $TempUsrLocal

# Disable the autoshutdown:
/usr/bin/touch /a/noautoshutdown

# Make cfengine believe it is really in / instead of /a:
/usr/sbin/chroot /a $UsrLocal/sbin/cfagent -v \

```



```

-DInstallationTime -f $UsrLocal/share/cfengine/cfengine.conf

# Clean up the mounting point since it will be
# recreated by autofs anyway:
/usr/sbin/umount $TempUsrLocal
rmdir $TempUsrLocal
\end{verbatim}
\item Configuration de l'infrastructure de démarrage réseau
\begin{exemple}
setenv INSTALL_DIR /home/gavotte/calice2/Solaris/Solaris-9
setenv JUMPSTART /usr/local/share/conf/Solaris/jumpstart
cd $INSTALL_DIR/Solaris_9/Tools
./add_install_client -c gavotte:$JUMPSTART \
    -s gavotte:$INSTALL_DIR -p gavotte:$JUMPSTART/LIT \
    -e 08:00:20:89:36:84 rodomouls.enst-bretagne.fr sun4u

http://docs.sun.com/?p=/doc/806-5205/6je7vd5rq&a=view

```



- Démarrage de l'installation sur le client depuis un vieux Solaris :

```

# To enable later boot if the security is on:
eeprom security-mode=none
reboot 'net - install'

```

ou si on est devant la machine sous moniteur OpenPROM :

```
boot net - install
```

ou plus rapide :

```
boot net - install nowin
```

Résultats dans /var/sadm/system/logs

```

http://docs.sun.com/?p=/doc/806-5205/6je7vd5u9&a=view

```



- Serveurs en présences :

```
setenv INSTALL_DIR /opt/Solaris-9
setenv JUMPSTART /home/jumpstart
cd $INSTALL_DIR/Solaris_9/Tools
./add_install_client -c grace:$JUMPSTART
-s grace:$INSTALL_DIR -p grace:$JUMPSTART/Ultra5
-e 08:00:20:e7:69:ac fresnel.enst-bretagne.fr sun4u
```

- Règles

```
karch sun4u && model SUNW,Ultra-5_10 && disksize c0t0d0 2500-11000 Fprom Finsta
karch sun4u && model SUNW,Ultra-1 && disksize c0t0d0 1500-2500 Fprom Finstall_s
karch sun4m Fprom Finstall_sun4u_2000- Fupdate
```

Examiner les fichiers en TP



Créer un profil de configuration

304

- /usr/local/jumpstart/rules décrit les profils d'installation à utiliser : installation à faire en fonction des disques, du type de machine, du nom,...

```
# Les stations du maste're :
any - commence_installation stations termine_installation
```

- Un fichier de profil décrit l'installation :

```
install_type initial_install
system_type standalone
# Il n'y a pas de support OEM pour x86 :
cluster SUNWCall
fdisk all solaris delete
fdisk all solaris 2000
partitioning explicit
filesystem c0d0s0 200 /
```



```
fileSYS          c0d0s6 auto /usr
# Pour avoir plein de swap :
fileSYS          c0d0s1 free swap
fileSYS          palo-alto:/opt - /opt ro
fileSYS          palo-alto:/usr/local - /usr/local ro
```



- Nécessité de configuration de paramètres des machines
- Utilisation d'un système de nommage
- Fichier /usr/local/jumpstart/sysidcfg

```
root_password=u5nhT.xHroW5A
system_locale=fr.IS08859-15
timezone=MET
timeserver=ntp-server.cri.ensmp.fr
```



- Chaque OS a des contraintes sur les tailles de partition `fdisk` ~> laisser les OS se débrouiller avec une installation indépendante sur une machine de référence
- Installer dans les 4 partitions
 - ▶ Windows NT
 - ▶ Windows 95 ou 98
 - ▶ Linux (non testé : laissé comme projet élève...)
 - ▶ Solaris7
- Ensuite capturer les paramètres de partitionnement `fdisk` depuis Solaris

```
fdisk -W /usr/local/jumpstart/fdisk_multiple /dev/rdisk/c0d0p0
```

et copier une image des partitions autres que Solaris depuis le serveur `palo-alto`



```
cd /export/verre4
rsh miromesnil "dd if=/dev/rdisk/c0d0p1 bs=512 | \
  /usr/local/bin/gzip -9" > nt-4.0.miromesnil.gz
rsh miromesnil "dd if=/dev/rdisk/c0d0p2 bs=512 | \
  /usr/local/bin/gzip -9" > windows-95.miromesnil.gz
```

- On fera confiance au système multi-boot altruiste de Solaris



- Script de démarrage avant le début de l'installation

```

/usr/local/jumpstart/commence_installation :

# /usr/local/jumpstart est monté dans /tmp/install_config
df
pwd
# Partitionne le disque pour NT & 98 en plus :
/sbin/fdisk -F /tmp/install_config/fdisk_multiple /dev/rdisk/c0d0p0

```
- Script de fin après l'installation

```

/usr/local/jumpstart/termine_installation :

# À la fin de l'installation de Sun,
# le futur / réside en fait dans /a,
# / n'étant que temporaire pour l'installation.
set -vx
# Comme automount ne tourne pas, ni NIS ou DNS :
/usr/sbin/mount -F nfs 10.2.16.200:/usr/local /a/usr/local
# Fait croire à cfengine qu'il est dans /
# alors qu'il est dans /a :

```



```

/usr/sbin/chroot /a /usr/local/sbin/cfengine \
    -v -f /usr/local/share/cfengine/cfengine.conf
# Met le boot par défaut sur le disque Solaris
# pour un démarrage automatique :
eeprom bootpath=/isa/ata@1,1f0/cmdk@0,0:a
## A French Keyboard :
eeprom kbd-type=French
mkdir /tmp/verre4
mount palo-alto:/export/verre4 /tmp/verre4

echo Installation of Windows NT at 'date'...
/a/usr/local/bin/gunzip -c /tmp/verre4/nt-4.0.miromesnil.gz
| dd bs=512 of=/dev/rdisk/c0d0p1
echo Installation of Windows NT ended at 'date'.

echo Installation of Windows 95 at 'date'...
/a/usr/local/bin/gunzip -c /tmp/verre4/windows-95.miromesnil.gz
| dd bs=512 of=/dev/rdisk/c0d0p2
echo Installation of Windows 95 ended at 'date'.

```



- La variable `$lstations` contient la liste de toutes les machines clientes du mastère
- Vérifier que `/etc/hosts` ou le DNS contient le nom et les numéros IP des machines à installer et `/etc/ethers` contient leur numéros Ethernet (pour RARP)
- Création des configuration de démarrage des installations sur le serveur `palo-alto`

```
cd /export/Solaris7_CD-ROM/Solaris_2.7/Tools
for i in $lstations; do ./add_install_client
    -c palo-alto:/export/local/jumpstart
    -p palo-alto:/export/local/jumpstart
    -s palo-alto:/export/Solaris7_CD-ROM $i i86pc;
done
```

- Lancer les installations JumpStart sur les machines avec la disquette Solaris sur PC ou `boot net - install` sur Sun



- <http://www.cfengine.org>
- Automatise des tâches d'administration à partir de fichiers déclaratifs
- Langage à prédicats
- `/usr/local/share/cfengine` contient les fichiers de configuration
- Sorte de système imunitaire lancé régulièrement qui effectue des configurations et réparations
- <http://www.lit.enstb.org/~keryell/publications/conf/2001/JRES2001/cfengine/>
- `/usr/local/share/cfengine/cfengine.conf`

```
control:
    # Where all the configuration files for cfengine are:
```



```

# Will be /usr/local/share/cfengine/cf soon
cf_directory = ( /usr/local/share/cfengine/cf )

import:
# Split things up to keep things tidy
# The main file...
$(cf_directory)/main.cf
# and all the other files sorted by function:
$(cf_directory)/accounting.cf
$(cf_directory)/accounts.cf
$(cf_directory)/apache.cf
$(cf_directory)/automount.cf
$(cf_directory)/cfengine.cf
$(cf_directory)/cron.cf
$(cf_directory)/exportfs.cf
$(cf_directory)/holidays.cf
$(cf_directory)/localmounts.cf
$(cf_directory)/logging.cf
$(cf_directory)/minitel_access.cf
$(cf_directory)/naming.cf
$(cf_directory)/network.cf
$(cf_directory)/ntp.cf
$(cf_directory)/patch.cf

```



```

$(cf_directory)/ppp_access.cf
$(cf_directory)/printing.cf
$(cf_directory)/sendmail.cf
$(cf_directory)/solaris.cf
$(cf_directory)/ssh.cf

```

- `$(cf_directory)/main.cf`

```

control:
193_48_171::
# We are at ENSMP/CRI:
AddClasses = ( ENSMP CRI )

192_44_75::
# We are at ENSTBr/LIT:
AddClasses = ( ENSTBr LIT )

CRI::
site      = ( CRI )
domain    = ( ensmp.fr )
sysadm    = ( respinfo_cri@$(site).$(domain) )

