Organisation et administration des systèmes informatique

Christophe Gouinaud

1er décembre 2005

Table des matières

Chapitre 1

théorie

1.1 Introduction

Un système informatique d'entreprise sert à mettre à disposition des employés les données et les moyens de les modifier utiles à leur travail. En langage contracté, nous dirons : mettre à disposition des utilisateurs des ressources.

Citons quelques-unes de ces ressources :

- les moyens de communication électronique
  Par exemple, un système de messagerie nominatif façon Email.
- les moyens de produire des textes et de faire des comptes
  Par exemple, un tableur et un traitemnent de texte.
- les moyens de commande ou de contrôle de certaines machines ou de certaines unités de production.
  Le terminal de commande d'une fraiseuse numérique ou d'un système de cogénération électrique.

D'un autre côté, au coeur du système informatique, les données et les programmes sont en général stockés dans des fichiers. La mise à disposition, la consultation et la modification de ces fichiers est donc le mécanisme de base de l'utilisation de l'informatique.

Le problème principal de l'organisation d'un système informatique sera donc de faire en sorte que tout utilisateur dispose rapidement de l'ensemble des fichiers dont il a besoin pour réaliser sa tache.

Nous avons donc coutume de dire que la personne qui décidera de l'organisation de ce système devra par conséquent concentrer ses efforts sur la simplification des procédures informatiques et bien tenir compte de l'impact des choix qu'elle fait par rapport aux méthodes de travail des divers employés.

Notre lecteur aura compris qu'appliquer la maxime précédente revient à choisir une bonne organisation pour l'ensemble des fichiers qui représente la partie cachée du système informatique. Une deuxième conclusion est qu'il faudra également définir qui accède à quels fichiers et pour quelles actions. Cela induit qu'il faudra identifier chaque utilisateur et définir des actions autorisées en fonction de groupes communs à plusieurs utilisateurs.

Les deux sous-parties suivantes présentent les règles d'identification des utilisateurs et les méthodes permettant de partager les ressources de façon à en garantir une utilisations sûre.

1.2 Identification des personnes

Sur la plupart des systèmes d'exploitation, les utilisateurs sont définis par un nom et un mot de passe. Par habitude, le nom de l'utilisateur pour le système informatique est appelé login et le mot de passe est en général désigné par le mot password. Le login sert à identifier l'utilisateur et le mot de passe à l'authentifier, c'est-à-dire à vérifier son indentité.

Afin de faciliter le travail des utilisateurs, il est souhaitable que ceux-ci n'aient besoin d'utiliser qu'un seul couple login mot de passe pour toutes les ressources informatiques dont ils ont besoin :

- Accès interactif au poste de travail
- Accès au système de fichiers de l'entreprise
- Accès aux applications fédératives et aux progiciels
- Accès aux services messagerie, workflow et intranet
- Accès externe vers le système de l'entreprise

Cette liste n'est bien évidemment pas limitative, et il est souhaitable que tout soit mis en oeuvre pour que le système ne demande qu'une seule fois par session le fameux sésame.

Réaliser cela demande de mettre en interopérabilité de nombreux outils. Si par exemple vous utilisez un système de messagerie basé sur Imap sous Linux, une base de données Oracle et un poste Windows 2000, il faudra que tous ces systèmes acceptent les mêmes logins et les mêmes mots de passe. La partie **??** explique comment mettre tout cela en oeuvre.

Croire que cette paire de clefs unique est un gadget est une erreur lourde de conséquences. Dans ce domaine, on voit souvent de tout. Beaucoup de structures ont un seul login mot de passe pour accéder au poste interactif alors que de nombreuses données confidentielles sont stockées sur le disque des postes, ou alors multiplient les login différents pour les différentes ressources, conduisant ainsi les utilisateurs des mêmes services à utiliser un seul compte pour toutes les opérations. Dans les deux cas, cela pose un problème de responsabilité en cas de fausse manoeuvre ou de fuite.

L'avantage de l'identifiant unique est qu'il permet de créer un sentiment de propriété chez les utilisateurs et donc à les responsabiliser quant à leur usage. Si tout le monde utilisait sa carte bancaire pour s'identifier au système, peu de personnes accepteraient de communiquer leur code à leurs collégues. A partir du moment où l'identifiant est associé à une messagerie internet, où circulent forcément des informations personnelles, et que l'utilisation de son mot de passe est impérative pour ouvrir une session permettant d'y accéder, la communication de mots de passe devient marginale.

Une fois tout le monde identifié et authentifié, il devient facile de restreindre les droits d'accès des utilisateurs à ce qui est nécessaire et par conséquent à créer une amorce de politique de sécurité satisfaisante. Il ne s'agit pas de réaliser un bunker impénétrable mais de pouvoir associer à chaque personne utilisant le système un profil qui lui permette de travailler facilement en ayant sous les yeux ce dont il a besoin à l'exclusion de ce qui lui est inutile et pourrait par exemple le conduire à perdre du temps en utilisant un outil inapproprié. L'identification permet aussi de restreindre facilement les accès de l'extérieur vers le système informatique de l'entreprise tout en le rendant possible par des procédures standards. Dans ce cadre, il est également plus facile de débrouiller une panne dans la mesure où l'utilisateur qui en a été victime laisse des traces à son nom dans le système.

Les systèmes informatiques actuels implémentent des niveaux d'accès tels qu'ils sont représentés sur la figure 1.1.. La partie world accède via le réseau à des services publics qui ne distinguent pas de droit d'accès. La partie entreprise est gérée via la notion de domaine qui est une communauté d'ordi-

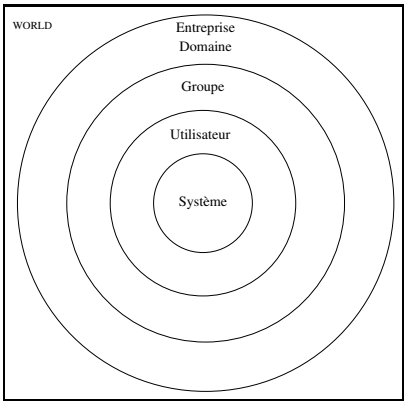


FIG. 1.1 – Niveau d'accès des systèmes d'exploitations.

nateurs partageant les mêmes clefs d'identification. Les groupes correspondent à un sous découpage du domaine regroupant des utilisateurs ou des machines permettant l'accès à certains services. Le niveau utilisateur se caractérise par l'accès interactif et par la définition du propriétaire des fichiers. Le niveau système est en général réservé à un utilisateur particulier dont le rôle est de configurer le système.

Un des rôles de l'ordinateur dans l'entreprise est de faciliter le travail en groupe. L'efficacité d'un système informatique reposera donc en particulier sur une bonne définition des regroupements d'utilisateurs. Afin de faciliter celui-ci, il est le plus souvent préférable de n'avoir qu'un seul domaine dans toute l'entreprise. En effet, gérer plusieurs domaines facilitent le cloisonnement et limite les risques de panne mais complique les possibilités d'échanges internes et demande des manipulations importantes pour gérer les déplacements, occasionnels, périodiques ou définitifs des utilisateurs. La stratégie de partage devra donc reposer complétement sur la structure de groupe.

La définition des groupes informatiques est un problème complexe du fait de l'organisation même des entreprises. En effet les entreprises sont en général structurées en services comportant des groupes de travail. Mais il existe aussi une hiérarchie, donc un *groupe* des chefs de projet ou de services, des métiers transversaux, l'administration, le secrétariat qui constituent autant de pôled ayant leurs propres besoins de communication et d'outils.

D'un autre côté, les groupes informatiques permettent de contrôler les accès à :

- des fichiers partagés,
- des applications partagées comme Oracle ou SAP,
- des fonctions particulières, comme valider une commande, dans des applications,
- des moyens d'accès réseau, intérieur ou extérieur,
- la distribution de logiciels sous licence sur un réseau

La structure de groupe devra donc permettre de mettre en correspondance les utilisateurs et les droits d'accès aux ressources. Il convient donc de respecter les règles suivantes quant à la structure des groupes :

- définir un groupe regroupant tous les utilisateurs. Son rôle sera de permettre d'agir globalement sur le système pour, par exemple, mettre en place une stratégie de sécurité minimale.
- Définir un groupe par rôle vis a vis du système informatique et des applications : générique, développeur et administrateur système. Il faudra également des groupes pour des tâches spéci-

fiques telles que la manipulation de périphériques (lecteur de bandes, sauvegarde, imprimante).
- Définir un groupe par service de l'entreprise
- Définir des groupes reflétant les niveaux hiérarchique. Le plus souvent, chef de service, chef de projet et cadre.
- Définir des groupes pour les métiers transversaux, secrétariat par exemple.
- Des groupes pour les communautés de travail de l'entreprise, projet, groupe de vente, ...
- Définir un groupe permettant de contrôler l'accès extérieur via le réseau. Il s'agit de contrôler l'accès au VPN ou tout autre moyen permettant de connecter une machine extérieure au réseau de l'entreprise. Nous verrons dans la partie **??** que ceux-ci peuvent être très limités.

Afin que cette structure soit utile, facile d'usage et maintenable, il convient de respecter les règles suivantes :

- Utiliser un nommage simple et significatif des groupes, utilisant des préfixes. Par exemple Scompta, pour le service compta, Scom pour le service communication, Rtel pour les accès téléphoniques et par exemple chefs pour le staff. Il faut impérativement éviter d'utiliser des noms trop longs ou trop cryptiques.
- Avoir pour chaque groupe un outil de communication, mailling list, accessible via le nom du groupe (info@jackscie.com pour l'informatique). Là encore, moins les utilisateurs auront de noms à retenir mieux ça marchera.
- Séparer les données des groupes.
- Avoir une définition des groupes partagés entre les différents types de systèmes d'exploitation et donc une liste unique.
- Documenter la liste des groupes et diffuser le contenu de ceux-ci.
- Ne pas créer de login au nom du groupe. Ceci pose des risques de confusion et risque de pousser tout un service à ne plus utiliser que celui-là.
- Avoir une gestion la plus automatisée possible des groupes. Si, par exemple, aux groupes sont associés des mailling lists, celles-ci devront être mises automatiquement à jour à partir de la base des groupes. Il s'agit ici d'un point clef permettant de garantir la cohérence du système et par conséquent son efficacité.
- Créer rapidement des groupes pour des besoins ponctuels en étant rigoureux quant à leur disparition.
- Avoir un système de mise en cohérence automatique quand des hétérogénéités informatiques rendent impossible la mise en place d'une base de données unique de groupe.
- La manipulation des groupes peut être déléguée pour une partie. Nous pouvons aisément permettre à un chef de service de manipuler les groupes de ses groupes de travail à condition de lui fournir une interface simple pour le faire. Il faudra bien évidemment limiter les possibilités de modification au strict nécessaire en premier lieu pour éviter les pannes.
- L'attribution des groupes aux fichiers doit être transparente. En effet, pour que les utilisateurs d'un groupe puissent manipuler un fichier, il faut que celui-ci appartienne au groupe. Or la plupart des systèmes créent des fichiers au groupe par défaut de l'utilisateur. Il faut donc modifier le système de fichiers de façon que les droits soient automatiquement attribués.

L'appartenance à un groupe doit se manifester comme l'accès à des ressources partagées sans autre mot de passe. Cela permet une gestion souple des droits d'accès notamment en ce qui concerne leurs révocations. En effet, en cas de départ d'un utilisateur, si l'accès à un service est géré par un mot de passe, celui-ci doit être changé, ce qui sera soit un problème, soit oublié une fois sur deux.

Les groupes permettent aussi de limiter la mauvaise diffusion d'informations et donc de limiter les fausses manipulations dans les applications.

1.3 Organisation logique des espaces de données

1.3.1 Définition et règles de conception

Les systèmes de fichiers des ordinateurs actuels ont pour caractéristique commune d'être arborescents et de reconnaître au moins trois niveaux d'accès (utilisateurs, groupes et le reste du monde). Un système de fichiers d'entreprise sera donc une agrégation d'arborescences diverses devant reflèter au mieux l'organisation et les modes de travail de l'entreprise.

Dans ce domaine, l'administrateur devra s'efforcer de concevoir des choses simples. En effet une organisation trop complexe, même si elle apparaît totalement logique sur le papier, posera à l'usage trop de problèmes aux utilisateurs et multipliera les risques d'erreurs d'administration. Le risque principal pour l'administrateur est que les utilisateurs stockent leurs fichiers sur des supports autres que ceux prévus (disques durs locaux, disquettes, diskonkey, cdrw) qui causeront à terme de nombreuses pertes de données.

A l'inverse, une absence d'organisation globale conduira les utilisateurs à multiplier les partages de données sauvages et, à l'extrême, à s'échanger des données uniquement par la messagerie. Le risque induit par ce type de pratique est important du fait de la fragilité de ce service, de sa sensibilité au virus et du fait que l'envoi vers plusieurs personnes d'un même document comporte forcément un risque de création de versions concurrentes.

La plupart des postes de travail actuels sont pourvus de disque dur de grande capacité qui permettent le stockage d'un grand volume de fichiers. La tentation est donc grande de les utiliser comme support de données pour les fichiers utilisateurs. Chaque utilisateur enregistre ces données sur le disque dur de son poste de travail et les espaces centralisés sont inexistants ou d'utilisation facultative. Des deux situations, la deuxième est la pire, et les deux conduisent à des coûts de maintenance prohibitifs et des pertes de temps importantes. En effet, la panne d'un poste de travail se traduira par un risque de perte des données de l'utilisateur, le changement de poste de travail, même temporaire, sera difficile, le travail en groupe sur un document obligera soit à de nombreuses recopies, soit à un déplacement des personnes générateur de perte de temps. Cette solution pose de plus un problème de sauvegarde redoutable du fait qu'il est difficile de sauvegarder pendant les périodes de travail et que l'on n'est pas garanti que les postes de travail seront allumés la nuit[1]. Quand on additionnera le temps perdu, le coût des logiciels de sauvegarde et l'énervement généré, la nécessité d'un espace centralisé devient évidente.