# Site specific stuff:
LIT::

```



```

site      = ( LIT )
domain    = ( enst-bretagne.fr )
sysadm    = ( Ronan.Keryell@enst-bretagne.fr )

# Locations needed for installation:
LIT.solaris::
  PatchDirectory = ( /home/gavotte/calice2/Solaris/Solaris-8-4_01/Patches/8_Recommended )
  OptionalSoftwareDirectory = ( /home/gavotte/calice2/Solaris/Solaris-8-4_01/solaris8_401_suppcd )

# Put the OS religion as a string in the variable $(os):
solaris::
  os = ("solaris")

linux::
  os = ("linux")

# Generic specifications:
any::
  # Where the reference files are to be found:
  shared_conf = ( "/usr/local/share/conf" )

# Declare all the classes that are used to communicate between actions:

```



```

AddInstallable = (
  InstallationTime
  RelaunchSendmail
)

Access      = ( root )      # Only root should run this

timezone    = ( MET )

# Do not collect disabled files in a central repository :
#Repository = ( /var/spool/cfengine )

# What to put in front of cfengine verbosity :
OutputPrefix = ( "cf:$(host)" )

nfstype     = ( nfs )

SecureInput = ( on )

EditfileSize = ( 100000 )

WarnNonOwnerMail = ( true )
WarnNonUserMail  = ( true )

```



```

# Output stuff to Syslog :
Syslog = ( on )
#Verbose = ( on )
Warnings = ( on )

# What to do, in this order:
actionsequence = (
    netconfig
    resolve
    checktimezone
    directories
    files
    copy
    editfiles
    links
    tidy
    shellcommands
    processes
# At least for the Minitel access installation:
copy
editfiles
)

```



- Mise en place de l'*accounting* \$(cf_directory)/accounting.cf

```

control:
    any::
        AddInstallable = ( StartAccounting )

files:
    solaris::
        # Create this file to log the login failures:
        /var/adm/loginlog mode=600 owner=root group=sys action=touch

editfiles:
    solaris::
        {
            # Log all the login failures:
            /etc/default/login
            AppendIfNoSuchLine "SYSLOG_FAILED_LOGINS=0"
        }

        {
            # Add accounting jobs to the adm crontab:
            /var/spool/cron/crontabs/adm

```



```

AutoCreate
AppendIfNoSuchLine "# Accounting stuff:"
AppendIfNoSuchLine "0 * * * * /usr/lib/acct/ckpacct"
AppendIfNoSuchLine "30 2 * * * /usr/lib/acct/runacct 2> /var/adm/acct/nite/fd2log"
AppendIfNoSuchLine "30 7 1 * * /usr/lib/acct/monacct"
DefineClasses "StartAccounting"
DefineClasses "RestartCron"
}

{
# Add accounting jobs to the root crontab:
/var/spool/cron/crontabs/root
AutoCreate
AppendIfNoSuchLine "# Accounting stuff:"
AppendIfNoSuchLine "30 22 * * 4 /usr/lib/acct/dodisk"
DefineClasses "StartAccounting"
DefineClasses "RestartCron"
}

```

links:

```

# To start accounting at boot time :
solaris::

```



```

/etc/rc2.d/S22acct ->! /etc/init.d/acct
/etc/rc0.d/K22acct ->! /etc/init.d/acct

```

shellcommands:

```

solaris.StartAccounting::
# Start the accounting now to avoid waiting for next reboot :
"/etc/init.d/acct start"

```

- Gestion des comptes \$(cf_directory)/accounting.cf

editfiles:

```

solaris::
# Authorized root to remote login as root.
# Useful only for deseperated case...
# That means that in normal circumstance, this should never be
# used except with ciphered methods (ssh...).
{ /etc/default/login
CommentLinesStarting "CONSOLE=/dev/console"
}

```

copy:

```

any::

```



```

$(shared_conf)/$(site)/etc/passwd
  dest=/etc/passwd
  type=byte

$(shared_conf)/$(site)/etc/shadow
  dest=/etc/shadow
  type=byte

$(shared_conf)/$(site)/etc/group
  dest=/etc/group
  type=byte

# Authorized log in without password from machines of the group:
$(shared_conf)/$(site)/etc/hosts.equiv
  dest=/etc/hosts.equiv
  type=byte

# Root can log in from administration machines:
$(shared_conf)/$(site)/rhosts
  dest=/.rhosts
  type=byte

```

- Configuration du réseau `$(cf_directory)/network.cf`



```

control:
  # The netmask to used in our networks:
  any::
    netmask  = ( 255.255.255.0 )

broadcast:
  any::
    # Standard brodcast with all bits 1:
    ones

# It is just for verification by the "netconfig" actionsequence.
# Skip the verification during the OS installation process: it is too early...
defaultroute:
  LIT.!InstallationTime::
    192.44.75.1

  CRI.!InstallationTime::
    193.48.171.48

copy:

```



```
solaris::
  # Setup the file for the default route:
  $(shared_conf)/$(site)/etc/defaultrouter
  dest=/etc/defaultrouter
  type=byte
```

- Configuration du système de nommage
\$(cf_directory)/naming.cf

```
editfiles:
  solaris.CRI::
    # Use DNS and NIS :
    { /etc/nsswitch.conf
      SetCommentStart '#'
      CommentLinesStarting 'hosts:      nis [NOTFOUND=return] files'
      CommentLinesStarting 'hosts:      files'
      CommentLinesStarting 'hosts:      files dns'
      AppendIfNoSuchLine '# Put the DNS in first place to have FQHN. RK.'
      AppendIfNoSuchLine 'hosts:      dns nis files'
    }

  solaris.LIT::
    # Use DNS and files :
    { /etc/nsswitch.conf
```



```
      SetCommentStart '#'
      CommentLinesStarting 'hosts:      nis [NOTFOUND=return] files'
      CommentLinesStarting 'hosts:      files'
      CommentLinesStarting 'hosts:      files dns'
      AppendIfNoSuchLine '# Put the DNS in first place to have FQHN. RK.'
      AppendIfNoSuchLine 'hosts:      dns files'
    }

  copy:
    solaris::
      # Set up the host file :
      $(shared_conf)/$(site)/etc/hosts
      dest=/etc/inet/hosts
      type=byte

      # Set up the netgroup file :
      $(shared_conf)/$(site)/etc/netgroup
      dest=/etc/netgroup
      type=byte

  resolve:
    # Declare the DNS servers to use :
    CRI::
```



193.48.171.40
193.48.171.215
193.48.180.100

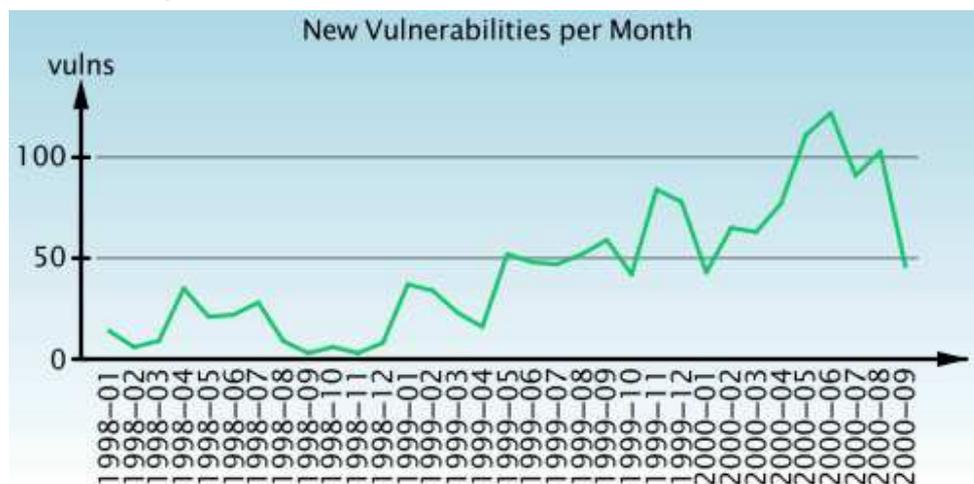
LIT::

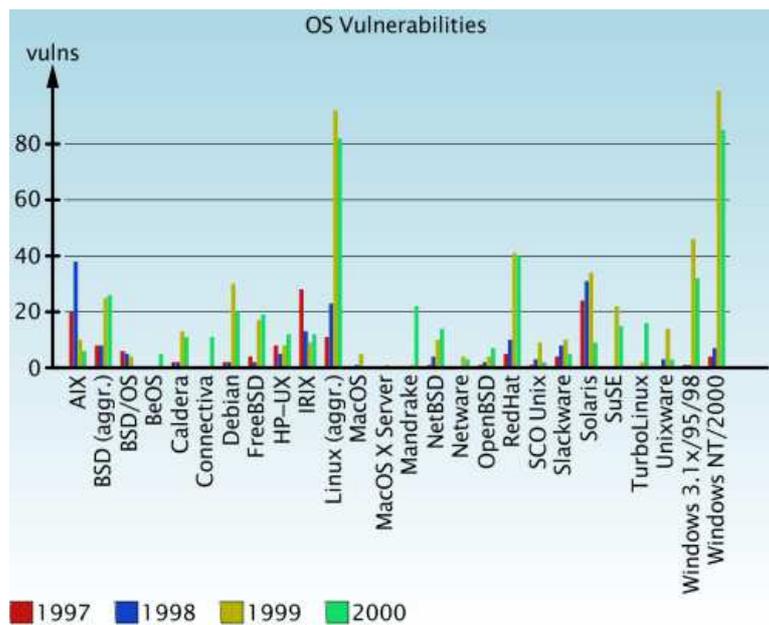
192.44.75.10
192.108.115.2
192.44.77.1



Trous de sécurités

<http://www.securityfocus.com/vdb/stats.html> : BUGTRAQ
Vulnerability Database Statistics





Quelques attaques classiques

Quelques attaques présentes et passées...

- Chevaux de Troie : logiciel qui a des fonctions autres que celles auxquelles on pense
- Portes dérobées : logiciel contenant un moyen d'accès caché au système
- Bombes logicielles : programme qui s'arrête ou détruit des choses sous certaines conditions
- Virus : programme qui modifie le comportement d'un autre pour se diffuser
- Vers : programme se propageant à travers le réseau sans modifier de programme
- Bactérie : se reproduit jusqu'à effondrer le système
- Refus de service : attaque ralentissant le système ou



l'endommageant pour le rendre inutilisable

- Attaques réseau
- Options par défaut « larges »
- Trous de sécurité locaux (et par connexion distants...)
- Authentification faible
- Outils d'audit sécurité utilisés avec malveillance
- Blagues (*hoaxes*)
- Une attaque qui marche peut être publiée sur des listes de discussion *underground*...



- Virus sous Windows en général
- Macro-virus dans *template* de document MicroSoft Word
- BackOrifice et ses semblables : télécommande de Windows
- Faux message de MicroSoft contenant une fausse mise à jour du système
- Fichier *.reg* exécuté par l'administrateur et qui modifie la *base de registres*
- Piratage du site FTP contenant TCP *wrapper* avec une version donnant un shell à une certaine adresse IP
- Utilisation de fausses clés PGP d'un auteur de logiciel
- Potentiellement tout programme téléchargé depuis le réseau...
- Modification du noyau pour cacher tout ce qu'on veut : *root-kit*



de l'extrême

- ▶ FreeBSD : <http://thc.pimml.com/files/thc/bsdkern.html>
- ▶ Linux : http://thc.pimml.com/files/thc/LKM_HACKING.html
- ▶ Solaris : <http://thc.pimml.com/files/thc/slkm-1.0.html>

↔ Réinstaller tout le système...

- Reprogrammation du BIOS (*Basic Input Output System*) de PC, du microcode de Pentium II, d'un modem, d'un disque dur,...



Variantes plus douces

332

- Logiciels qui envoient des informations d'utilisation
 - ▶ Logiciels qui envoient des infos sur leur usage
 - ▶ RealPlayer qui envoie des informations sur ce qui est reçu
 - ▶ Des documents de traitement de texte ou de tableur qui font référence explicitement à une URL : accès à chaque ouverture du document... ↔ Spécialisation de chaque document et traçage de leur usage et propagation ! Regarder les messages qui sortent...
- Logiciels d'échanges de fichiers style Napster ou Gnutella
 - ▶ Pompe de la bande passante
 - ▶ Laisser filer des informations internes (utilisateurs, adresses)
 - ▶ Envoyer des chevaux de Troies

<http://www.research.att.com/~smb/talks/NapsterGnutella/index.htm>



- Trop d'entêtes HTTP, trop gros entêtes HTTP ~> ralentissement
- Inondation de paquet TCP SYN : plus de place pour les vraies demandes de connexion. Problème intrinsèque de TCP/IP..
- Ping de la mort : plusieurs fragments d'un paquet dont la taille est trop grande pour loger dans le tampon de réassemblage
- Paquets TCP avec drapeau OOB chez MicroSoft
- URL trop long pour IIS : arrêt
- IIS meurt avec GET ../..
- Envoie de caractères sur le port 1031 assassine IIS
- DNS de NT meurt si on lui envoie une réponse sans question
- 1 caractère sur le port 53 fait que le DNS de NT utilise le CPU sans discernement



- N'importe quoi sur le port 135 bloque le CPU à 100 % sur NT
- Applet qui prend tout le temps CPU avec plein de threads, ouvrent plein de fenêtres
- Frames HTML récursives



- *Sniffers* de mots de passe (telnet, rlogin, FTP,...)
- Adresses de retour d'erreur du style `|/usr/bin/commande pour sendmail`
- Confusion entre liens Internet et liens sur le bureau dans Windows ↔ page HTML qui peut lancer des commandes arbitraires locales
- ActiveX peut *par construction* faire n'importe quoi
- Divers bugs dans Internet Explorer et Netscape Communicator au niveau de JavaScript et Java qui sortent du « bac à sable » d'exécution
- Programmes CGI foireux, souvent installés en standard, permettant l'exécution de commandes



- Programmes CGI mal conçus trop libres avec trop de droits
- Trop de fichiers visibles via WWW, liens symboliques sortant de `htdocs,...`
- Rajout de « `.` » à la fin d'ASP pour récupérer les sources avec IIS
- « `..` » dans une URL permet de remonter avec IIS
- Exécution de commandes arbitraire par IIS avec des `.bat` ou `.cmd`
- Extension FrontPage permettant d'installer des scripts CGI arbitraires
- Transferts de tables NIS sous Unix
- Faible authentification RPC Unix de base



- Exécution de commandes à distance dans des logiciels de discussion
- Pollution de cache DNS
- Faux sites Internet qui usurpent ou redirige vers les vrais mais en espionnant au passage



- « + » dans `/etc/hosts.equiv` de SunOS 4
- Configuration de serveurs WWW
- Scripts CGI de test
- Compte utilisateur par défaut (`guest`,...)
- Groupe `Everyone` sur MicroSoft peut mettre un cheval de Troie dans `... \system32`
- Partage NetBIOS à tout le monde sous Windows



- Problèmes de conception : droits foireux, conflits dans `/tmp` (lien symbolique vers fichier sensible),...
- Oubli de perte de privilège de programmes *suid* : démon *finger* et *sendmail* capable de lire n'importe quel fichier
- Débordement de tampons alloués dans la pile (variables *automatic* du C) : si donnée trop grande débordement dans une zone mémoire stockant l'adresse de retour de fonction ~> exécution de code arbitraire dans des programmes *suid*, *ftpd*,...
- Exploitation de méta-caractères dans un *shell*
`mail moi ; mail pirate < /etc/passwd`
- Changement de la variable `$IFS` qui définit les séparateurs.
Commande `/bin/ls` comprise comme `bin ls`... Si commande `bin` dans le `$PATH` d'une commande *suid* `root` qui fait un



`system()` : bug de preserve

- Modification d'un fichier provoquant exécution pirate
 - ▶ `~/.login`, `~/.profile`, `~/.cshrc`, `~/.bashrc`,... exécutés lors du lancement de shell
 - ▶ `.exrc` dans `~` ou dans le répertoire local exécuté par `vi` ou `ex`
 - ▶ `~/.emacs` Emacs-Lisp lancé au démarrage d'Emacs
 - ▶ `~/.forward` ou `~/.procmail` exécuté lors de la réception d'un courriel
 - ▶ `~/.netscape` lancement de *plug-ins*, `~/.mailcap` lancement de méthodes MIME
 - ▶ ...



- Mots de passe trop simples : approche par dictionnaire
- Si récupération des mots de passe hachés (/etc/shadow Unix ou SAM NT) comparaison accélérée sans passer par le processus de *login* (outils Crack UNIX/NT & L0phtCrack NT)
- Différentes méthodes d'authentification SMB plus ou moins fortes (compatibilité avec le passé...). Attaque avec plus ou moins de force possible
- Mots de passe hachés stockés dans le SAM (*Security Accounts Manager*) de NT par DES avec une clé dérivée du Relative Domain ID. Après un *Emergency Restore Disk* reste une copie lisible par tout le monde dans ... \system32\ERD
- Mots de passe d'Access 97 stockés simplement avec un XOR d'une constante de 13 octets



- Très pratique pour faire un audit de réseau
-  Mais les failles trouvées peuvent aussi servir à un attaquant...
- Beaucoup d'outils existent
 - ▶ SATAN
 - ▶ COPS
 - ▶ Tiger
 - ▶ ISS
 - ▶ Nessus
 - ▶ ...
- Les utiliser pour fermer les failles avant que d'autres les essayent...



- « Si vous recevez un mail avec le sujet *truc-muche* votre disque dur s'autodétruira, votre mari/femme vous quittera, etc » signé IBM ou MicroSoft pour donner du poids
- « Surtout propagez ce message à tous les cyber-crétins que vous connaissez tellement que c'est important »
- Faire une chaîne de pétition pour une raison humanitaire qu'il faut envoyer à une personne : submergée de courriels
- Fait perdre du temps et de la bande passante
- Essayer de vérifier l'origine des informations, des programmes,...
- Mais qu'est-ce qui est vrai ? Qu'est-ce qui est faux ?...



- Faire tourner avec un droit *minimal* (pas `root` !)
- Virer tous les caractères néfastes en entrée qui pourraient entraîner une utilisation malicieuse (méta-caractères dans adresse de mail pour exécuter une commande en plus de l'envoi du mail)
- Utiliser le mode « teinté » de `perl` qui empêche tout ce qui vient de l'extérieur d'être utilisé dans des commandes qui modifient des fichiers ou processus ou dans des sous-shells
- Faire de l'audit de code
- Faire des tests intensifs hors-norme (envoi de tonnes de données,...)



- Serveur de commerce électronique
 - ▶ Ordinateur
 - ▶ Connexion Internet
 - ▶ Serveur WWW, FTP, courriel, connexion à distance
 - ▶ Système d'exploitation
 - ▶ Divers logiciels
- Pas de point faible ⇔ sécurisation à tous les niveaux
- Même si le serveur ne contient pas toutes les données de l'entreprise sa compromission est critique : image de marque, données clients,...
- Avoir toujours les versions à jour (ou alors carrément obsolètes ?...)



- Faire aussi face aux refus de service
- Comment faire face lorsque les temps de développement sont revus à la baisse ?...
- Programmeurs non formés à la sécurité en général...
- Formation par la bande dessinée ! CdV Entrevue N°17, 2000



- Ordinateur possédant plusieurs services
- Mettre en place le *minimum* de services nécessaires : WWW, courriel,...
-  Beaucoup de services lancés par défaut...
- Compromis à trouver entre fonctionnalités pratiques et trous de sécurité potentiels
- Bien configurer chaque service
- Vérifier le contrôle d'accès et l'authentification des utilisateurs (mots de passe,...) : accès limités au minimum
- Bien définir les droits sous lesquels tournent les serveurs : minimaux
- Bien cerner les fichiers accessibles par les différents serveurs :



minimaux. Si par ftp on peut modifier des fichiers du serveur WWW (.htaccess, CGI,...)...

-  <http://defaced.projectgamma.com> : les attaquent existent



- Utilisation de pages WWW dynamiques : scripts CGI (Common Gateway Interface) à base de langages style perl, shell,...
- Usage de langages de script plutôt que des langages de programmation plus classique comme C : plus simples à mettre en place pour de petites applications, plus haut niveau
-  Fonctions permettant d'accéder au système à partir des entrées (méta-caractères,...) si mal conçus
-  Trou dans un CGI : passe à travers les pare-feux !
 - ▶ Modification de fichiers locaux
 - ▶ Envoi d'informations locales par courriel à des destinataires hostiles arbitraires
 - ▶ Chargement et exécution de programmes d'espionnage ou de portes dérobées



- ▶ Refus de service par effondrement de la machine locale
- ▶ ↔ programmer proprement et bien vérifier : traiter tous les cas possibles et  impossibles, gérer les dépassements de capacité, les retours d'erreur, plus de place disque, plus de mémoire,...
- ▶ Faire vérifier ses programmes par d'autres programmeurs
- ▶ Rajouter des assertions partout ou presque
- ▶ Faire une analyse statique simple : recherche de `system()`, `open()`, `popen()`, `eval()` et « ' ' » dans les scripts perl par exemple
- Limiter l'importance d'un trou en faisant tourner les CGI avec des droits minimaux (`nobody`) et en l'enfermant dans un répertoire (`chroot`) si possible
- Centralisation de tous les CGI et vérification par une instance



spécifique

- Éliminer les CGI installés par défaut qui peuvent être des passoires
- Vérifier régulièrement que les CGI n'ont pas changé (stockage de leur signature MD5 ailleurs)
- Ne pas mettre trop d'options sur le serveur : compromis...
 - ▶ Affichage automatique du contenu d'un répertoire si pas de `index.html` : pratique mais si dans un répertoire sensible dont on veut cacher le contenu...
 - ▶ SSI (*Server Side Include*) permet d'exécuter des commandes citées dans des pages HTML. Pratique mais si on peut écrire une page HTML avec