S'il est aisé de déployer un serveur de fichiers, sa maintenance est par contre plus délicate. En effet, s'il est utilisé par toute l'entreprise, il devient une ressource indispensable. En cas de panne tout le système informatique s'écroule, il faut donc le surveiller étroitement et être capable de le réparer rapidement. Une centralisation complète des fichiers posera plusieurs problèmes dont :

– la consommation d'espace
  L'histoire récente de l'informatique montre qu'elle est sans borne et exponentielle. Les fichiers

[1] ce qui représente une dépense importante en électricité et n'est guère citoyen

deviennent de plus en plus volumineux et les utilisateurs gardent de plus en plus d'archive numérique. Il faudra donc borner intelligemment l'espace qui leur est donné à l'aide de quotas et leur fournir une façon simple de dicerner les travaux en cours des archives dont la disponibilité immédiate n'est pas requise. Il faudra également surveiller étroitement le niveau de remplissage des disques de façon à prévenir toute pénurie d'espace, soit par l'ajout de nouveaux disques, soit par une politique d'archivage musclée.

– la saturation de l'accès réseau du serveur
  Il faut bien distinguer le cas où à certain moment le service de fichiers est lent et conduit les utilisateurs à attendre quelques secondes la lecture ou la sauvegarde des fichiers, du cas où la saturation conduit à des pertes de données du fait d'un enregistrement incomplet des fichiers. Le premier cas est souvent l'annonciateur du deuxième et les causes devront en être systèmatiquement recherchées. A l'usage, il apparaît que ce problème empire rapidement, typiquement sur une semaine ou deux, et est souvent concomittent au déploiement d'une nouvelle version de logiciels, à l'apparition d'un nouveau service ou d'une nouvelle classe d'utilisateurs.
  Ce dysfonctionnement se produira en général si le nombre de postes connectés est trop grand ou s'il existe des utilisateurs travaillant sur de grandes masses de données, comme par exemple des personnes réalisant des tâches de traitement d'images. Dans le premier cas, la solution sera de passer à un nombre de serveurs plus grand en trouvant un découpage compatible avec le fonctionnement de l'entreprise. Le deuxième cas pourra se résoudre en distribuant mieux les données.
  Une dégradation brutale du service, aléatoire le plus souvent ou périodique quelques fois, mais sans lien apparent avec l'écoulement de la journée de travail de l'entreprise correspond le plus souvent à une mauvaise configuration de poste client ou à l'emploi d'opérations inappropriées. Par exemple, des recherches de fichiers dans un répertoire partagé volumineux peuvent conduire à une saturation temporaire qui peut devenir problématique si elle se reproduit trop souvent. Une recherche du coupable et une bonne explication résoudra le problème pour un coût très inférieur à celui d'un nouveau serveur.

Ces contraintes étant posées, nous allons maintenant exposer la logique de la conception d'un système de fichiers partagés d'entreprise. L'implémentation pratique (ie : avec des serveurs et des disques) sera quant à elle abordée dans la partie **??**.

Une entreprise est en général découpée en services regroupant des collaborateurs. Ces services comportent des groupes de travail. Les individus ont en plus un métier et une fonction. Il existe donc beaucoup de découpages et d'associations possibles. Le pyramidage de l'entreprise est donc conçu de façon que les travaux soient réalisés suivant une hiérarchie qui exprime les responsabilités. Or, bien souvent, des tâches demandent un travail transversal qui devra être réalisable par des personnes de service, de fonction et de position hiérarchique différents.

Le problème principal pour refléter cette organisation est, comme cela est dit plus haut, que les systèmes d'exploitation reconnaissent trois niveaux de découpage : l'individu, le groupe et le domaine. La tendance naturelle de l'administrateur système est donc de concevoir les échanges de fichiers comme devant se faire à l'intérieur d'un groupe, représentant un niveau de hiérarchie concrétisé comme une sous-arborescence. Ceci est malheureusement la plupart du temps impossible ou, à terme, ingérable.

A la lecture de ce qui précéde, vous aurez compris que la création d'un système de fichiers d'entreprise relève de la partie de corde raide. Il s'agit de trouver une stratégie de découpage des espaces

qui facilite le travail des utilisateurs et ne leur complique pas trop la vie. C'est une opération difficile pour un informaticien qui, lui, jongle facilement par nature avec les fichiers et les répertoires et aura donc du mal à évaluer la complexité de ce qu'il fait. Dans ce domaine, il faudra donc essayer de faire le plus simple possible. Il faut donc concevoir les choses dans un esprit dynamique de groupe de travail en ayant une arborescente la plus courte possible et nécessitant le mimimum de déplacement récurent de données.

Les règles minimales d'organisation à respecter pour la gestion des systèmes de fichiers sont :

- ne pas mélanger les services, car cela pose des problèmes insolubles en cas de changement de répartition géographique nécessitant l'ajout de serveur.
- ne pas s'appuyer sur le niveau hiérarchique. Les chefs de services vont et viennent dans la plupart des entreprises.
- ne pas découper un répertoire de service sauf par ordre alphabétique.
- ne pas mettre les répertoires utilisateur dans des répertoires de groupe de travail car ceux-ci ne sont pas pérennes.
- découper de façon à gérer les fichiers suivant leur durée de vie dans le système
- tenir compte du temps qui passe, c'est-à-dire mettre en place des procédures d'archivage et être rigoureux dans la suppression des espaces et des comptes devenus inutiles.

Le non respect de règles fixes, même si celles que je donne constituent pas un dogme, risque de conduire à une situation difficile à géré générant forcément de gros ennuis.

Par exemple, si l'administrateur système d'une entreprise créer un espace pour personnes en contrat à durée déterminé et un espace pour les contrats à durée indéterminé, il se rajoute aussitôt une tâche de gestion du fait des transformation de contrat. Dans ce cas le problème est en plus aggravés par le fait que le ces changements se feront suite à la communication du changement de contrat par le service de gestion du personnel et non du fait d'arrivé à une échéance. Nous constatons ici qu'il faut poser le problème en terme de gain de souplesse et de facilité de maintenance rapporté au temps passé. Dans, l'exemple présent le ghetto à CDD n'apporte que le gain d'éviter les risques d'effacement de l'espace d'un permanent lors de l'archivage et de la surpression d'un espace appartenant à un ex-collaborateur.

1.3.2 Découpage de l'espace disque

L'expérience montre qu'il est impératif de disposer de deux types d'espace :

- un répertoire par utilisateur associé au login
- un répertoire par groupe de travail associé au groupe du système d'exploitation.

Ces deux types suffisent en général. Il faut en général éviter les espaces publics où tout le monde peut tout faire car ils deviennent rapidement des jungles aussi *indispensable* pour les utilisateurs qu'inextricables pour les administrateurs.

Du point de vue de l'organisation informatique, il faut concevoir une arborescence qui permette de dégrouper rapidement des fichiers. Cela permettra de faire face à l'augmentation de volume des fichiers en permettant la dispersion sur plusieurs disques ou à simplifier la conception du réseau en permettant l'éclatement sur plusieurs serveurs. Plus les différents services seront mélangés, plus leur séparation sera difficile et moins le système sera capable de subir les changements d'échelle.

D'un point de vue pratique, on créera une arborescence à deux niveaux contenant trois types d'espace :

- un premier correspondant à un découpage fonctionnel de l'entreprise, en général les départements,
- un deuxième correspondant aux utilisateurs, eventuellement redécoupé de façon à donner des morceaux de taille à peu près comparable.
- Un troisième corespondant aux groupes de travail effectifs.

Le répertoire de chaque département contiendra les groupes de travail corespondant aux service. Un utilisateur pourra donc aisément voyager entre différents répertoires en n'ayant le plus souvent qu'un seul niveau de répertoire à franchir. Cela lui permettra de gagner du temps et lui évitera de se perdre et permettra aux administrateur de régler facilement les déplacmeents de fichiers entre utilisateurs quand la structure de groupe de travail sera prise en défaut.

Pour éviter de fréquentes recherches à l'utilisateur et pour faciliter l'archivage des travaux finis, il est bon de structurer, à l'aide de quelques répertoires types, les données utilisateurs. En effet, la recherche de fichiers est une tâche fastidieuse et chronophage pour les non informaticiens. De plus, la mise en archive automatique des données non utilisées ne sera possible que si elles sont localisées à un endroit précis. Une bonne organisation dans ce domaine permettra de lutter efficacement contre l'engorgement des disques, d'obtenir une séparation des données des différents projets et donc de faire gagner du temps à tout le monde. Nous pouvons d'ailleurs remarquer que la plupart des systèmes informatiques fournissent une organisation type à ce niveau (image, son,...). Quelque soit l'organisation choisie, elle devra, bien evidemment, être documentée et expliquée aux nouveaux utilisateurs pour qu'ils l'utilisent de façon correcte.

Les répertoires types devront au moins contenir :

- un répertoire pour les boîtes emails que l'on peut structurer en boîtes types.
  Ceci facilite pour l'administrateur la gestion des espaces consommés en autorisant un nettoyage automatique des mails effacés ou un archivage des mails envoyés depuis longtemps.
- un répertoire personnel privé pour que l'utilisateur stocke les données n'ayant rien à voir avec le travail.
  Cet espace peut paraître superflu, mais il permet aux personnes inquiètes de disposer d'un espace où elles se sentent chez elles et donc d'éviter beaucoup de mauvaises manipulations des droits du répertoire principal, génératrices d'ennuis. Le fait qu'il existe un espace privé chez chacun limite l'espionite entre salariés en encadrant strictemnt les explorations via les droits des fichiers. Cela pose également une limite entre données liées à la vie privés et ce qui appartient à l'entreprise et donc permet de régler facilement les conflits en cas de séparation.
- un répertoire pour tous les documents à faible durée de vie : compte-rendus de réunion, notes, courriers, circulaires, etc ...
- un répertoire d'archives lui-même structuré temporellement.
  Les utilisateurs devront mettre ici les documents qu'ils n'utilisent plus mais qui doivent être gardés au cas où. Obtenir une bonne utilisation d'un tel espace n'est pas chose facile mais il s'agit d'un impératif pour permettre un historique correct de la vie de l'entreprise. Le découpage temporel pourra se faire par exemple en années et être archivé automatiquement sur des supports non réinscriptibles comme nous l'expliquons dans la partie 1.3.4.
- un répertoire de projet type
  Ceci doit être fait quand le travail sur un projet nécessite la manipulation de plusieurs do-

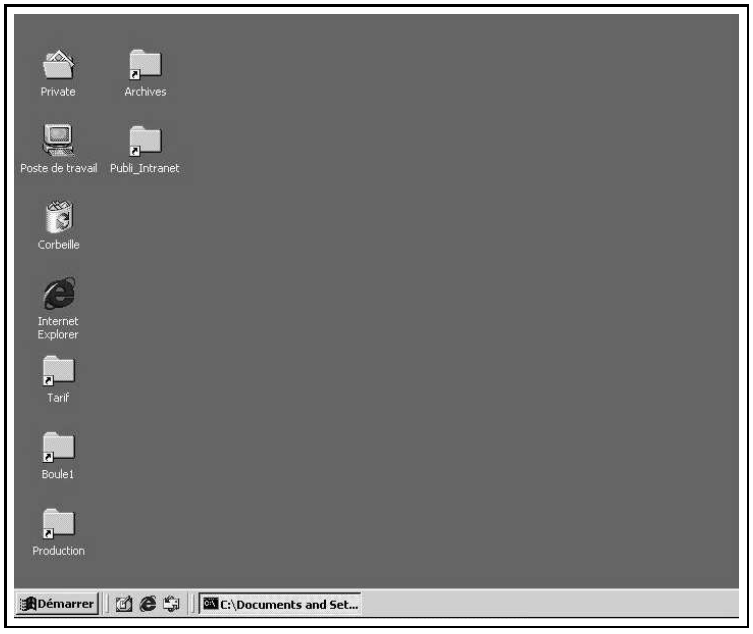


FIG. 1.2 – Bureau utilisateur préconfiguré.

cuments. S'ils ne se limitent pas à la création d'un unique répertoire, l'administrateur devra fournir une commande permettant de les créer à la demande.

– un répertoire de publication de documents
  Ce répertoire doit permettre de publier en lecture uniquement, par exemple sur un intranet, et permettre d'éviter de recourrir à la messagerie systématiquement pour ce genre de tâche.

Pour réaliser cette organisation, il est souvent bon de demander son avis à plusieurs personnes de l'entreprise et en particulier aux secrétaires qui connaissent en général très bien le mode de travail de l'entreprise.

Ensuite il convient également de créer des arborescences distinctes pour :
– les données de base de données ou d'applications partagées,
– les logiciels partageables,
– les fichiers d'intranet,
– les fichiers à destination de l'extérieur,
– et une contenant les outils de maintenance (correctifs, scripts, distribution de logiciels, images de disque, etc ...).

Ceci dans le but de faciliter la sauvegarde, maitriser l'occupation des espaces en évitant qu'une fausse manipulation d'utilisateur ne bloque une base de données importante et en facilitant la vie de l'équipe système qui aura toujours les outils dont elle a besoin sous la main.

Une fois cette arborescence réalisée, il va falloir lui faire un peu de publicité pour qu'elle devienne perceptible aux utilisateurs. Une bonne documentation facilement accessible est un mimimun mais la bonne solution est de fournir des bureaux préconfigurés en fonction du profil de l'utilisateur.

Sur ce bureau, il faut faire apparaître une icône pour chaque espace que l'utilisateur doit avoir à utiliser. Ainsi celui-ci n'aura pas à rechercher ou écrire ces fichiers. Il est aussi possible de lui mettre des icônes fonctionnelles lui permettant de changer les droits des fichiers quand il n'est pas possible de le faire automatiquement. Un exemple de bureau est donné sur la figure 1.2.

Bien évidemment, une fois ces arborescences créées, il faut affecter à tous les répertoires les bons droits. Bien souvent les administrateurs ont tendance à restreindre énormément les droits de lecture ce qui est en général une erreur. D'une maniére générale, il est raisonnable qu'au sein d'un même groupe on puisse lire les fichiers de tout le monde à l'exception bien évidemment des répertoires privés décrits précédemment.