```
<!-- #exec cmd="/bin/cat /etc/passwd" -->
```
 - ▶ Suivit des liens symboliques : plus une hiérarchie simple



mais un tas de spaghetti difficile à cerner...

- ▶ Existence de répertoires de CGI personnels : l'administrateur ne sait plus où aller vérifier tout ce qui est rajouté



Besoin de spécialiser l'accès de l'information en fonction du client

- Par nom de machine ou numéro IP : accès en Intranet par exemple
 - ▶ Accès de pages WWW réservées à telle machine
 - ▶  Piratage de DNS ↔ au moins double vérification
IP → nom → IP
 - ▶  Possibilité d'usurper une adresse IP ou de pirater une machine autorisée
- Par utilisateur & mot de passe
 - ▶  Éviter les mots de passe faibles. Si c'est le client qui choisit...
 - ▶ Vérifier qu'il n'y a pas d'erreur par des tests...
 - ▶  Mots de passe passant en clair sur le réseau



- ▶ Ne pas divulguer le fichier de mots de passe
- ▶ Fichier de mots de passe doit être crypté de manière forte
- ▶ Possible de répartir les fichiers de contrôle d'accès dans les répertoires à protéger (.htaccess,...). Plus simple pour l'utilisateur, cauchemar pour l'administrateur.  Si bug permettant la récupération de ces fichiers...
- ▶ Si authentification ultérieure par *cookies* pour éviter de redemander le mot de passe, revérifier au moins le numéro IP car les cookies passent en clair sur le réseau...
- Authentification forte par certificats
 - ▶ Protocole style SSL
 - ▶ Chaque client doit demander un certificat à une entité officielle de certification
 - ▶  Vol de son certificat



- En général besoin d'une base de données pour stocker diverses informations
 - ▶ Liste de produits
 - ▶ Liste de clients
 - ▶ Transactions en cours
 - ▶ Autorisations d'accès (fichier ou format DBM)
- Certaines bases de données ont une interface WWW ~> hérite des problèmes inhérents
-  Souvent options par défaut très libérales pour aider la mise en œuvre
- Bien cerner les accès aux différents domaines de la base de données pour l'extérieur mais aussi par service en interne à l'entreprise



- Encore une fois : éviter les mots de passe faciles
- Interfaces pour simplifier les CGI : oraper1,...



- Ubiquité : tout ce qui est fait sur une machine peut être fait sur un groupe de machines (Unix)
 - ▶ Terminaux virtuels distants (rlogin, telnet)
 - ▶ Accès de fichiers à distance (NFS)
 - ▶ Courrier électronique
 - ▶ Annuaire distribués (DNS, finger, whois, ph, LDAP)
 - ▶ Distribution du temps (NTP)
 - ▶ Téléconférence (multidiffusion, Mbone)
 - ▶ Écrans distants (XWindowS version 11 Release 6)
 - ▶ Remote Procedure Call (RPC), RMI, CORBA
 - ▶ WWW (synonyme d'Internet par réduction...)
- Fait partie de la vie de tous les jours



- Trous de sécurité \rightsquigarrow *World Wide Bugs...*



- Gros site : trop de machines variées pour faire un sécurisation par machine
- ~> approche centralisée plus simple : filtrage au niveau de l'accès réseau extérieur
- Goulet d'étranglement forçant le passage Internet par un pare-feu
- Ne laisse passer que ce qui est permis : courriel, FTP, connexion à distance,...
- Essaye d'éliminer les inconvénients d'Internet en gardant les bénéfiques
- Constitué par un ou plusieurs matériels : routeurs et/ou ordinateurs avec différents logiciels



- Tenir le système à jour en fonction des nouvelles attaques à la mode
- Possible d'acheter un système clé en main ou d'en faire un soi-même. Utile d'avoir une expertise de toute manière pour la configuration d'un système clé en main
- Garde éventuellement des traces des transferts ou des attaques
- N'empêche pas une protection minimale des machines du réseau
- N'empêche pas une charte de sécurité interne
-  Si accès Internet supplémentaire autre que par le pare-feu
-  Pas de protection efficace contre les virus
- En cas de panne : bloquer tout plutôt que tout laisser passer. Multiplier les défenses pour limiter les problèmes de



configuration erronée à un endroit

-  Endroit de rêve pour un pirate pour espionner et contrôler ce qui passe par Internet...
- Tester régulièrement le système avec des tests d'intrusion
- Faire quelque chose de simple et de maîtrisable



Permission optimiste ou pessimiste ?

- Interdire tout par défaut
 - ▶ Naturelle pour les administrateurs
 - ▶ Dès qu'un nouveau service apparaît : bloqué !
 - ▶ Autorisation si utile après inspection et analyse des faiblesses et de leurs implications ~> expertise même si système clé en main
 - ▶ Nouveau trou de sécurité : plus de chance d'être bloqué
- Autoriser tout par défaut
 - ▶ Naturelle pour les utilisateurs : + de liberté
 - ▶ Dès qu'un nouveau service apparaît : utilisation immédiate aussi bien client que serveur
 - ▶ Nouveau trou de sécurité ~> probablement exploitable



- ▶ ↔ Cauchemar des administrateurs
- Nécessité d'un bon charisme



- Par numéro IP source
- Par numéro IP destination
- Par protocole (TCP, UDP, ICMP, IGMP, IPIP,...)
- Par port (\approx service) TCP ou UDP source
- Par port (\approx service) TCP ou UDP destination
- Par type de message ICMP
- Par interface (interne, externe,...) en entrée ou en sortie
- Si début de connexion TCP

Utile pour filtrer des services simplement associés à la notion de port
: SMTP,...



- Certains services ne sont pas simplement associés à un port : FTP, RPC, applications MBone,...
- Passer par un serveur intermédiaire sur le pare-feu qui accède au service distant : notion de *proxy* (mandataire, délégué)
- Passerelle applicative
- Doit être transparent à l'utilisateur