1.3.3 Atribution et gestion quantitative des espaces et des quota

Le volume de données disponible pour chaque utilisateur doit être bornées par un quota pour au moins les raisons suivante :

- Cela permet de garantir les possibilité d'écriture globale. En effet, sans quota un utilisateur maladroit ou indiscipliné peut s'atribuer tout le disque et donc bloqués toutes possibilités d'écriture à tout les autres utilisateurs se trouvant dans la même partitions.
- Cela évite la colectionnite au niveau des fichiers
  Le système interdisant les écriture au bout d'un certain temps les fichiers qui peut-être serviront un jours seront éliminné naturellement et n'encobrerons plus le disque.
- Cela permet de limité a quantité de disque nécessaire
  Comme on place une limite au utilisateurs, il est possible de prédire la croissance de l'espace consommé est donc d'acheté moins de disque.
- Cela permet de facilité les sauvegarde
  Le volume consommé étant moindre le volume à sauvegardé diminu d'autant.

Bien évidement le problème principale posés par l'utilisation de quota est la fixation du niveau de ceux-ci. En effet, dans l'atribution d'espace il faut impérativement tenir compte du type d'utilisation de l'informatique qui est fait par l'utilisateur et donc disposé de quota différencier par classe d'utilisateur. Ces classes devront définit en fonctions des outils utilisés, du travail effectué et non en fonctions d'autres critéres comme par exemple hierarchique. L'administrateur doit aussi tenir compte du fait que la limitations d'espace n'est pas l'arme absolut contre l'utilisateur indicipliné mais plus un outils permettant de garantir à chacun une disponibilité d'écriture permanente.

L'atribution d'espace doit donc se faire en surbookant largement les disques et en faisant en sorte qu'aucun utilisateur ne puisse bloqué un disque. Cela est garantie en faisant en sorte qu'il n'existe pas de quota tel qu'un utilisateur dispose de plus de la moitié de l'espace libre des partitions ou il posséde un droit d'écriture. En dehord de ces grandes régles fixer des quota reléve surtout de l'expérimentation et de la connaissance que l'on a de ces utilisateurs. L'administrateur procédera donc en profilant ces utilsateurs et en atribuant les nouveaux arrivant à un profil.

Dans le cas ou l'on fixe des quota sur un système ou les utilisateurs n'en avaient pas il faut commencer par fixer des limites différenciés. Une premiere générique qui sera la limite vers l'aquelle on veut tendre. L'administrateur appliquera celle-ci à tout les utilisateurs se trouvant en dessous et il fixera aux autres une limite a peine supérieur à ce qu'ils consomment. Ensuite, il devra communiqué aux réfractaires un écheancier réaliste de diminution de leurs quota les contraignant à aléger leurs espaces tout en respectant leur mode de travail. Bien, évidement chaque risque d'iterdiction d'écriture devra être largement signalé à l'utilisateur par un dispositif automatique.

L'ors de la creation de quota il peut-être commis des erreurs qui font que le système ne bride pas du tout les ecritures. Dans ce cas il est impératif de ne pas le rendre opérationnel brutalement mais

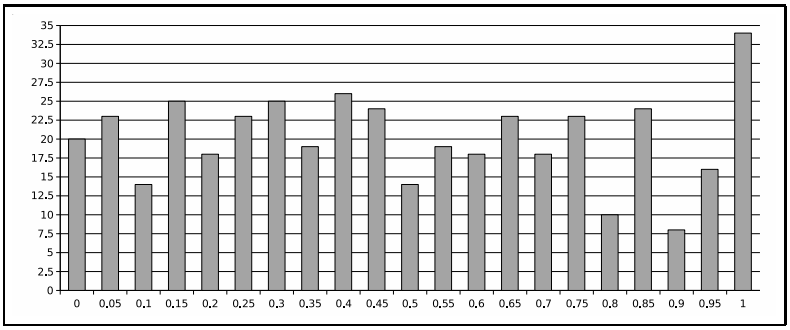


FIG. 1.3 – Distribution des quota réel pour les étudiant de l'isima. Le graphique représente l'histogramme des quotients d'utilisations des quota de disque.

de procédé comme décrit au paragraphe précédent. Le non respect des cette règle peut conduire à une catastrophe en terme d'heure de travail perdu.

La gestions des quota est une opérations dynamique ce qui implique qu'une politique à ce niveau ne peut-être fixés dans le temps. Les changements qui pourront affecter cette politique sont principalement :

- L'arrivé de nouveaux utilisateurs. Elle doit normalement être absorbé par un surbookage raisonnable des disques et une politique de réduction de l'espace ne peut-être qu'un cache misère dans l'attente d'achat de nouveaux disques. L'arrivée de nouveau utilisateur doit-être, autant que possible, planifié et intégré à la gestion courrante plutot que subit, via une réduction de quota, par les autres utilisateurs.
- L'arrivée de nouveaux moyens de stockage de données ou de sauvegarde
  Normalement, de nouveaux moyens permettent de destresser l'espace est donc d'attribuer aux utilisateurs plus de place, mais il faut bien avoir à l'esprit : que l'ajout d'espace ne doit ce faire qu'en ayant les moyens de sauvegarde corespondant et qu'il faut être bien sur du fonctionnement des nouveaux moyens avant d'atribuer les nouveaux espace.
- L'upgrade ou le déploiement de nouveaux logiciels
  Le passage d'un codage ASCII vers un codage UTF est un excellent exemple d'évenenement extérieur à l'entreprise qui sans actions particulieres de l'utilisateur conduit à un doublement de la taille de beaucoup de fichiers.

La surveillance des quota doit se faire par la construction d'histogramme d'occupation de disque par catégorie d'utilisateur. Cela donne des diagrame tel que celui de la figure 1.3 dont la lecture et l'évolution dans le temps permette une bonne gestion. Sur ce diagramme on note une distribution uniforme du taux d'occupation et une légere suroccupations dénoté par le fait que la situation la plus fréquente est proche des quota saturé.

L'évolution d'une tel situation doit être surveiller de près de façon à permetre une évolutions des choses vers une situation idéal tel que représentée sur la figure 1.4 au lieu d'une situation franchement problématique tel que représentée sur la figure 1.5

Le diagrame représenté sur la figure 1.5 nous montre une situation ou une majorité d'utilisateurs sont prés de l'occupation maximum ce qui dénotte forcément une situtation de stress. Il faut également remarqué sur ce graphique la bosse qui ce trouve autour du taux d'occupation de 0.5 qui montre que l'on a attribuer à deux classes d'utilisateurs distinctes les mêmes quota. Cette erreur est probablement

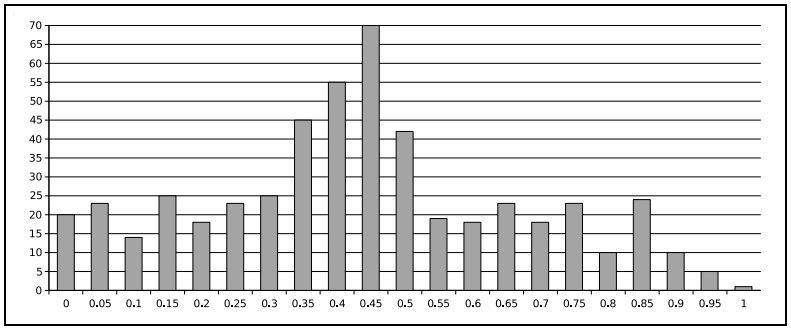


FIG. 1.4 – Distribution de quota idéal. Le graphique représente l'histogramme des quotients d'utilisations des quota de disque.

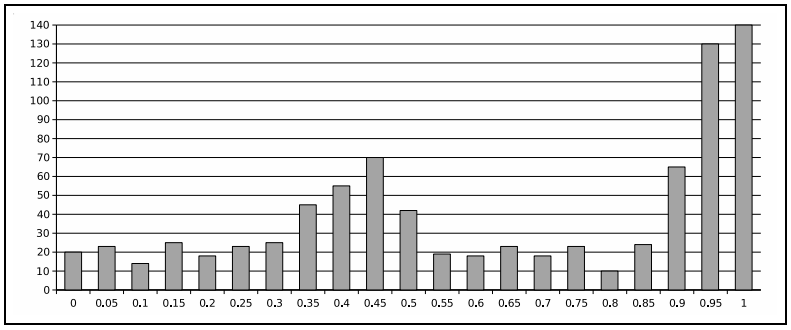


FIG. 1.5 – Distribution de quota trop petit. Le graphique représente l'histogramme des quotients d'utilisations des quota de disque.

la cause du stress observé.

1.3.4 Archive

Une politique d'archivage des données doit aussi être mise en place. Le but de cette politique est de débarassé le systèmes des données dont la disponibilité immédiate n'est pas nécessaire et qui ne doivent plus être modifiés. Afin d'être efficace l'archivage doit reposé sur un maximum de procédure automatique. En effet, cette tâche n'étant pas immédiatement productive et sa réalisation étant fastidueuse elle sera le parent pauvre de l'activité des informaticiens. D'autre part pour qu'une archive soit intéressante elles doit-être bien rangés et classé de façon systématique qualités qu'il est beaucoup plus facile d'obtenir en automatique que via une procédure manuel.

Les règles de constitution d'une archive sont les suivantes :

- La structure de l'archive doit être temporelle
  Le rangement des fichiers doit se faire dans des dossiers correspondant aux années puis correspondant aux divers projets. Un découpage inverse nuirait aux possibilités de découpage et ne poserait pas clairement les moments où doit être réalisée cette archive.
- Elle doit être indexée par thèmes, projets et mots clefs.
  Quand ils auront besoin des fichiers de l'archive, les utilisateurs n'auront qu'une information partielle sur ce qu'ils recherchent, certaines personnes auront quitté l'entreprise, la structure et les délais du projet auront été oubliés et, dans certains cas, une bonne indexation des archives évitera de réinventer le pneumatique.
- Elle doit être dupliquée sur des supports non réinscriptibles fiables.
  Les deux buts principaux d'une archive sont de disposer d'un historique et de réduire la quantité de données à sauvegarder. Il est donc bon d'utiliser des supports tels que CdRom ou DvdRom qui présentent une bonne durée de vie et ne peuvent être modifiés. Le stockage sur bande magnétique serait par contre une erreur majeure du fait de la faible durée de vie de ces supports et des temps d'accès qu'il génère.
- Le contenu doit être accessible via une procédure simple.
  A ce niveau, un serveur intranet est une solution. Il peut être réalisé à l'aide d'une machine peu puissante ayant simplement un grand espace disque ou, plus élégamment, via une tour de distributions de CD ou de DVD.
- Les fichiers archivés ne devront pas être compressés.
  Cela peut paraître étrange mais s'explique par le fait que la perte d'une partie d'un fichier compressé équivaut à sa destruction totale. Comme nous allons stocker les données pour longtemps, on aura probablement à gérer des dégradations partielles de supports qui conduiront à des pertes de morceaux de fichier.

Les utilisateurs devront soumettre les données à l'archivage via des répertoires particuliers ou en utilisant une commande permettant de soumettre un répertoire entier. Peu importe la méthode, mais elle devra être simple, documentée et connue. Il va sans dire que la fiabilité de la procédure d'archivage est critique. Une bonne façon de pousser les utilisateurs à archiver des documents est de limiter l'espace de données utilisateurs. La plus mauvaise façon de le faire est l'archivage autoritaire de documents non utilisés depuis longtemps

Une des grosses limitations des possibilités de réutilisation des archives est l'obsolescence des formats de fichiers du fait de l'évolutivité des logiciels. Il est donc souvent bon de recoder les données

dans un format différent (pdf ou postscript) ou plus simple (txt, html). Ceci induira certainement une perte de qualité mais augmentera la durée d'utilisation potentielle des documents. Nous pouvons ici regretté qu'il n'existe pas de format universellement reconnu d'archivage de documents autre que le papier. les normes en se domaine sont Norme NF Z 42-013 pour l'informatique et ISO 15489 pour les documents d'une maniére général.

Une fois votre système de fichiers réalisé, il convient maintenant de le sauvegarder correctement, ce qui est l'objet de la partie suivante.

1.4 Sauvegarde

Une équipe d'administration système doit garantir des conditions minimales de récupération de données aux utilisateurs. Or, dans beaucoup d'entreprises, la sauvegarde est le parent pauvre, que ce soit en matériel ou en temps. Cette négligence est le plus souvent due à une mauvaise estimation du coût de perte des fichiers.

Il faut donc admettre que les investissements au titre de la sauvegarde sont à comptabiliser avec les coûts d'assurance car, de la même façon qu'une assurance ne sert qu'en cas de sinistre, une sauvegarde ne montre son utilité que lors de perte massive de données. Si on examine les choses d'un point de vue humain, les responsables système sont rarement motivés pour cette tâche ardue, répétitive et non immédiatement productive. Les décideurs de l'entreprise devront donc prendre en compte ce fait en fournissant des moyens importants dans ce domaine et en valorisant ce type de tâche. Afin d'établir clairement les responsabilités, les sauvegardes devront être de préférence réalisées par une personne unique et, pour limiter les risques d'erreur, via un point unique du réseau. Il vaut mieux dépenser un maximum sur un système de sauvergarde permettant que sur N petits systèmes répartis dans N serveurs qu'il faudra surveiller et manipuler en permanence.

Toute politique de sauvegarde doit tenir compte du fait qu'il y a au moins cinq méthodes pour perdre des fichiers :

- Effacement accidentel par un utilisateur,
- Corruption volontaire par un acte de malveillance,
- Dysfonctionnement d'application, soit à cause de bugs, soit suite à l'infection par un virus,
- Panne matérielle du support de fichiers, panne de disque dur ou destruction accidentelle [2],
- Perte ou vol du support des données [3].

Toujours par analogie avec l'assurance, se prémunir contre l'ensemble de ces risques va demander des précautions multiples.

La question à se poser est ensuite "que va-t-on sauvegarder ?". A ce niveau on peut distinguer quatre types de données réclamant un traitement différencié :

- Les fichiers de bases de données
  Ce type de fichier ou de répertoire est en général d'une localisation et d'une taille aisément prévisibles. Le principal problème qu'il pose est lié à sa cohérence. En effet la plupart des systèmes de base de données ne sont exploitables que si les tables sont dans un état où toutes les écritures d'une même transaction ont été réalisées correctement. Si nous réalisons une sauve-

[2] les chaises de bureau adorent écraser les CD

[3] Les diskonkeys sont les derniers exemples de support facile à perdre

garde brute d'une base en cours d'utilisation, il existe un risque de corruption non négligeable qui peut rendre les données inexploitables. Il est donc impératif dans ce domaine de respecter les préconisations de l'éditeur du logiciel ou *d'arrêter la base*, c'est-à-dire de bloquer toute écriture.

Certains gestionnaires de base de données contiennent une fonction intégrée de sauvegarde. Il faudra l'utiliser en déclenchant le dump dans un répertoire convenu que l'on sauvegarde ensuite de façon classique. Une autre technique possible est d'utiliser des fonctions de réplication de bases. Il faut alors procéder en créant une réplique que l'on arrête et dont on sauvegarde les fichiers. Cette solution est de loin la plus sûre et limite énormément le temps d'arrêt nécessaire à pratiquer la sauvegarde.

– Les fichiers utilisateurs

Ceux-ci posent également de nombreux problèmes. Déjà certaines applications peuvent rester ouvertes et verouiller les fichiers. Demander ou provoquer la déconnexion des utilisateurs en fin de journée permet en principe de régler ce problème.

Le volume de données modifiées chaque jour par les applications modernes est par contre très important. En effet celles-ci ont tendance à stocker des paramètres d'utilisation du fichier dans celui-ci, comme par exemple les imprimantes utilisées ou simplement la feuille active dans un tableur. Il faudra également faire très attention au volume généré par les caches de navigateurs. L'utilisation de quota est un impératif si l'on souhaite maîtriser la quantité de données des utilisateurs pour pouvoir réaliser des sauvegarde sur un seul support.

Une des difficultés de ce type de données sera, si nous n'avons pas pris la précaution de réaliser un espace de données centralisé, de trouver les fichiers à sauvegarder sur chaque poste. Le seul moyen de s'en sortir sans recourrir à des usines à gaz logiciels sera de verrouiller au maximum les accès au disque local à l'exception d'un unique répertoire où l'utilisateur peut écrire.

Dans le cas où on créé un espace centralisé, il faudra cependant se méfier des logiciels qui enregistrent leurs données localement dans un répertoire propre. C'est un comportement très fréquent des outils de messagerie. Il faut donc faire en sorte de renvoyer chez l'utilisateur ces fichiers, et cela de façon transparente, avant de déclencher la sauvegarde.

– Les fichiers de systèmes d'exploitation et de logiciels

Il s'agit de tout le paramétrage de la machine. Ces fichiers sont le résultat du travail de l'administrateur. A ce niveau, il ne faudra sauvegarder que ce qui est modifié par rapport à l'installation originale. Dans ce domaine il faut être capable de reconstruire plus que de restaurer. Il faudra par contre prêter une grande attention à l'archivage des distributions originales.

Le rythme de sauvegarde dépend de l'activité du système. Il doit respecter les contraintes suivantes :

– Respect de la cohérence des fichiers
– Adapter, exhaustivité
– Coût de perte supérieur au coût de réparation
– Taille des modifications inférieure à la taille de bande

En général, une à deux sauvegardes par jour suffisent amplement, sauf en cas de rythme de travail important où les sauvegardes doivent être inférieures à une demi-journée. Dans ce cas, il faut envisager d'utiliser des système de disques RAID assurant une tolérance de panne.

La durée de la sauvegarde peut paraître anodine mais c'est en fait la principale limitation à ce qu'il est possible de faire. Les disques modernes ont une capacité énorme et leur remplissage se

fait de façon progressive. Ceci fait que l'on ne se rend pas bien compte du temps nécessaire à leur recopie. Il faut donc choisir un support et un calendrier de sauvegarde permettant de limiter le temps de sauvegarde au temps d'inactivité du système.

Pour le matériel il existe actuellement un grand nombre de système à base de support magnétique qui offre des performances très différentes. La partie **??** détaille les types de lecteur disponibles.

Dans ce domaine, il faut privilégier les systèmes ayant le meilleur débit de façon à réduire le temps de sauvegarde, quitte à devoir utiliser plusieurs supports pour une sauvegarde complète. Cela permettra en outre de réaliser rapidement les restaurations ce qui représentera un gain de temps d'exploitation important en cas de panne.

Les bon logiciels de sauvegarde doivent permettre une restauration conditionnelle des fichiers en étant capable de travailler en multivolume. En effet quelque soit le système que vous choisirez, il sera de toute façon trop petit un jour ou l'autre. Ils doivent être aussi capable d'éviter qu'une nouvelle sauvegarde efface une précédente et donc de reconnaître les supports via un label. Une bonne stratégie dans ce domaine est que le logiciel tente d'ajouter la sauvegarde sur le support sans toucher aux données que celui-ci contient.

Pour pouvoir restaurer un fichier bien longtemps après que l'utilisateur l'ait effacé, il faut bannir les sauvegarde snapshot que font certains administrateurs. Ceux-ci font une recopie exhaustive des données sur un autre disque en écrasant la copie précédente. Celà ne garantie que la panne matérielle du disque et encore, en espérant que celle-ci se produise à un moment où l'on est disponible pour empêcher l'effacement précédent la recopie du jour. Le plus souvent, le disque utilisé est, de plus, dans le même serveur que les originaux ce qui ne protègera rien en cas d'incendie ou de surtension électrique. Bref, c'est insuffisant.

L'utilisation de support amovible inerte ne comportant aucune partie électronique est la seule façon de se prémunir contre tous les risques décrits plus haut à condition bien évidemment de les stocker dans une pièce différente de celle où se trouvent les données originales. L'emploi d'une sauvegarde sur disque ne peut se justifier que par une création rapide de fichiers contenant les modifications du jour que l'on doit ensuite recopier sur bande. L'avantage est de disposer alors d'une copie permettant une restauration rapide et de s'affranchir du temps de sauvegarde à partir du moment où il ne dépasse pas la journée.

Le seul moyen d'obtenir des sauvegardes fiables est de chiffrer les riques de destruction et de définir un calendrier de sauvegarde décrivant précisément ce qui est sauvegardé et quand. Sa définition est l'objet de la partie suivante.

1.4.1 Calendrier de sauvegarde

Afin de limiter la quantité de données écrites chaque jour, il faut faire des sauvegardes incrémentales. Dans un premier temps, on réalise une sauvegarde complète pour ensuite ne sauvegarder que les fichiers qui ont été modifiés. Pour cela, on s'appuie sur l'heure de modification des fichiers, le système compare l'heure de chaque fichier avec une référence créée avant la sauvegarde complète et copie les fichiers plus récents. Bien évidemment pour qu'un restauration complète soit possible, il faut conserver l'ensemble des supports d'une série. Cette façon de procéder permet aussi d'utiliser au minimum le matériel et donc de prolonger sa durée de vie.

On classe les sauvegarde en niveaux par exemple de 0 à 10, ce qui signifie que le système ne

sauvegarde au niveau 1 que les fichiers créés ou modifiés depuis le niveau 0. Ce type de sauvegarde se fait par exemple avec ufsdump sous Solaris, dump sous Linux ou peut être programmé via un script sur les systèmes ne disposant pas de ce genre de possibilité en standart. Le système réalise des comparaisons de dates pour déterminer les fichiers à sauvegarder au niveau supérieur. Il faut donc porter une attention suffisante à la synchronisation des horloges sur les différents systèmes, faute de quoi, des oublis pourraient se produire.

Un plan de sauvegarde est par exemple :

semaine	1	2	3	4	1	2	3	4
L	1	3	3	3	1	3	3	3
M	4	4	4	4	4	4	4	4
M	5	5	5	5	5	5	5	5
J	6	6	6	6	6	6	6	6
V	7	7	7	7	7	7	7	7
S	8	8	8	8	8	8	8	8
D	9	9	9	9	9	9	9	9

Ce calendrier n'exploite pas le niveau 0 qui peut être gardé pour des sauvegardes exceptionnelles, par exemple avant un changement de système ou si les volumes à sauvegarder dépassent la capacité des supports que l'on a à sa disposition. Dans ce cas il faudra inclure une sauvegarde de niveau 0 que l'on fera en manuel et en restant à côté de la machine pour changer les cassettes si nécessaire. Les niveaux 2 et 10 sont quant à eux gardés en réserve pour une sauvegarde d'urgence en cas de risque de défaillance d'un disque ou d'autre panne que l'on sent venir.

Les changements de cassette dans le lecteur devront intervenir au début d'une série et, si nécessaire, à un niveau de sauvegarde fixe. L'idéal est de le faire une fois par semaine, ce qui n'est pas trop contraignant tout en permettant d'éviter une grosse catastrophe en cas de support défectueux. Quand le personnel manque pour changer les cassettes, comme par exemple lors des congés d'été, il faut confier cette tâche à un utilisateur sérieux ou diminuer la fréquence des sauvegardes de façon à pouvoir utiliser le même support plus longtemps.

Pour que les sauvegardes soient sûres, il faut avoir :

12	Jeux 1	N K7 Jeux
16	Jeux 3	1 K7 Jeux
10	Jeux 4 à 9	1 K7 Jeux

De cette façon nous conservons un an de sauvegarde tout en ayant une résolution d'une journée sur deux mois, une semaine sur trois ou quatre mois et de un mois sur l'année. Une autre précaution qui ne mange pas de pain est de stocker uniquement les cassettes du jeu courant dans la même pièce que la machine qui porte les disques des fichiers utilisateurs de façon à éliminer les risques de perte en cas d'incendie. Un certain nombre d'administrateurs dorment avec leurs cassettes de niveau 1 sous leur lit [4] c'est quelquefois une précaution valable mais ceci doit être en accord avec la direction de votre organisme.

[4] Certaines mauvaises langues disent que c'est pour vérifier leur présence en cas d'angoisse de nuit

Date	K7	Niveau	Notes	Opérateur
12/3/03	*DAT100*	*1*		
17/3/03	*DAT101*	*1*		
24/3/03	*NET2*	*---*	*Nétoyage normal du dat*	
24/3/03	*DAT102*	*1*		

FIG. 1.6 – Cahier de sauvegarde

Les cassettes devront être labélisées et le label devra être écrit magnétiquement sur la bande et sur une étiquette indélébile sur le boîtier et la boîte. Ceci permettra d'attraper rapidement la bonne cassette, de contrôler leur présence et donc de limiter les erreurs ou négligences.

Après la mise en place d'un système de sauvegarde, il est prudent de le tester en créant et effaçant quelques fichiers à des dates différentes. S'il est possible de retrouver ces fichiers, le système pourra être considéré comme fiable.

Associé au calendrier de sauvegarde, on doit faire un cahier[5] de sauvegarde indiquant les numéros de cassettes utilisées, les dates des changements, les sauvegardes exceptionnelles, les nettoyages de lecteur et les éventuels incidents en prenant exemple sur la figure 1.6. C'est une stupidité que de stocker ces données dans un fichier ou une base de données car elles pourraient se trouver justement sur le disque que vous essayez de restaurer.

1.4.2 Pratique de la restauration

La restauration de fichiers est une opération qui peut paraître simple mais elle est comporte cependant des risques qu'il ne faut pas négliger. Lors de ces opérations, il convient de respecter les règles suivantes :

–
– Après avoir restauré, retirez tout de suite la cassette du lecteur pour éviter que la sauvegarde suivante soit problématique ou qu'un utilisateur indélicat en profite pour en récupérer le contenu.
– Toujours restauré dans un répertoire différent de l'original de façon à éviter le remplacement de fichiers récents par des fichiers plus anciens. Un utilisateur verra d'un très mauvais oeil le fait de se retrouver avec exactement les mêmes mails qu'il y a un an suite à la récupération d'un fichier effacé.
– Aprés avoir restauré, vérifiez les propriétaires et droits des fichiers restaurés pour que leur exploitation soit possible, contrôlez qu'ils sont exploitables et ne comportent pas de virus.
– Si vous avez récupéré plusieurs versions d'un même fichier, demandez à l'utilisateur de choisir

[5]Il ne faut pas utiliser un classeur car les pages peuvent être perdues

celle qu'il veut.

Dans ce domaine, la plus grosse difficulté est souvent de déterminer à quelle date le fichier a été endommagé ou perdu. Cela est d'autant plus difficile que l'incident est ancien. Le dialogue avec l'utilisateur, stressé par la perte, se résume en général à :

ADM : Savez-vous quand le fichier a disparu ?
UTI : Euh, ben, il n'y est plus.
ADM : Bon, et qu'avez-vous fait pour qu'il disparaisse ?
UTI : Rien, rien du tout, il n'y est plus.

Il faut donc s'y prendre d'une manière différente. Premièrement, ne pas culpabiliser l'utilisateur de telle sorte qu'il ne panique pas et réfléchisse. Deuxièmement, il faut lui parler du contenu du fichier de façon à déterminer quand ce travail a été entrepris et modifié en s'appuyant sur l'agenda ou les notes manuscrites qui vont avec. Ensuite, il faut déterminer précisément la position et le nom du fichier. Puis prendre congé et aller rechercher tranquillement le fichier.

Avant de commencer à chercher dans les cassettes, il est souvent intéressant de rechercher le fichier en question dans les divers emplacements du système où l'utilisateur a accès. En effet, s'il s'agit d'une fausse manipulation, le fichier est tout simplement perdu et donc la version retrouvée sera la plus récente.

Un autre type de restauration auquel il faut prendre garde est la restauration de fichier de base de données. Il faudra en vérifier la cohérence et la qualité avant de remettre les fichiers à leur emplacement original de façon à ne pas reprovoquer encore des saisies inutiles.

1.5 Gestion des logiciels

1.5.1 Introduction

Les ordinateurs modernes utilisent de plus en plus de composants logiciels divers fournis par des constructeurs différents. La gestion de cet ensemble demande donc une grande rigueur de façon à limiter au maximum les pertes de temps.