- Restrictions et contrôle effectués par le mandataire
- Possibilité d'un niveau d'authentification supplémentaire au niveau du mandataire
Exemple : `telnet` intermédiaire sur le pare-feu avant `telnet` sur cible
- Mandataire à effet de bord positif : cache WWW pour tout un site



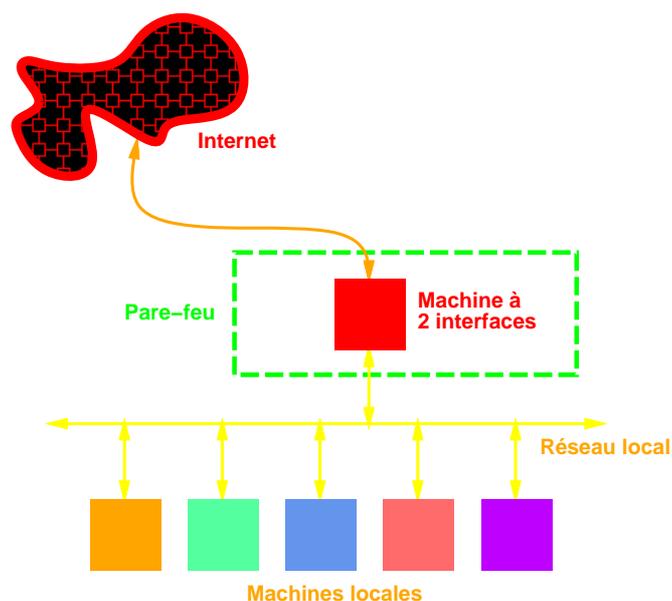
-  ~> Clients internes ne doivent pas communiquer directement avec serveurs extérieurs
- Problème : certains serveurs mandataires imposent d'avoir des clients mandataires aussi (SOCKS,...)
- Certains services sont difficilement déléguables et filtrables...
MBone



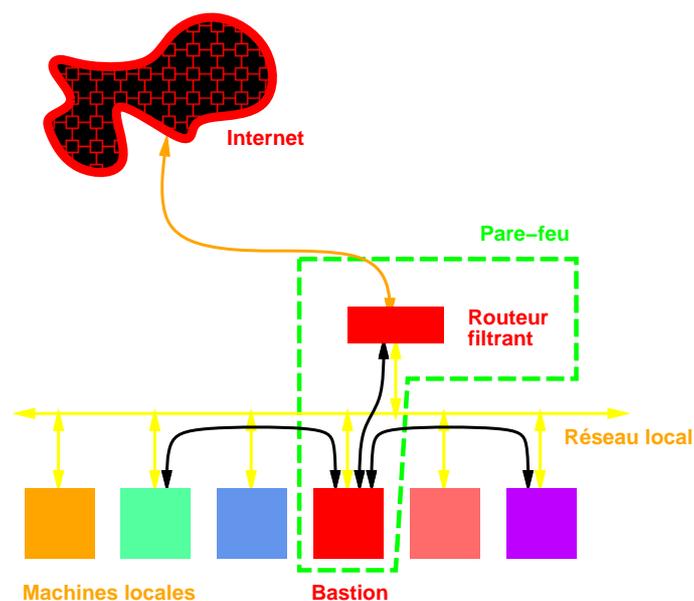
Faire apparaître une machine avec une autre adresse

- Fait sortir des machines avec adresses IP privées RFC 1597 avec des adresses publiques
- Cache les vrais numéros IP derrière une adresse (ou plusieurs pour dépasser le nombre de ports/machine)
- Orthogonal à la notion de pare-feu mais souvent associée
- Réalloue les ports de sortie pour éviter les conflits (unicité des connections)
- Problème : faire la traduction (IP local, port local) → (IP masquante, port masquant) en fonction de (IP destination, port destination). Comment savoir quand une traduction n'a plus de raison d'être si pas de fin explicite (UDP,...) ? Éliminer la traduction au bout d'un « certain » temps...

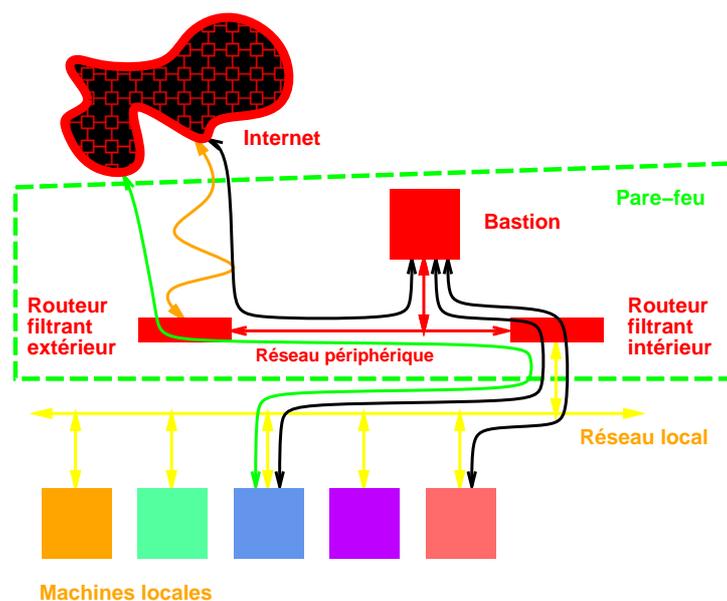
Pare-feu avec machine à double réseau



- Ordinateur à 2 interfaces sur 2 réseaux différents
- Architecture simple
- Supprimer le routage entre les 2 réseaux
- Seuls possibilité de passer d'un réseau à l'autre : mandataires
- Se connecter sur une machine interne : se connecter d'abord sur le pare-feu
- ⚠ Si le pare-feu est piraté, la sécurité disparaît : écoute du réseau,...



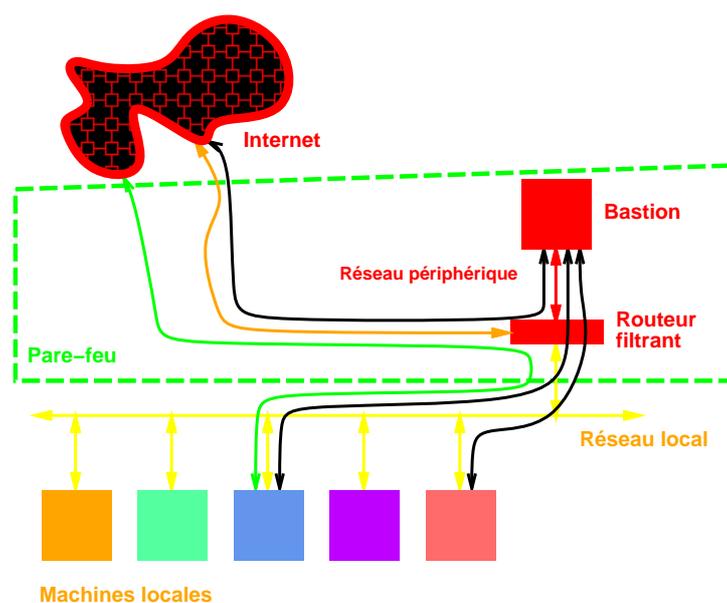
- Seule connexion possible avec l'extérieur : le bastion
- Filtrage oblige à passer par les mandataires sur le bastion
- Filtrage par routeur : moins sujet à des piratages ~> probablement plus sûr que l'approche machine à double réseau
- ⚠ Si le bastion est piraté, la sécurité disparaît : écoute du réseau,...



- Pare-feux précédents : problème instantané si piratage du pare-feu car en contact avec le réseau interne
- Idée : isoler le bastion dans un réseau périphérique (DMZ : *DeMilitarized Zone*) séparé du réseau interne
- Si piratage du bastion : encore difficile d'accéder au réseau interne
- Pas possible d'espionner le réseau interne depuis le bastion
- Possibilité de fusionner routeur extérieur avec bastion
- ⚠ Ne pas fusionner bastion et routeur intérieur : si piratage bastion...
- Si DMZ connectée à plusieurs réseaux intérieurs, n'utiliser qu'un routeur interne : si plusieurs routeurs internes, du trafic interne pourrait passer par la DMZ et être espionné



Pare-feu avec réseau périphérique et routeur unique



- Transparent pour les machines internes : le routeur détourne certains paquet vers le bastion



- 1 ou plusieurs machines visibles depuis Internet
- Accueille les amis comme les ennemis
- Place fortifiée
- Faire simple pour être simple à sécuriser
- Prendre en compte dès le départ le piratage potentiel du bastion
- Avoir une procédure de reconstruction automatique du bastion
- Installer système d'exploitation minimal
- Un Unix de bonne facture est un bon choix car de nombreux outils disponibles pour Internet
- Pas besoin d'une puissance énorme : peut être une vieille machine



- Mémoire raisonnable pour éviter le swap tout en gardant trace de toutes les connexions et faire tourner tous les mandataires
- Besoins de plus de puissance si liaison rapide et services plus gourmand : WWW avec bases de données, News,... Répartir la charge sur plusieurs machines
- Utiliser un modèle de machine et de système d'exploitation déjà présent sur le site : développements du système, installation, maintenance,...



- Partir d'un système standard
- Appliquer toutes les mises à jour de sécurité
- Mettre le minimum de services
- Désactiver les services inutiles pour le bastion
- Services sécurisables simplement par filtrage : ne passent pas par mandataire sur le bastion et n'apparaissent pas sur le bastion
- Mettre les services mandataires sur le bastion
- Installer les services peu sûrs sur une machine jetable
- Ne pas mettre de comptes utilisateurs
- Protéger les traces de fonctionnement



- ▶ Consulter de manière routinière les fichiers sur le bastion
- ▶ Envoyer en plus une copie à une machine reliée par liaison série pour les coups durs
- Faire un audit sécurité avant de mettre en production



Suppression de services Unix

- 2 sortes de serveurs :
 - ▶ Services lancés au démarrage : `/etc/rcx.d`
 - ▶ À la demande par le super-démon `inetd` : `/etc/inetd.conf`
- En plus système de RPC (*Remote Procedure Call*) : `rpcbind` ou `portmap` qui fait la traduction entre numéro du service RPC et le vrai port alloué sur la machine. `rpcinfo -p [machine]` donne la liste des services RPC enregistrés
- Supprimer NFS (services RPC) : `nfsd`, `biod`, `mountd`, `statd`, `automountd`,...
- Supprimer NIS : `ypbind`, `ypserv`,...
- Supprimer services de démarrage : `tftpd`, `bootpd`,...
- Supprimer la majorité des services via `inetd`



- Protéger les services `inetd` restant par système style TCP Wrapper
- Remplacer certains services utiles par des sous-services : `finger` permet d'avoir des informations sur les gens ayant des comptes sur la machine. Remplacer par un programme donnant seulement de l'information administrative sur le site. Dans `/etc/inetd.conf` :