Afin que l'ensemble reste gérable, l'administrateur système doit faire particulièrement attention à ce que les règles suivantes soit respectées :

- Il faut que les besoins des utilisateurs soient satifaits et que les outils choisis les mettent en autonomie. Par exemple, il faut éviter que l'administrateur ait à intervenir pour transcoder des fichiers.
- Il faut limiter le nombre de logiciels au maximum pour non seulement abaisser le coût mais aussi réduire le nombre de pannes possibles.
- Il faut éviter le doublonnage des fonctions entre logiciels. Si vous installez deux logiciels qui font la même chose, même s'ils exploitent les mêmes formats de fichier, cela compliquera toute upgrade. En effet, il n'est pas dit que les nouvelles versions resteront aussi compatibles et la disparition d'une des deux suites sera vécu comme un traumatisme par les utilisateurs.
- Il faut maintenir une homogénéité des versions dans toute l'entreprise. En effet, outre les échanges de données, gérer différentes versions conduit à une perte de temps non négligeable et augmente le nombre de bugs potentiels. De plus, toute mise à jour des systèmes d'exploitation

sera compliquée car il faudra tester la compatibilité avec toutes les versions ce qui demandera de nombreuses manipulations.

Le choix doit aussi être orienté vers des applications qui ne sont pas interdépendantes et il faut éviter les applications qui sont censées faire tout et qui en général ne font rien correctement. Par exemple, disposer d'un navigateur web indépendant du système d'exploitation améliore grandement la sécurité et rend le changement de l'un ou l'autre moins risqué. Sur les serveurs, il faudra se méfier des interdépendances qui se créeent à la longue. Cela se produit par exemple pour les outils de gestion de mailling list qui fonctionnent à partir d'une base de données qui sert également à gérer autre chose comme par exemple la paye. Dans un cas pareil toute mise à jour demandera un travail de test fastidieux et sera de toute façon risqué.

Une fois le choix des logiciels arrêté, il est important de construire un graphe de dépendance des différents outils. En effet, il faut apprécier quelle est la portée de cette dépendance avant d'entreprendre ces installations.

1.5.2 Mise à jour et déploiement de logiciels

Avant de déployer un nouveau logiciel, il faut prendre le temps de faire deux choses :

– Configurer correctement les options du logiciel.
  Il faut identifier les fichiers de données qu'il exploite et cacher ou verrouiller les fonctions inutilee. Dans les fichiers de données de l'application, il faut déterminer ceux que l'utilisateur doit pouvoir modifier de ceux qui doivent être verrouillés. Un exemple type est le cas des dictionnaires des traitements de texte. S'il est important que l'utilisateur les manipule, il est important qu'ils résident sur son compte sinon il faut les laisser à l'emplacement standart et les verrouiller. Pour beaucoup de logiciels, il faudra adapter la configuration par défaut à celle du site de déploiement. Par exemple pour un navigateur web, le proxy devra être configuré automatiquement.
– Former les utilisateurs et écrire une documentation
  Les applications modernes étant souvent très conviviales, la formation peut sembler un luxe inutile, mais c'est une betise de le croire. Les applications sont souvent très complexes et les utilisateurs ne les exploitent que très partiellement. Or, l'emploi d'une fonction inappropriée conduira forcément à une impasse qui se terminera dans le bureau de l'administrateur. D'autre part les utilisateurs n'exploitent quasiment jamais les possibilités d'automatisation des logiciels, comme par exemple les maillings ou le mode plan des traitements de texte, et passent un temps non négligeable à rechercher les fonctions dont ils ont besoin. Organiser de petites formations augmente par conséquent la productivité. S'il s'agit de formation courte, 1/2 journée par exemple, elles pourront être dispensées par l'équipe d'administration ce qui présentera en plus l'avantage de renforcer les contacts entre utilisateurs et administrateurs. Il est bon qu'il s'écoule un minimum de temps entre une formation et la mise à disposition du nouveau logiciel.
  La documentation des logiciels modernes est souvent de bonne qualité mais elle est plétoresque. Il faut compiler une petite documentation décrivant les fonctionnalités les plus utiles et la rendre disponible à tous par exemple via un intranet.

Dans le cas d'introduction d'un nouveau logiciel ou dans celui où l'on fait une mise à jour, il faut vérifier la check liste suivante :

– Compatibilité avec le matériel existant et particulièrement en ce qui concerne la quantité de ressources (CPU, mémoire) consommées.
– Compatibilité avec les systèmes d'exploitation utilisés
– Compatibilité avec les applications existantes et en particulier celles qui sont employées par toute l'entreprise
– Récupérer tous les correctifs existants

Dans le cas de mise à jour il faudra en plus veiller à :

– vérifier la qualité des échanges avec les versions précédentes ou avec l'outil précédent remplissant la même fonction.
– régler les problèmes de migration des configurations utilisateurs. Dans ce domaine, il faudra utiliser toutes les possibilités de configuration pour que les utilisateurs ressentent le moins possible de changement. Il faut aussi se méfier des petits outils annexes animés *cpuphages* qui ruineront tous vos efforts si vous ne les désactivez pas. Dans le domaine de la migration du paramétrage utilisateur il faut bien évidemment que tout soit automatisé car il n'est pas possible de compter sur les utilisateurs dans ce domaine.

Il faut ensuite choisir un calendrier et une stratégie de déploiement. Un administrateur doit comprendre que tout changement applicatif est vécu par les utilisateurs comme l'arrivée d'un chien dans un jeu de quilles. Il faut donc choisir un moment de calme dans l'activité de l'entreprise de façon à limiter au maximum le stress et l'accumulation de travail pendant le temps nécessaire au déploiement[6].

Dans le cas où des mises à jour de matériel ou de système d'exploitation sont nécessaires, il faudra si possible le faire indépendamment de l'application car même si cela induit un peu plus de travail pour l'administrateur, cela réduira les ennuis potentiels et le traumatisme pour l'utilisateur.

Il faut en particulier éviter de faire du déploiement le vendredi ou le lundi, les utilisateurs étant en général charette pour finir la semaine ou dans l'urgence de ce qu'il n'ont pu terminer. Le mercredi est en général un bon jour et laisse la fin de semaine pour régler les éventuels problèmes. Dans le cas où l'on utilise des serveurs d'application, il est facile de déterminer précisement le moment où une application est la moins utilisée.

D'un point de vue stratégique il faut procéder en pan entier. Il faut donc upgrader les suites logicielles (comme openoffice) dans leur ensemble et faire déterminer dans quel ordre des logiciels interdépendants doivent être traités.

Quand le système utilisé (unix par exemple) permet une centralisation des logiciels, ou quand on utilise des serveurs d'application ou encore des applications Web, il est possible de déployer une nouvelle application sur toute une entreprise et c'est de loin la meilleure stratégie. Mais dans le cas d'un déploiement poste par poste il est préférable de procéder par service ou entité homogène.

Le déploiement doit alors aller dans le respect du flux de données de l'entreprise car les versions sont rarement compatibles dans les deux sens. Il faut donc déterminer quel est l'ordre de manipulation des données (fig. 1.7) et remonter ce flux dans le cas où les logiciels présentent une compatibilité ascendante ou le descendre dans le cas inverse.

Bien évidemment, il faut sauvegarder l'ensemble des applications avant de se lancer dans un changement de façon à pouvoir rapidement revenir en arrière en cas de catastrophe. Pour rattraper rapidement les éventuels problèmes il faut aussi faire en sorte qu'une personne compétente soit rapidement

[6] A ce titre il ne faut pas suivre l'exemple de l'Agence de Modernisation des Universités qui ne tient aucun compte des calendriers universitaires pour le déploiement de ses releases

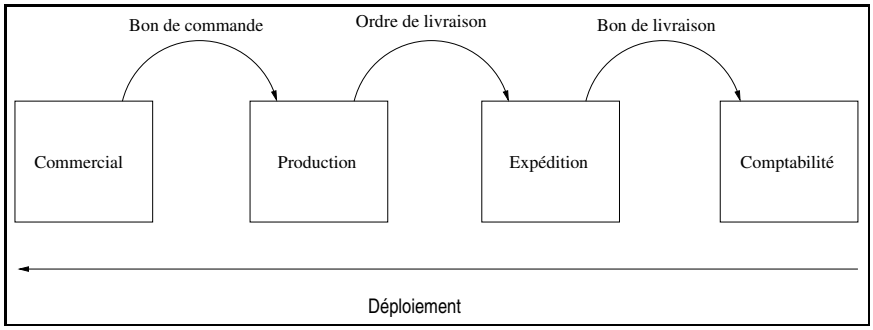


FIG. 1.7 – Stratégie de déploiement des applications en cas de compatibilité ascendante.

joignable.

Il faut également penser à rendre inutilisable l'ancienne version de l'application la où la nouvelle est déployée de façon a éviter l'encroutement qui conduira les utilisateurs à ne pas migrer vers la nouvelle version. Dans ce cas, l'administrateur risque de se retrouver à devoir maintenir les deux versions et à affronter de nombreux conflits le jour de l'arrêt effectif de l'application.

La fréquence des changements de logiciel est aussi quelque chose qui mérite une attention soutenue. D'une manière générale le changement est décidé soit pour des raisons intérieures à l'entreprise, comme par exemple de nouvelles méthodes de travail qui impliquent un nouveau développement, soit pour des raisons extérieures (évolution de matériel impliquant un nouveau système d'exploitation, ...). Dans le premier cas, la fréquence des changements sera un paramétre de fonctionnement de l'entreprise mais qui sera dans des bornes strictes. La qualité de la nouvelle application et sa ressemblance avec l'ancienne seront essentielles pour que les utilisateurs vivnte bien le changement. Trop d'applications sont encore à notre époque débuggées sur le bureau des utilisateurs. En cas de recours à un prestataire extérieur, les administrateurs devront être partie prenante de tout le processus de développement.

Pour les logiciels génériques, comme les clients de messagerie ou les navigateurs Web, l'administrateur dispose d'une grande liberté de choix quant à leur mise à jour. Cela dit le changement est la plupart du temps impulsé par :

- l'échange de fichiers sur le réseau avec d'autres entreprises
- la disponibilité sur le marché des anciennes versions
- l'évolutivité des fonctions
- les changements de systèmes d'exploitation qui eux-mêmes sont impulsés par les changements de matériel.

Il faudra donc changer de toute façon, un jour ou l'autre, et il ne faut donc pas trop attendre. En effet, si une application devient franchement incompatible avec le reste du monde, elle génèrera une perte de temps importante pour les utilisateurs et une charge importante d'administration pour recoder proprement les multiples formats de fichier posant problème.

A l'inverse, si les changements sont trop fréquents les utilisateurs vont se détourner de l'utilisation de l'outil et il ne sera pas possible de les former correctement. De plus, quand l'administrateur suit de trop près les dernières versions il se trouvera en permanence confronté à des bugs qui seront exterminés dans des releases mineures. Il faut donc savoir attendre que les fanatiques de la modernité essuient les plâtres pour vous.

Dans tous les cas, il convient d'être prudent car un changement de logiciel modifie les méthodes de travail des gens et donc les ennuis des services informatiques commencent le plus souvent par un changement d'application mal vécu.

1.5.3 Licence et coût des logiciels

Les coûts logiciels sont une charge importante pour les entreprises et ne peuvent être réduits que par l'emploi de logiciels dit libres.

Ces derniers offrent des coûts d'achat nuls mais obligent à une vision artisanale qui consiste à faire beaucoup de choses soit même. Ceci fait que les coûts de possession sont mal connus et peuvent être mal maitrisés. Cela dit, quand on multiplie le coût d'une licence d'une suite bureautique commerciale par le nombre d'utilisateurs, toute entreprise de taille moyenne aura le temps de payer quelqu'un à temps plein pour gérer les éventuels déboires. Cependant, les logiciels libres restent à l'opposé de la culture de nombreuses entreprises qui préfèrent maitriser les coûts dès le départ en incluant licence, formation et maintenance.

Cassons tout de suite un mythe : les logiciels commerciaux ne sont pas forcément mieux que les logiciels gratuits, mais ils ont en général un meilleur packaging et une documentation de meilleure qualité. Cela dit, les logiciels commerciaux ne sont pas exempts de bugs et, à la moindre release, il faudra se méfier des abandons de fonctionnalité ou des bugs croisés. Pour ce type de logiciels, on dispose d'une license par personne utilisant le logiciel, sauf disposition contraire du contrat ou négociation de licence site avec le fournisseur. Il est important de bien lire les contrats et de les faire respecter sur le parc que l'on gère. Le piratage de logiciel est une action qui peut conduire aussi bien au tribunal civil que pénal et non une petite combine entre amis évitant de dépenser quelques argents. D'un autre côté, certains éditeurs de logiciel abusent de leur position de fournisseur exclusif en faisant payer des contrats de maintenance qu'ils tourneront en arrêtant le produit en question et en fournissant une nouvelle gamme de produits de nom différent. Pour éviter cela il faut, quand on achète un logiciel et une maintenance, forcer le distributeur à inclure la fourniture d'un produit équivalent en cas de disparition du produit.

Pour la distribution des licences de logiciel,il y a deux cas possibles : soit le produit n'est pas protégé par une clef et le nombre de licences se compte par poste configuré pour faire marcher le logiciel, soit le produit est protégé et il existe un système de distribution de jetons d'utilisation.

Dans ce dernier cas, il est très important de configurer et surveiller correctement le serveur de licences de façon à éviter toute pénurie qui bloquerait les utilisateurs. Un bon gestionnaire de jetons, tel que flexlm, permet de gérer un grand nombre de situations tels que les jetons multimachines ou multiarchitectures. Dans ce cas, les précautions à prendre sont :

- Le serveur de licences doit être joignable et sa charge doit être prévisible et maitrisée.
- Une application de gestion des jetons doit être accessible via le réseau pour arbitrer les conflits.
- Les accès au serveur de licences vers l'extérieur doivent être sécurisés pour éviter les vols de jetons.

Les très bons gestionnaires sont capables de faire de la redondance c'est-à-dire de proposer un système de secours permettant d'éviter du flux l'étiage de licences. En général ceci demande que trois serveurs soient activés et que deux soient joignables pour éviter que les petits malins coupent leur réseau en deux.

Dans ce genre de système, le nombre de licences peut être inférieur au nombre d'utilisateurs théoriques puisque l'on a acheté un droit d'usage simultané et donc seules comptent les licences actives. Cela dit, le système de licences décompte en général l'usage de*ès que l'application est ouverte et cela même si l'utilisateur est parti dragué à la machine à café. La tentation de surbooker fortement et donc de limiter le nombre de licences achetées doit être tempéré par la nécessité de conserver un fonctionnement transparent du système. Pour des logiciels très chers utilisés peu souvent, comme les logiciels de reporting, gérer la pénurie peut être parfaitement gérable à condition de le faire en accord avec les utilisateurs. La bonne stratégie est alors de définir des créneaux horaires d'utilisation et qu'un utilitaire les rappelle en cas de lancement hors des périodes autorisées. Pour arriver à un fonctionnement harmonieux d'un tel service, il faudra bien évidemment surveiller que les fins de période soient respectées et, de toute façon, ne pas être trop gourmand.

1.6 Réseaux et organisation des comunications

Les réseaux permettent de relier entre eux les ordinateurs de l'entreprise et éventuellement d'accéder à d'autres réseaux tels que l'Internet. Cet élément va donc jouer un double rôle : celui d'élément de fonctionnement des applications réparties et celui de support de communication. Ces deux choses sont trop souvent confondues bien qu'elles aient des implications très différentes sur le fonctionnement de l'entreprise :

- La coopération d'ordinateurs pour mener à bien une même tâche est une fonction transparente aux utilisateurs. Dans ce cas, le système doit être considéré comme un seul ordinteur et non comme un ensemble de machines indépendantes. Cela signifie en particulier qu'une défaillance peut induire une panne totale du système et donc que la disponibilité du média de communication devra être constante tant en délai qu'en débit. C'est le domaine des applications clients-serveur.
- La communication entre machines, par email ou autre, est un processus coopératif entre utilisateurs. Elle peut donc être portée par des réseaux de débit et de mode de connexion variables. Par exemple, la lecture des emails est un processus asynchrone de leurs envois et cela ne pose aucun problème aux utilisateurs.

Le deuxième type a été conçu pour s'accommoder de réseaux de débits, délais et modes de connexion variables alors que le premier, même s'il exploite des protocoles de transport identiques, demande à respecter des contraintes strictes pour fonctionner correctement. En effet, les ordinateurs ont des difficultés à tolérer l'asynchronisme alors que les utilisateurs s'y adaptent très bien sauf en ce qui concerne, bien évidemment, l'interactif.

Pour architecturer un réseau d'entreprise, il sera important de gérer différemment trois types de trafic :

- Le trafic interactif, c'est-à-dire la communication entre terminaux et serveurs comme par exemple l'utilisation des protocoles de bureau virtuel (X11, RDP ou ICA) ou le trafic telnet.
- Le trafic système, c'est-à-dire les services de nomage, d'authentification, de fichiers et les applications clients-serveur comme les SGBD.
- Le reste qui est principalement constitué des protocoles utilisés sur l'Internet, mail, web ou ftp.

Le tableau suivant donne une idée des contraintes de débits :

	Délai	Débit	Connexion
Interactif	$\frac{1}{25}$ s	constant	permanente durant une session
Système	0.8 s	constant	permanente
Internet	30 s	quelconque	quelconque

Les choix d'architecture des réseaux devront être guidés par le respect des contraintes liées à ces trafic et il apparaît évident que les conditions nécessaires aux deux premiers types de trafic ne pourront être garanti que sur des résaux locaux. Pour garantir la qualité de ces trafic, il faudra éviter de leur faire traverser de nombreux appareils réseaux et se méfier des regroupements de trafic sur des lignes à très haut débit mais ne prenant pas en compte les notions de garantie de bande passante.

Un administrateur suspicieux pourra aussi séparer les trafics interactifs et systèmes en utilisant des machines à deux cartes réseaux et des commutateurs réseaux.

Pour découper son réseau, il est possible d'utiliser des vlans qui sont des réseaux cloisonnés par logiciel sur des commutateurs. Il sont appelés virtuels car ils se comportent comme des réseaux physiques distincts alors qu'ils sont localisés sur des appareils physiques communs. En pratique, cela revient à supprimer tout dialogue de niveau 1 entre certains ports d'un commutateur.

Il ne faut pas, par contre, considérer toute liaison empruntant l'Internet (via des VPN[7] comme ayant un débit fixe et donc les trafics systèmes et interactifs devront en être bannis. Les administrateurs considèrent trop souvent que, une fois les problèmes de sécurité réglés, les VPN sont un prolongement de leurs réseaux locaux et oublient que le débit et les délais peuvent terriblement se dégrader et conduire à un blocage de serveur.

La gestion d'un réseau devra se faire avec précaution et est un problème très différent de l'architecturation de celui-ci. Afin d'éviter d'avoir un truc ressemblant au Centre Beaubourg à gérer, il faut être rigoureux et observer les règles suivantes peut aider :

- établir un plan d'adressage et noter rigoureusement les adresses en utilisation,
- avoir des règles de nommage rigoureuses et les respecter,
- identifier toutes les liaisons et connaître les dépendances,
- disposer de plans et de documentations complêtes sur papier en particulier en ce qui concerne les appareils actifs.

Du fait des connexions extérieures le réseau peut être un élément majeur d'insécurité informatique. Il est donc raisonnable de lui donner une architecture sécurisée où les rôles et les droits de chaque utilisateur soient clairement définis.

1.7 Rôle des serveurs

Le vocable serveur désigne en fait un grand nombre de choses différentes. En particulier :

- des programmes tournants dans une machine comme par exemple le programme qui rend accessible un sgbd,
- des ordinateurs ayant un rôle central dans un système informatique, un serveur de fichiers par exemple,
- des machines ayant un rôle de distribution interactif qui se nomme serveurs d'apllications.

[7] Il s'agit de réseaux virtuels qui sont des tunnels logiques empruntant un réseau public

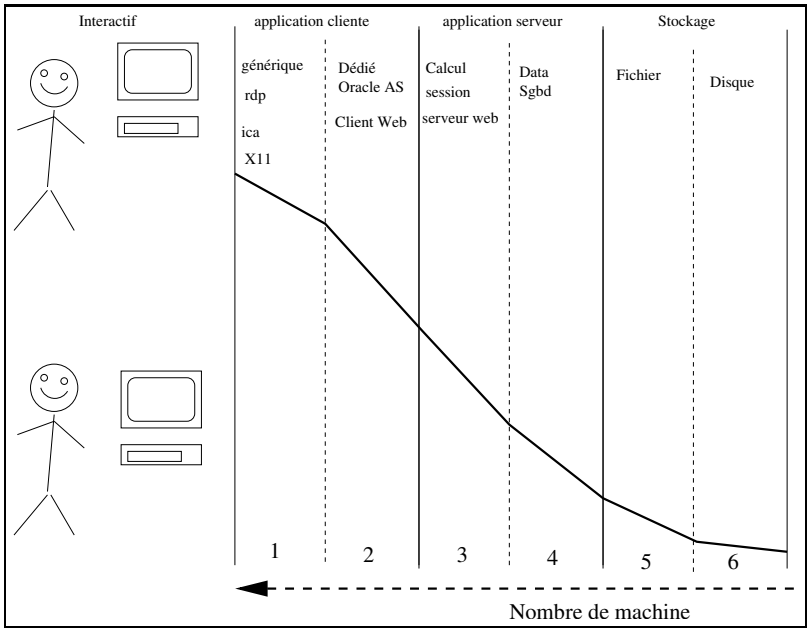


FIG. 1.8 – Couches de fonctionnalités logicielles séparables

Ces trois types s'incluent dans un schéma général qui est celui du modèle clients-serveur. Dans la pratique, tout le problème est de savoir combien de machines sont impliquées dans une application informatique et où se situe la frontière entre elles. Un service est constitué à un bout d'octets dans des fichiers et à l'autre bout d'un humanoide. Ce dernier est en général muni d'un poste de travail et tout le problème est de savoir ce qui tourne dedans et ce qui doit être ailleurs.

Il existe deux extrêmes :

– Les systèmes à base de maxis ordinateurs où tout est centralisé sur une machine et où les terminaux ont un rôle basique.
– Les micro-ordinateurs où tout est gérer en local et il n'y a pas de serveur.

La première approche pose le problème du coût et de la puissance nécessaire à obtenir un interactif riche. La deuxième pose le problème de la maintenance et du travail sur une information commune. Les systèmes clients-serveurs permettent d'apporter la richesse des fonctionalités tout en centralisant les données. Actuellement nous assistons à la rebanalisation des postes clients via l'intervention de pléthore de serveurs ayant chacun un rôle plus limité. En fait chaque ordinateur apporte sa brique à l'édifice commun qu'est le système d'information.

En pratique cela correspond au schéma de la figure 1.8. Il est donc possible d'avoir jusqu'à six briques impliquées dans une tâche et chacune peut être localisée dans une machine différente. Ce genre d'organisation est appelé modèle Ntiers et présente l'avantage d'avoir de grandes possibilités d'évolution d'échelle. Les autres avantages sont que chaque machine étant dédiée à une tâche précise, elle est facile à maintenir et qu'il est aisé d'obtenir des performances élevés notamment à forte charge.

Pour réalisé ce genre de découpage, les administrateurs doivent avoir comme but de concevoir un système protégé contre les fausses manipulations d'utilisateurs et où les tâches de maintenance et de déploiement de logiciels sont réduites au minimum.

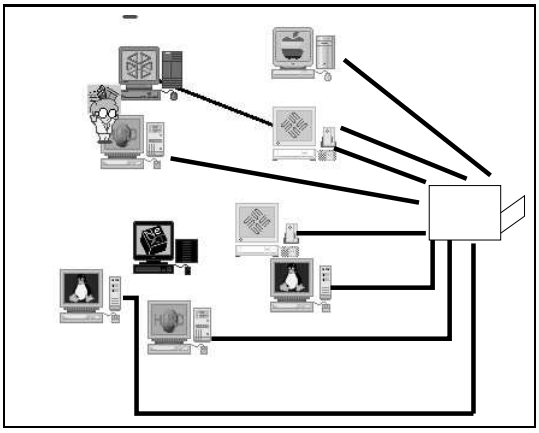


FIG. 1.9 – Sans serveur d'impression centralisé.

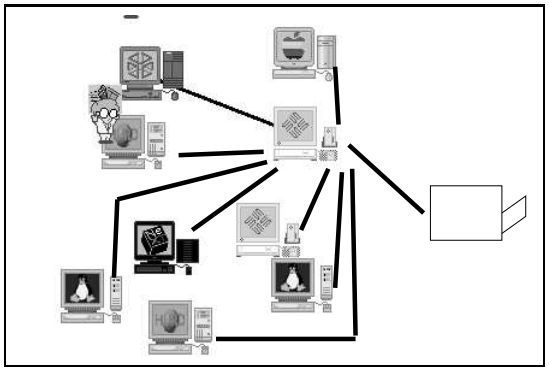


FIG. 1.10 – Avec un serveur d'impression centralisé.

Quand il est possible de déployer ce genre d'architecture, les postes clients doivent être les plus banals possibles de telle sorte qu'ils ne réclament que peu de maintenance. Les PC sont donc à proscrire sauf en cas de besoin particulier. Par exemple des utilisateurs faisant de la PAO ou de l'imagerie ont besoin d'un débit de données très important entre la CPU et l'affichage, il est donc malvenu de les mettre dans le panier commun car non seulement cela leur fera perdre du temps mais en plus cela risquera d'induire des pointes de charge importante sur les serveurs d'applications. Il en va de même pour les développeurs qui en plus risquent de générer des crashs par allocations trop violentes de mémoire.

L'idéal est l'utilisation d'applications Web qui ne demandent plus aucun déploiement. Un moyen terme est l'utilisation de clients légers (X11, RDP, ICA,...) qui permettent d'afficher tous les systèmes d'exploitation standart (UNIX, Windows, OS400).

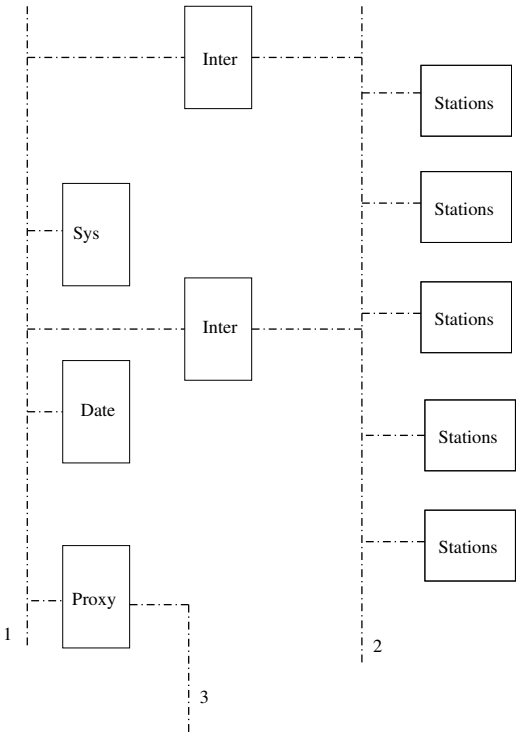
Il faut aussi essayer de créer en petit nombre des points de regroupement pour le stockage des fichiers et le dialogue avec des appareils de natures différentes. Le service d'impression est un modèle du genre. Si la configuration est celle de la figure 1.9, une panne totale d'imprimante est en théorie impossible, mais chaque panne sera difficile à régler car il faudra déterminer à la main quelle machine l'a provoquée. Si l'architecture choisie est celle de la figure 1.10, le serveur d'impression gardera la trace du travail qui a bloqué l'impression et on pourra réparer simplement en faisant les déductions suivantes :

- Si un ordinateur n'imprime plus sur une imprimante c'est lui qui est en cause,
- Si toutes les machines n'impriment plus sur aucune imprimante, c'est le serveur d'impression,
- Si aucun ordinateur n'imprime sur une imprimante donnée mais qu'il est possible d'imprimer sur une autre c'est elle qui est en panne.

La solution centralisée offre en plus l'avantage de faciliter les reconfigurations ou les remplacements d'imprimante et si une solution de comptage de pages doit être mise en place, cela se fera sans douleur. Cet exemple montre parfaitement qu'un choix de centralisation, qui introduit forcément un risque de panne total, s'avère par contre gagnant sur de nombreux points.

Cela dit, pour qu'une informatique en modèle n-tiers centralisé fonctionne bien, il est important de respecter certaines règles :

- Il faut dupliquer les services fondamentaux.