```
finger stream tcp nowait nobody /bin/cat cat /etc/finger_info
```
- Suppression du routage (IP *forwarding*) afin d'éviter que la machine serve de relais
- Suppression du routage IP par la source



- Contrôle l'exécution de serveurs lancés par `inetd`
- Modification des lignes d'`/etc/inetd.conf` pour lancer `tcpd` (TCP Wrapper) à la place de chaque serveur ~> pas de modification d'`inetd` ou des serveurs
- `tcpd` lance les vrais serveurs une fois les vérifications faites
 - ▶ Paramétré par `/etc/hosts.allow` et `/etc/hosts.deny`
 - ▶ Fait une vérification par adresse → nom → adresse
 - ▶ Autorisation par service
 - ▶ Par nom de la personne ayant la socket sur le client via IDENT (RFC 931) pour TCP mais attention aux mensonges
- Garde une trace de toutes les connexions via `syslog`
- Peut rajouter des bannières et des messages d'erreurs



paramétrables

-  Bien protéger tous les serveurs de `/etc/inetd.conf`
-  Problème intrinsèque à UDP : pas de notion de fin de connexion \rightsquigarrow un serveur UDP peut continuer de tourner après une requête autorisée et servir ensuite une requête qui aurait dû être interdite...
- <ftp://ftp.porcupine.org/pub/security/index.html>



- *Yet Another Solaris Security package*
- Durcit une version de Solaris fraîchement installée
- Configuration centralisée : `/etc/yassp.conf`

<http://www.yassp.org/>



- Contrôle du passage des paquets selon une politique d'autorisation
- En général mis en place dans un routeur qui a en plus la tâche de faire suivre les paquets entre différents réseaux : bon endroit pour faire du filtrage car point de passage
- Filtrage possible par adresses IP, ports (\approx service), sens,... mais pas utilisateur, données de plus haut niveau (contenu, virus,...)
- 1 (gros) routeur filtrant suffit à filtrer tous les paquets d'un (gros) site : simplicité
- Indépendant des utilisateurs et des mandataires
- Interdiction à des faux paquets de partir sur Internet avec fausses sources



- Penser que les communications sont en général à double sens : bloquer un seul sens ne suffit pas (certaines attaques se passent des réponses)
- Certains services ne sont pas contrôlables par simple filtrage (RPC, rsh, rlogin,...)



- Services basés sur IPv4 (aujourd'hui)
- Utilisation des protocoles
 - TCP (*Transmission Control Protocol*)
 - ▶ Mode connecté point à point
 - ▶ Bidirectionnel
 - ▶ Système d'ordonnancement des paquets et de retransmission des paquets perdus
 - UDP (*User Datagram Protocol*)
 - ▶ Mode non connecté : pas de perte de temps à établir une connexion
 - ▶ Plus fort débit
 - ▶ Sans garantie
 - ICMP (Internet Control Message Protocol)



- ▶ Echo Reply (ping)
- ▶ Destination Unreachable : pas de route vers la destination
- ▶ Source Quench : encombrement
- ▶ Redirect : change une route (en route plus directe par exemple)
- ▶ Time Exceeded for a Datagram : TTL arrivé à 0
- ▶ ...
- IGMP (*Internet Group Management Protocol*) : distribution de routes et de groupes de multidiffusion (Mbone)
- IPIP : encapsulation d'IP dans IP. Utilisé pour faire des tunnels (Mbone). Sécurité normalement gérée par les gestionnaires de tunnels (`mroute`,...)



- Protocole de transport sur Internet
- Options spécifiques dans l'entête IP rarement utilisées sauf pour la mise au point et les... attaques
 - ▶ Option de routage par la source : définition d'un point de passage obligé pour les paquets à l'aller comme au retour. Idée contourner des tables de routage défaillante. En pratique : contourner des routeurs filtrants...
 - ▶ Solution radicale efficace : supprimer tout paquet avec option(s)
- Fragmentation : certaines liaisons sur le parcours ne peuvent pas transmettre des paquets trop long (ne pas monopoliser une ligne/paquet trop longtemps...)
 - ▶ ↪ Système capable de fragmenter les paquets en cours de



route

- ▶ Défragmentation à l'arrivée
- ▶ Attaque par refus de service : envoi de paquets IP avec des fragments manquants pour occuper les tampons de défragmentation en attente de paquet qui n'arriveront jamais
- ▶ Seul le premier fragment contient les informations sur le type de paquet. Si filtrage sur protocole (TCP, UDP,...), port, etc. filtrage
 - Du premier paquet seulement : laisse passer tous les autres fragments ↪ possible de faire une attaque par refus de service en entrée et faire sortir de l'information en entrées/sortie ↪ tunnel dans les fragments suivants sans log potentiel...
 - De tous les paquets : système capable de suivre la fragmentation



-  Ne pas laisser entrer des paquets forgés avec des adresses internes provenant de... l'extérieur
- Notion d'adresses de diffusions au niveau réseau :  si paquet avec adresse source de diffusion, génération possible de tempêtes de diffusions...
- Sécurisation dans Unix : seul `root` à le droit d'ouvrir une socket de numéro < 1024 (ports privilégiés)
 - ▶ Évite qu'un pirate non `root` établisse un faux serveur de login en entrée pour capturer les mots de passes
 - ▶ Authentification rudimentaire : si on reçoit un paquet *d'une machine Unix* et d'un port privilégié c'est que cela vient de `root`



- Liaison bidirectionnelle entre une machine d'origine (IP,port) et une machine destination (IP,port)
- Connexion différente si couples différent
- En particulier plusieurs connexions sur un même port d'une même machine. Exemple HTTP sur port 80
- Données découpée en segments avec un numéro de séquence (position du premier octet du segment dans le flux) et un numéro d'acquittement (octet reçu jusqu'à cette position)
- Retransmission de segment si pas reçu d'acquittement au bout d'un certain temps
- Destruction de tout segment reçu avec somme de vérification incorrecte



- Établissement de la connexion
 - ▶ Envoyeur envoie un segment avec le drapeau SYN (*Synchronize Sequence Number*) et un numéro de séquence aléatoire qui servira d'origine depuis l'envoyeur
 - ▶ Destinataire envoie un segment avec le drapeau ACK (*Acknowledgement*) et le drapeau SYN avec un numéro de séquence qui servira d'origine depuis le destinataire
- Chaque paquet ensuite envoyé contient le drapeau ACK pour acquitter en même temps les paquets reçus
- Pour éviter trop d'attente d'acquiescement, chaque paquet a un champ fenêtre indiquant le nombre d'octets qu'on peut envoyer en avance sans attendre d'acquiescement. Si fenêtre à 0 : gel de la transmission
- Adaptation de la fenêtre au taux d'erreur et à la congestion



- Empêcher des connexions TCP : éliminer simplement le paquet de connexion (SYN sans ACK)
- Autorisation des connexions dans un sens (par exemple intérieur vers extérieur) en fonction du sens du premier paquet avec SYN sans ACK (règle de filtrage de type *ack* ou *established*)
- Forgeage de paquets : injecter des paquets avec les bons couples (IP,port) mais aussi les bons numéros de séquence \rightsquigarrow numéro de séquence aléatoire au début. Probabilité $\times 2^{-32}$
- Des comportements subtils sur les cas limites servent de signature des implémentations de IP... Fuite d'informations pour une attaque spécifique



- Mode non connecté : pas de perte de temps à établir une connexion
- Unidirectionnel entre origine (IP,port) et une destination (IP,port)
- Simple
- Mode de diffusion et multi-diffusion \rightsquigarrow attention aux paquets avec comme source des adresses de diffusion !
- Plus fort débit
- Sans garantie : peut ne pas arriver ou au contraire en plusieurs exemplaires
- Simple à forger : injecter un paquet avec les 2 couples (IP,port) adéquats
- Applications avec communications bidirectionnelles : renvoyer



des paquets dans l'autre sens. Nécessite d'ouvrir un passage subséquent dans le sens (IP destination,port destination) \rightarrow (IP origine,port origine) momentanément



- 64K ports UDP et TCP associés à des services bien connus par IANA
- Insuffisant pour tous les services imaginables \rightsquigarrow service d'annuaire transformant un numéro de service RPC 32 bits en port UDP ou TCP : `portmap` ou `rpcbind`
- Chaque serveur RPC lancé *localement* s'inscrit via le port TCP 111 pour enregistrer son port
- NFS en général sur UDP et TCP 2049
- Les clients interrogent `portmap/rpcbind` pour un service RPC donné pour avoir le port serveur TCP ou UDP à contacter
- Problème de filtrage : difficile de connaître les ports à contrôler
- Par contre une attaque peut essayer les 64K ports rapidement



jusqu'à tomber sur un service connu

- Nécessite une interaction entre le filtre et `portmap/rpcbind`
- Bloquer NFS et NIS en RPC : tous les 2 en UDP \rightsquigarrow bloquer tout UDP sauf quelques services non RPC (DNS)



- Écrire les règles dans un fichier de configuration plutôt que de manière incrémentale sur le routeur
- Règles exécutées *dans l'ordre* : recharger (tftp) à chaque fois *tout* le fichier pour éviter des problèmes
- Utiliser des adresses IP plutôt que des noms : évite attaques DNS et des étreintes mortelles (DNS filtré par nom...)
- Possibilité de garder une trace des paquets refusés ou acceptés



Filtrage par adresse

400

- Empêcher des paquets avec source interne falsifier de rentrer
- Altruisme : empêcher des paquets de sortir avec une adresse non locale
- Faire confiance à certaines machines extérieures.  adresses sources falsifiées

Exemple sur Cisco :

```
interface Ethernet0
ip address 194.214.157.2 255.255.255.0
ip access-group 100 in
media-type 10BaseT
! Effacer l'access-list avant de commencer
no access-list 100
! Les adresses locales
```



```

access-list 100 deny ip 192.54.148.0 0.0.0.255 any
access-list 100 deny ip 192.54.172.0 0.0.0.255 any
access-list 100 deny ip 192.54.173.0 0.0.0.255 any
access-list 100 deny ip 127.0.0.0 0.255.255.255 any
! Adresses privées du RFC 1597
access-list 100 deny ip 10.0.0.0 0.255.255.255 any
access-list 100 deny ip 172.16.0.0 0.15.255.255 any
access-list 100 deny ip 192.168.0.0 0.0.255.255 any
! École des Mines, site de Paris
access-list 100 permit ip 192.54.165.0 0.0.0.255 any
access-list 100 permit ip 194.214.158.0 0.0.0.255 any
    
```



Filtrage par port

- Filtre les services associés à des ports UDP ou TCP
- Interdiction de certains services entrant mais autorise en sortie
- Empêche certains services de passer directement sans mandataire du bastion adresses sources falsifiées

Exemple sur Cisco :

```

no service udp-small-servers
no service tcp-small-servers
!pour le cri (R.Keryell) roazhon, chailly - dmi.ens.fr, trefle.ens.fr
access-list 100 permit tcp 129.199.96.17 0.0.0.0 192.54.172.242 0.0.0.0 eq 6000
access-list 100 permit tcp 129.199.96.17 0.0.0.0 192.54.172.200 0.0.0.0 eq 6000
access-list 100 permit tcp 129.199.96.11 0.0.0.0 192.54.172.242 0.0.0.0 eq 6000
access-list 100 permit tcp 129.199.96.11 0.0.0.0 192.54.172.200 0.0.0.0 eq 6000
access-list 100 deny udp any any eq echo
access-list 100 deny tcp any any eq echo
access-list 100 deny tcp any any eq 11
access-list 100 deny tcp any any eq 15
access-list 100 deny udp any any eq bootps
    
```



```
access-list 100 deny udp any any eq tftp
access-list 100 deny tcp any any eq 87
access-list 100 deny tcp any any eq 95
access-list 100 deny tcp any any eq sunrpc
access-list 100 deny udp any any eq sunrpc
access-list 100 deny tcp any any eq 144
access-list 100 deny udp any any eq snmp
access-list 100 deny udp any any eq xdmcp
access-list 100 deny tcp any any eq exec
access-list 100 deny udp any any eq biff
access-list 100 deny udp any any eq who
access-list 100 deny tcp any any eq cmd
access-list 100 deny udp any any eq syslog
access-list 100 deny tcp any any eq lpd
access-list 100 deny udp any any eq rip
access-list 100 deny tcp any any eq 2000
access-list 100 deny tcp any any eq 2001
access-list 100 deny tcp any any eq 2002
access-list 100 deny tcp any any eq 2003
access-list 100 deny udp any any eq 2049
access-list 100 deny tcp any any eq 2049
access-list 100 deny tcp any any eq 6000
access-list 100 deny tcp any any eq 6001
```