Légen de :
INTER Serveur d'applications
DATA Serveur de Sgbd
SYST Serveur de services systèmes, fichiers, ...
PROXY Serveur mail, web, proxy web, routeur

FIG. 1.11 – Modèle de système Ntier.

- Il faut regrouper sur un même serveur les fonctionnalités d'un même service.
- Il faut limiter le nombre de fois où une même pièce est présente.
- Il faut que chaque élément soit d'une taille suffisante.

En pratique, l'utilisation du modèle suivant donne satisfaction. On distingue trois groupes de serveurs que l'on répartit sur trois réseaux physiques (ou vlans) distincts. Les trois groupes de serveurs sont :

- les serveurs de services systèmes et de données qui n'ont accès qu'aux réseaux systèmes et ne sont par conséquent pas accessibles aux utilisateurs.
- les serveurs d'applications qui préparent l'interactif destiné aux postes clients. Ils assurent le dialogue entre ces dernières et les serveurs systèmes en ayant une patte sur les réseaux systèmes (1) et interactifs (2).
- les serveurs de type proxy qui assurent le dialogue avec l'extérieur et qui ont une patte sur le réseau système (1) et une sur le réseau extérieur (3).

Du fait que l'on utilise trois réseaux distincts, seuls les trafics autorisés peuvent être établis, ce qui limite de fait les possibilités de fausse manipulation. En fait, ce type de solution marche d'autant mieux que l'architecture choisie permet d'éviter le mélange de l'interactif et du trafic système.

Dans ce genre d'informatique très centralisée, le risque principal est lié au crash des serveurs principaux et aux interdépendances mal maîtrisées. Il faut donc surveiller attentivement les serveurs et bien connaître les services que l'on utilise.

1.8 Métier (Le Gourou / Le Technicien / L'Idiot)

Le Gourou : Personne hyper technique

Le Technicien : il fait en sorte que les autres ne voient rien

L'Idiot : il n'a pas le goût à ça

L'administrateur système est le garant du bon fonctionnement du système informatique de l'entité qui l'emploie, il doit également assumer une tâche de conseil et de communication envers ses administrés, assurer la configuration correcte du système et des applications. Il sera aussi appelé à optimiser la configuration du système pour obtenir les meilleures performances possibles.

1.8.1 Le conseil aux utilisateurs

Non seulement il faut être capable d'assurer le conseil aux utilisateurs, en s'efforçant d'être complet et compréhensible de façon à ce que les questions ne deviennent pas récurrentes.

Il y a deux types de situations très fréquentes et souvent conflictuelles :

- La fausse panne, c'est-à-dire que l'utilisateur ne sait pas faire marcher un outil et assimile un refus de l'outil à un dysfonctionnement[8]. Dans ce cas, il ne faut pas s'énerver et expliquer au mieux le problème et la façon de s'en sortir. Il est hyper important que l'utilisateur n'ait pas l'impression d'être pris pour un idiot sinon l'explication ne passera pas et la question reviendra.
- L'incompréhension par rapport à des habitudes (bonnes ou mauvaises) prises ailleurs ou dans l'utilisation d'un outil obsolète.

Pour éviter ces deux problèmes, il faut rédiger un manuel d'utilisation (en HTML par exemple) et en distribuer un résumé papier à chaque personne.

Pour maintenir une bonne ambiance, il faut toujours être aimable, exiger que les utilisateurs soient polis et traiter tout le monde de la même façon.

1.8.2 L'optimisation du système (le tuning)

L'optimisation consiste à faire marcher le mieux possible et le plus vite possible le système en jouant sur divers paramètres.

Dans ce domaine tout est bon, à condition de respecter les règles précédentes. On peut par exemple :

- Jouer sur la taille des serveurs
- Jouer sur la séparation des fonctionnalités
- Limiter les erreurs systématiques (recherche dans le PATH)
- Réduire la bande passante du réseau en dupliquant certaines informations

Faire attention dans ce domaine aux faux-semblants.

[8] Cas typique d'un éditeur qui diagnostique qu'un fichier à été ouvert deux fois et refuse donc de faire les sauvegardes.

1.8.3 La déontologie

Un ingénieur est un médecin et un curé à la fois. Il doit :

- Respecter le secret des informations
- Respecter la vie privée
- Décrire et expliquer ce qu'il fait. (J'ai arrêté ton process parce que ...)

Chapitre 2

Quelques informations pratique

2.1 Gestion des grands sites (où comment nager ...)

2.1.1 Qu'est-ce qu'un grand site ?

Une définition officielle est donnée par le SAGE. Pour eux, un grand site est une entité où il y a plus de 10 machines ou plus de 10 utilisateurs avec la règle suivante :

$$Nbmachine * NbUtilisateur > 100$$

Ma définition personnelle et donc officieuse est : un grand site est une entité telle que :

$$\sqrt{Nbmachine * NbUtilisateur * Nbarchitecture * Nbprofilutilisateur} > 500$$

Là, le bricolage n'est plus tolérable sinon le risque de devenir fou devient trop grand. Toute la difficulté de gestion de tels sites revient au choix entre centralisation et répartition

Les sections suivantes donnent quelques pistes pour s'en sortir.

2.1.2 Identification des utilisateurs

Ceux-ci doivent être exactement recensés par un serveur. Aucun accès anonyme sur les moyens informatiques internes ne doit être toléré.

– fournir un identificateur unique, un seul mot de passe pour que la gestion en soit simplifiée.
– les informations collectées permettent d'informer tout le monde.

Il y a deux types de système :

les systèmes à base de RPC NIS/YP ou NIS+ pour une version plus récente et sécurisé pour le monde UNIX. Les système à bases d'identification NETBEUI (sam) pour les systèmes NT ou suivant la norme LAN/MANAGERS.

En cas d'hétérogénéité, des outils comme SAMBA, ou YPkit de NCD ou tout autre système basé par exemple sur des annuaires LDAP permettent de maintenir une base unique dans les deux, même si elle est répartie.

Le service de nom ne doit tolérer aucune défaillance, il faut donc qu'il soit redondant :

– pour NIS/YP il faut un Master server et au moins un Slave server ne serait-ce que pour des raisons de performances.
– Pour les bases SAM de NT au moins un Contrôleur principal (PDC) et un Contrôleur secondaire (BDC) pour limiter les risques de perte d'information en cas de crash définitif du serveur. En effet si sous UNIX les UID sont donnés par l'utilisateur, ce n'est pas le cas sous NT. La perte d'un utilisateur signifie donc un grand nombre d'ACL à refixer, car bien évidemment NT stocke les numéros d'utilisateurs dans ces dernières.

La gestion des licences est assimilée à la gestion d'utilisateurs c'est-à-dire qu'il faut garantir que les logiciels sont utilisés en conformité avec le nombre de licences dont on dispose.

2.1.3 Partage disque

Ils sont nécessaires afin de permettre les sauvegardes et un accès transparent à tous les espaces utilisateurs. L'utilisateur doit disposer d'un espace disque unique associé au login (le home ou home-dir).

Plusieurs techniques permettent cela :

– NFS pour UNIX / NT
– SMB pour NT / W9X
– Appleshare pour MAC OS

NFS repose sur le protocole UDP/IP et permet de transporter des fichiers sécurisés par UNIX. C'est un protocole non connecté, il est donc contraignant mais très sûr et peu consommateur de ressources, c'est-à-dire que le serveur ne garde pas trace des fichiers ouverts et ignore les verrous classiques UNIX. Les échanges se font donc via des ordres d'écriture de blocs dans les fichiers et les verrouillages se feront via un processus (nfslockd) particulier qui s'assure auprès des clients du maintien des verrous et redemande les verrous existants lors de son démarrage, après un crash par exemple. Il fournit une sécurité par site ce qui limite son utilisation à des réseaux de confiance.

SMB permet une gestion de sécurité façon NT, donc très complète. C'est un protocole connecté, ce qui signifie que le serveur garde une trace des fichiers ouverts et des verrouillages. Le plantage d'un client avec un verrou actif sera donc plus problématique que sous NFS.

– Le choix des serveurs se fait en fonction des clients. Si le parc est hétérogène, le choix UNIX peut s'imposer.
– Les utilisateurs doivent être regroupés par thèmes et par durée de vie dans le système.
– Leur espace disque doit porter leur environnement de travail pour assurer la transparence. Il faut limiter l'espace utilisable par chaque utilisateur par des quotas (réalistes). On peut surbooker les disques dans le cas où les consommateurs sont ponctuels et hétérogènes. Le but n'est pas de gêner les utilisateurs mais d'éviter tout blocage accidentel.
– Il faut limiter autant que possible le nombre de serveurs (un si possible) pour limiter le nombre de manipulations nécessaires.

2.1.4 Configuration et hétérogénéité :

Il faut gommer les différences d'architectures au maximum, et ne conserver que les différences utiles à l'utilisateur. Concrètement, on procède de la façon suivante :

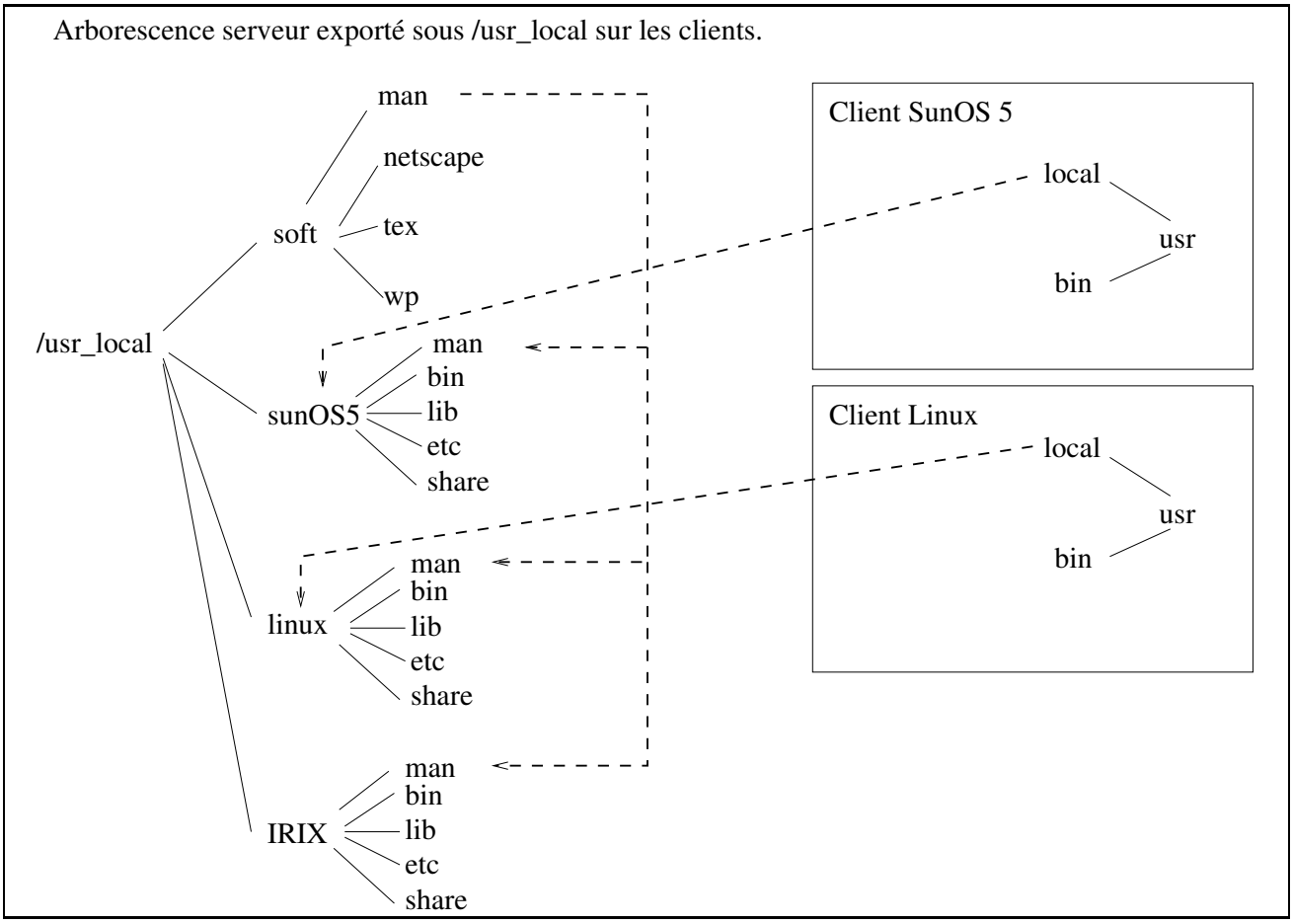


FIG. 2.1 – Arborescence logiciel en réseau sous UNIX

- standardisation des noms et des répertoires,
- standardisation des outils, les logiciels doivent être présents partout ou aliasés pour se lancer sur une architecture qui les supporte,
- standardisation des commandes, utiliser par exemple des versions de ps prennant les mêmes options,
- dans la présentation, configurer les bureaux pour leur donner un aspect similaire.

Le plus simple est de regrouper l'arborescence des logiciels sur une machine UNIQUE, puis d'en dupliquer automatiquement les morceaux par architecture si cela s'avère nécessaire. Ceci est facile à gérer sous UNIX mais quasiment infaisable sous NT.

On crée une partition /usr_local que l'on exporte en NFS sur notre réseau. On prend bien soin de séparer les fichiers binaires propres à une architecture des fichiers communs à toutes les architectures. On met les premiers dans /usr_local/'arch'/ et les seconds dans /usr_local/soft/. On monte cette partition en réseau sur toutes les stations et on délivre la bonne version en /usr/local à l'aide d'un lien symbolique. Cette structure est illustrée par la figure 2.1.

Remarque : L'intérêt de faire un lien symbolique de /usr_local/'arch' plutôt que de faire un montage NFS est qu'en cas de problème, on peut supprimer temporairement le lien sans trop perturber le fonctionnement des machines. Dans ce cas, les logiciels ne sont plus disponibles et on peut donc faire

toutes les manipulations nécessaires dessus. De plus, dans cette configuration, on ne monte qu'une seule partition et par conséquent on simplifie énormément son fichier d'export (ex : /etc/exports ou /etc/dfs/dfstab) et son fichier de montage. Un autre avantage est que tous les liens symboliques dans /usr_local sont conservés.