```
access-list 100 deny tcp any any eq 6002
access-list 100 deny tcp any any eq 6003
access-list 100 permit ip any any
```



- Filtre IP qui marche avec la majorité des Unix (module chargeable)
- Autorise/empêche des paquets de passer
- Fait le tri entre toutes les interfaces
- Trie tous les protocoles IP
- Filtre les paquets IP fragmentés
- Gère les sessions TCP
- Trie les paquets IP avec des options spéciales (`traceroute`,...)
- Peut renvoyer des erreurs ICMP ou reset TCP à réception de paquets bloqués
- Peut garder trace de l'état des connexions pour éviter de repasser par tout le processus de filtrage



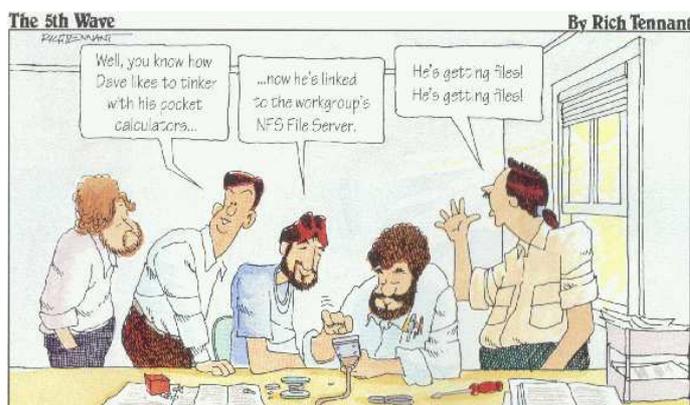
- Traduction d'adresse (NAT)
- Redirection de connexions vers des mandataires
- Possibilité d'enregistrer des informations sur ce qui se passe
- Utilisés sur des pare-feux commerciaux (BSD)
- Testé sur BSD, Solaris, HP-UX
- <http://coombs.anu.edu.au/~avalon>



- Plusieurs systèmes de pare-feu existent avec leur propres langages...
- ↔ Utiliser un langage commun et un compilateur vers les différents langages
 - ▶ IP Filter (Unix)
 - ▶ ipfw/ipfwadm (Linux)
 - ▶ ipfirewall (Linux, BSD)
 - ▶ Cisco avec les *extended access-lists*
 - ▶ screend
- <http://coombs.anu.edu.au/~avalon/flc.html>



-  Beau-
coup de
choses à
vérifier...
- Utiliser une
check-list



Cf. *Practical UNIX & Internet Security*

http://www.bigmouse.net/literature/Oreilly/puis/appa_01.htm



- Dans la lignée de la fête des mères, des pères,...
- Le dernier vendredi de juillet
- Penser au moins un jour dans l'année au pauvre ingénieur système !
- Jour de reconnaissance de la personne qui fait marcher l'entreprise

<http://www.sysadminday.com/>



Conclusion

- Unix est un système robuste et éprouvé : vieux mais a évolué
- Couvre tous les types d'ordinateur
- Ouvert : accepte les standards
- Énormément de possibilités
- Très configurable
- Solaris 9
 - ▶ Extensible : rajout de modules et de matériel à chaud (processeurs, disques, mémoire)
 - ▶ 64 bits : gros fichiers, grosse mémoire pour chaque processus
 - ▶ Administration à plusieurs niveaux en fonction des besoins : basique avec interface graphique jusqu'aux systèmes très



spécialisés avec les fichiers textuels

- ▶ Installation automatique avec installation de plusieurs OS sur la même machine

↔ Travaux pratiques



Table des transparents

1 Administration système ?

Introduction

2 Plan

3 Introduction

4 Fragments d'archéologie

13 Sources d'information

17 Distribution Solaris 9

19 Quoi de neuf ou mieux dans Solaris 9 ?

22 Outils d'administration dans Solaris

Outils

23 Admintool

24 Solstice

25 Solaris Management Console

26 Penser « infrastructure »

28 Composant d'une infrastructure

30 Gestion de versions et Emacs

32 Utilisateurs et mots de passe

Responsabilités utilisateur

33 Choix des mots de passe

36 Stockage des mots de passe

39 Mots de passe jetables

Mots de passe jetables

41 S/Key

44 OPIE

47 Compte utilisateur

Utilisateurs

50 Numéros d'utilisateurs et de groupes Unix

Identificateurs utilisateurs

53 Administration des comptes

55 Outils textuels de gestion des comptes



58	Le fi chier /etc/passwd	86	<i>Devices</i> — Fichiers conducteurs de périphériques
60	Fichier /etc/shadow	88	Fichiers cachés
62	Le fi chier /etc/group	89	Prévention sur les fi chiers
64	Répertoires utilisateurs	91	Access Control List
65	Shells et fi chiers de login	92	Représentation des types de droits
68	Variables d'environnement	94	Manipulation des ACL
70	Role-Based Access Control (RBAC)	96	Au début était le démarrage...
Contrôle d'accès basé sur des rôles		Démarrage	
71	Utilisation d'un rôle	100	Niveaux de fonctionnement
72	Inspection des rôles	104	Démarrage en mode mono-utilisateur
73	Gestion des rôles	105	Changer un niveau de fonctionnement
74	Base de donnée attributs des utilisateurs	106	Lancer des choses au démarrage
75	Base de donnée autorisations	109	La hiérarchie de fi chier dans Solaris
76	Base de donnée profi ls	Hiérarchie	
77	Base de donnée attributs d'exécution	111	La hiérarchie de /
79	Droits des fi chiers	121	La hiérarchie de /usr
Attributs fichiers et sécurité		127	La hiérarchie de /export
82	Numéros réels et effectifs	129	Paquets de logiciels



Administration Solaris —Conclusion—

DÉPARTEMENT INFORMATIQUE — ENST BRETAGNE



Paquets de logiciels		157	Services réseau
131	Automatisation de l'installation de paquets	159	Courrier électronique
133	Hiérarchie /usr/local	Courrier électronique	
Hiérarchie /usr/local		160	Le sendmail standard client
135	pkg-get ou « les paquets pour les nuls »	161	Le sendmail central routeur de courrier
136	Exemple d'organisation de /usr/local	163	Ajouter du matériel
142	Patches	Ajout de matériel	
Rajout de patches		164	Rajout de périphérique
143	Systèmes de nommage	165	Rajout de périphérique — version hardcore
Systèmes de nommage		166	Information sur la confi guration matérielle
145	Système de nommage par fi chiers	167	(Disques magnétiques)
146	NIS	Disques	
149	NIS+	169	Partitionnement des disques
150	Confi guration réseau	172	Nommage des disques logiques
Configuration réseau		175	Utilitaire format
151	Interface	Formatage	
153	Routage	176	format à l'œuvre
155	Nommage	179	Étape fdisk sur PC



Administration Solaris —Conclusion—

DÉPARTEMENT INFORMATIQUE — ENST BRETAGNE



181	Systèmes de fichiers sous Solaris	221	Restauration de / et /usr
Système de fichiers		222	Transport portable de système de fichiers
185	Montage/démontage d'un système de fichiers	223	Clonage de système de fichiers
186	Table des fichiers montés	225	Quotas : contrôle de l'usage des disques
188	Montage de systèmes de fichiers	227	Network File System — NFS
190	Démontage	NFS	
191	UFS journalisé	230	Exportation de fichiers
196	Vérification d'un système de fichiers	232	Montage de fichiers via NFS
199	Système de fichiers cache (CacheFS)	234	Auto-monteur
201	Mise en œuvre CacheFS	235	Cartes de montage
204	Rajouter de la mémoire virtuelle	238	Gestion des médias amovibles
207	Fichiers temporaires en mémoire — TMPFS	Médias amovibles	
209	Nécessité des sauvegardes	239	Insertion médium amovible
Sauvegardes		241	Éjection médium amovible
211	Organisation de sauvegardes	242	Formattage disquettes
215	Dériveurs de bande	244	Système d'impression
216	ufsdump/ufsrestore	Impression	
219	Création de snapshot	245	Rajouter une imprimante — client



Administration Solaris —Conclusion—

DÉPARTEMENT INFORMATIQUE — ENST BRETAGNE



247	Contrôle imprimante	Installation	
249	Tâches planifiées	281	Installation manuelle avec Web Start
Planification de tâches		282	Installation manuelle avec suninstall
251	Interaction avec d'autres systèmes	283	Création d'un serveur d'installation
Interactions autres systèmes		286	Serveur d'installation depuis images réseau sans média
253	ftp	288	Installation automatique
255	SSH[2] — <i>Secure Shell</i>	Installation automatique	
SSH		289	Déclaration d'une nouvelle machine
261	Utilisation de ssh	290	Fichier de préconfiguration
264	SSH — exemple d'utilisation en Intranet	291	Fichier de règles d'installation
267	Détection des changements et mise à jour	292	Profils d'installation
Audit		293	Erreurs classiques & mise au point
269	Audit et journaux de fonctionnement	294	Création d'un profil client
273	Syslog	Installation automatique LIT	
275	Analyse de log : swatch	303	Exemple ENSTBr/enseignement
276	Autres traces	ENSTBr/enseignement	
277	Résoudre les problèmes	304	Créer un profil de configuration
278	Procédures d'installation	Exemple maître	



Administration Solaris —Conclusion—

DÉPARTEMENT INFORMATIQUE — ENST BRETAGNE



306	Fichier d'identification sysidcfg	345	Sécurisation serveur commerce électronique
307	Clonage autres OS sur PC	Sécurisation serveur WWW	
309	Scripts d'installation	347	Sécurisation serveur
311	Installation du boot JumpStart des machines	349	Sécurisation serveur WWW
312	Outil de configuration GNU cfengine	353	Contrôle des accès
326	Trous de sécurité	355	Bases de données
Attaques		357	Réseaux et Internet
328	Quelques attaques classiques	Internet	
330	Chevaux de Troie	359	Sécurisation réseau
332	Variantes plus douces	Pare-feu	
333	Refus de service	362	Permission optimiste ou pessimiste ?
335	Attaques réseau	364	Filtrage de paquets
338	Options par défaut « larges »	365	Filtrage par mandataire
339	Trous de sécurité locaux (et distants...)	367	Traduction d'adresse
341	Authentification faible	368	Pare-feu avec machine à double réseau
342	Outils de sécurité	370	Pare-feu avec machine filtrante
343	Blagues (<i>hoaxes</i>)	372	Pare-feu avec réseau périphérique de filtrage
344	Script CGI	374	Pare-feu avec réseau périphérique et routeur unique



Administration Solaris —Conclusion—

DÉPARTEMENT INFORMATIQUE — ENST BRETAGNE



376	Bastion	399	Écriture de règles de filtrage
Bastion		400	Filtrage par adresse
378	Réalisation d'un bastion	402	Filtrage par port
380	Suppression de services Unix	405	IP Filter
382	TCP Wrapper	Filtrage IP avec du logiciel libre	
384	YASSP : sécurisation de Solaris	407	Filter Language Compiler — flc
385	Mise en place filtrage paquets	408	Check list
Filtrage paquets		409	La fête des ingénieurs systèmes !
387	Internet = IP ?	Conclusion	
389	IP	410	Conclusion
392	TCP/IP	412	Table des matières
395	UDP		
397	RPC		