On réalise la configuration des utilisateurs comme suit :

En shell façon Bourne — Fichier ~/.bashrc

```
#############################################
#      .bashrc utilisateur lamboda          #
#############################################
#site
if [ -r /usr/local/etc/shnoa ]; then
  . /usr/local/etc/shnoa
fi
#architecture dependant
if [ -r /usr/local/etc/shrc ]; then
  . /usr/local/etc/shrc
fi
#machine
if [ -r /etc/shrc_commun ] then
  source /etc/shrc_commun
endif

#initialisation personnelle

lut="$HOME/lut "
```

En shell façon Cshell — Fichier ~/.cshrc

```
#############################################
##     .cshrc utilisateur lamboda          #
#############################################
#site
if ( -r /usr/local/etc/cshnoa ) then
  source /usr/local/etc/cshnoa
endif
#architecture dependant
if ( -r /usr/local/etc/cshrc ) then
  source /usr/local/etc/cshrc
endif
#machine
if ( -r /etc/cshrc_comm un ) then
  source /etc/cshrc_com mun
endif

#initialisation personnelle

setenv lut $HOME/lut
```

Une configuration utilisateur est donc réalisée à l'aide d'un fichier utilisateur qui appelle une fichier indépendant de l'architecture (fichier shnoa et cshnoa) puis un fichier propre à l'architecture (fichier /usr/local/etc/shrc ou /usr/local/etc/cshrc) puis un fichier propre à chaque machine (/etc/shrc_commun ou /etc/cshrc_commun). Les utilisateurs peuvent placer leurs initialisations personnelles à la fin du fichier.

On peut donc ainsi configurer par Site / Architectures et Machines, ceci permet de gérer plus facilement les changements et minimise les risques d'erreur utilisateur se transformant en question insoluble.

Grâce aux jeux de répertoire et à ces fichiers, on peut faire en sorte que les variables (exemple PATH = /usr/bin :/usr/local/bin) soient les mêmes quelles que soient les machines du site.

Pour NT, il n'y a pas vraiment de solutions à l'installation réseau de logiciels même si le système fournit un export de partitions très satisfaisant. Le problème est que les logiciels NT ont besoin de trois choses pour fonctionner correctement :

- un programme exécutable
- des entrées de bases de registre
- des librairies dynamiques (DLL)

Les wizard d'installation copient les fichiers exécutables, réalisent les inscriptions dans la base de registre et copient les fichiers de librairie dynamique nécessaires mais ne le font que sur la machine

locale. Pour un fonctionnement parfait, on sera donc conduit à réaliser l'installation sur chaque poste, car si il est possible de copier les exécutables et les DLL, il sera impossible de réaliser à travers le réseau les inscriptions en base de registre. Quelques logiciels sont cependant conçus pour autoconstruire une install sur un serveur qui devra être ensuite exécuté sur chaque station mais de façon simplifiée.

Il arrive souvent sur un système NT ou W98 que l'installation d'un logiciel provoque une instabilité. De ce fait, l'install réseau devra être fait prudemment la solution la plus simple étant de cloner les disques soit physiquement (à l'aide de ghost ou d'un boot linux) ou par le réseau via une procédure automatisée fonctionnant à l'aide d'un autre système ou via une double installation.

La configuration utilisateur se fait, elle, par l'utilisation de profils. Les profils sont constitués d'un fichier NTconfig.pol destiné à aller dans le partage netlogon du serveur et d'une arborescence de fichiers au nom de l'utilisateur dans le répertoire /winnt/system32/profile. Dans cette dernière, on trouve deux éléments intéressants le défaultprofil et comonprofil, ce dernier permettant les ajouts dans les profils existants.

2.1.5 outils

Il existe de nombreux outils d'aide à la gestion de grands sites. La plupart sont orientés réseau. On peut en détailler quelques uns :

- Les outils basés sur le protocole SNMP
  Traditionnellement SNMP permet de transporter des statistiques sur l'état des machines et du réseau. Ce n'est pas un outil d'administration en propre mais un outil de mesure et de surveillance
- Les outils d'administration constructeurs
  Par exemple solstice network administrator de SUN qui est une suite très complète d'outils.
- Les outils intégrés à certains systèmes
  Le système de répartition en cluster de VMS ou de True64 unix de compaq (ex : DEC)
- Les outils fournis par d'autres sociétés de logiciels,
  Par exemple Tivoli par IBM/Sycomore ou Prat'X de la société ALX[1]

D'une manière générale, on utilise les outils fournis et surveillés par le constructeur et comme on est en général en milieu hétérogène, tout repose bien souvent sur une série de Shell scripts écrits les uns après les autres.

D'une manière générale, il faut se construire une boîte à outils permettant de passer les PATCH (correctifs) constructeur et d'assurer une reconfiguration complète et automatique d'une machine. Ce script se constitue peu à peu en ajoutant les modifs que l'on fait et en ajoutant une ligne à la crontab permettant de tester si des modifs sont nécessaire.

Si on a pris la précaution de faire ce qui précède, l'installation d'une nouvelle machine se résumera à une install de son OS, puis à la réalisation d'un montage NFS permettant d'accéder au script et au lancement du dit script.

[1] En plus elle embauche des ZZ

2.2 Choix OS/Matériel

Bien souvent, le choix de matériel est la partie qui paraît la plus facile à l'administrateur de systèmes. Il faut pourtant faire très attention à réfléchir ces choix à long terme et toujours avoir à l'esprit qu'un matériel informatique a trois ans de durée de vie et que trois ans, ça peut-être très long.

- Il faut respecter l'existant.
- Il faut respecter les désirata des utilisateurs, mais sans verser dans le clientelisme. Dans ce domaine il ne faut jamais critiquer ce que veut l'utilisateur, il faut plutôt lui expliquer le bien fondé et l'équivalence de ce que vous proposez. Un changement de matériel imposé n'est jamais bien vécu et si vous faites passer quelqu'un d'un MacIntosh à un terminal X sous windows, n'oubliez pas de lui expliquer un minimum de choses, sans quoi il vous maudira.
- Pour les postes de travail, il faut toujours viser à diminuer la maintenance probable et donc essayer de mettre en place des serveurs d'applications (RDP WTSE/X11 UNIX/VAX VMS/META FRAME) et des terminaux appelés clients légers[2].
  Si on adopte cette solution, il faudra être particulièrement rigoureux dans la gestion des serveurs et repérés les utilisateurs qui poseraient problème sur ce type de matériel. D'où la règle suivante :
- Fournir une puissance locale aux utilisateurs gros consommateurs.
- Il faut limiter l'hétérogénéité car c'est l'ennemi d'une bonne gestion du fait des overhead de maintenance qu'elle introduit.
- Il faut segmenter le réseau de façon hiérarchique en classant les trafics de façon à augmenter la bande passante disponible pour chaque groupe de travail.
  Une erreur classique dans ce domaine est d'investir la majorité des fonds dans le fédérateur. En effet, si les entités dialoguent peu et si on a bien conçu le système, cela n'est pas nécessaire.

L'achat de matériel se fait en général soit via le service comercial du constructeur ou via un revendeur. Les deux ont leurs spécifités et leurs tarifs propres. Le constructeur vous apporte en général un service de qualité conforme à des standarts alors qu'un revendeur vous apportera un service plus brouillon mais certainement plus souple.

Le malheur de l'informatique est que les prix ne sont pas fixés une bonne fois pour toutes et qu'il faudra donc marchander chaque appareil. Ces discutions ne devront par faire perdre de vue les buts initiaux de l'achat et inclure le prix d'un contrat de maintenance prépayé pour les matériels sensibles.

2.3 Sécurité d'un système

La sécurité et son dual, le piratage, sont souvent des sujets excitants pour les informaticiens. Le point de vue de l'ingénieur système est bien souvent différent : pour lui, la sécurité ça lui fait perdre son temps et pour le pirate c'est un simple jeux. En fait dans tout cela, il ne faut jamais oublier que derrière les machines se cachent des gens et que les Giga-octets représentent souvent un grand nombre d'heures de travail.

[2] C'est-à-dire qui ne font tourner que les programmes nécessaires à l'affichage à travers le réseau

2.3.1 Définitions

Il y a trois types d'atteinte à la sécurité :

- physique,
- intérieure,
- extérieure.

L'armée des Etats-Unis a défini les classes suivantes : (il s'agit de la classification du livre orange)

- D : pas de sécurité, (exemples : MS-DOS, W9X)
- C1 : les usagers sont authentifiés. Chaque usage a la délégation de sécurité. Chaque usager a accès à ses informations de sécurité. (exemples : SUN OS, Linux)
- C2 : les utilisateurs n'ont pas accès à leurs informations de sécurité. (exemples : Solaris, NT)
- B1 : droits d'accès à plusieurs niveaux. (exemples : VMS, NT, S7)
- : obligation de droits d'accès à plusieurs niveaux. Absence de délégation. (exemples : VMS, TRUSTED UNIX)
- : Isolation matérielle double. Code d'identification + identification physique.
- : Modèle formel de sécurité, preuve mathématique de validité. (exemple : SCOMP de HP)

2.3.2 Sécurisation

Dans un système, la sécurité physique est assurée par une bonne porte, une serrure correcte et un système anti-incendie.

Prévention interne

Au niveau interne, plusieurs types d'atteintes (? ?)

- via le changement d'UID, (ex : SETUID, vol de mots de passe, Pseudo montage)
- via le changement des droits d'accès,
  Si on ne prend pas garde à brider les possibilités de suppression et d'ajout de fichiers (sticky bit c'est-à-dire chmod +t) dans l'arborescence des fichiers, on prend un gros risque car un utilisateur malveillant pourra remplacer les fichiers d'origine par les siens.
- via l'utilisation du matériel.
  Utilisation de périphérique amovible, par exemple, en utilisant le lecteur de disquette pour booter ou plus simplement en déroulant une cassette de sauvegarde laissée dans le lecteur.

Sous UNIX, de manière générale, les fichiers systèmes doivent être protégés rwx-x-x, mais certains doivent cependant être rwxr-xr-x.

Sous NT, il faut sécuriser l'arborescence système, l'usurpation de droits se faisant en général grâce à l'ajout d'un groupe.

Les SUID doivent être sévèrement contrôlés et écrits en C avec des chemins en dur construits à l'aide de plusieurs chaînes.

```
int main(int argc, char ** argv)
{
        char path[]= "/usr/bin/ ", niet[23],prog[]= " toto " ;
```

```
        int UID = getuid() ;

        signal(SIGSEGV,     badsegviol);
        signal(SIGPIPE,     badconnection);
        signal(SIGALRM,     badtimeout);
        umask(022);

        setuid(0) ;
        system(sprintf(buffer,      " %s %s ", path, prog)) ;
        setuid(UID) ;

        exit(0) ;
}
```

On ne donne les droits root que lorsque c'est nécessaire c'est-à-dire juste pour l'exécution de la commande système et on empêche le programme de créer des fichiers core qui pourraient être ensuite exploités pour loger des exécutables appartenant à root.

Les mesures minimum à prendre sont de plusieurs ordres. En voici un petit résumé.

La premiére des erreurs à ne pas commettre est d'avoir des login génériques, sans mot de passe ou avec des mots de passe évidents, cela se traduit par les deux régles suivantes :

Un compte par paire d'oeil C'est-à-dire qu'il faut absolument évité d'utiliser des comptes génériques pour un groupe de personnes des lors que la machine mise en question est sur le réseau. En effet un pirate peut profiter de ce login pour installer des chevaux de Troie qui lui permétttront de s'attribuer plus de pouvoir.

Un mot de passe correct par compte Un mot de passe correct doit comporter au moins 8 caractères dont un n'est pas alphabétique. Pour obtenir des mots de passe corrects, il suffit de prendre une phrase courte avec sa ponctuation et d'utiliser la première lettre de chaque mot, plus la ponctuation. Ex : "MultiPack 2 pour système Ultra, voir page 42" donne : M2SUVP4. Une tentative de crackage par dictionnaire échouera à coup sûr.

Pour des raisons similaires, la gestion des login est importante :

Supprimer les login inutiles Les utilisateurs considérent souvent qu'un compte est ouvert à vie. Comme il est naturel que quelqu'un qui quitte un lieu rende les clefs, il est naturel de fermer son compte En effet, si un compte n'est plus utilisé régulièrement, un pirate pourra l'utiliser sans risque de se trouver confondu.

Obliger les utilisateurs à changer de mots de passe fréquemment Les machines sont de plus en plus performantes, un pentium 200 permet de craquer tous les mots de passe à 6 caractères en 4 jours de calcul, il faut donc au moins changer tous les six mois si on respecte les régles d'établissement citées plus haut.

On doit pouvoir vous joindre :

Les mails destinés aux comptes root ou postmaster doivent vous parvenir Il faut que ces mails vous parviennent. Ils doivent être renvoyés sur votre compte, sur les machines dépourvues de compte root, il est important de créer un alias mail vers votre compte utilisateur.

Mesures de sécurité extérieure

L'atteinte à la sécurité par l'extérieur se fait via le réseau : (? ?)

- utilisation détournée d'un protocole
  Dans ce cas, l'ami pirate exploite les bugs d'un daemon système de façon à produire des modifications du système de fichiers de sa cible. L'exemple typique est l'utilisation de ftpd pour la production de fichier core contenant autre chose qu'un dump mémoire.
- vol de map NIS et/ou NFS, SMB
  La, il s'agit d'utiliser une mauvaise configuration de la machine pour accéder aux informations de sécurité ou aux fichiers. Dans les cas des NIS, le cracker cherche la plupart du temps à accéder aux chaînes publiques des mots de passe cryptés de façon à tester une base de mots de passe. Les versions modernes de NIS et NIS+ possède un contrôle d'accès par site dont il faut se servir.
  Le vol de partition NFS se fait via l'utilisation d'autorisation laissée alors qu'une machine a disparu ou par plantage volontaire d'une machine pour utiliser son identité. On ne doit donc exporter des partition NFS que vers des machines qui fonctionnent en permanence et limiter au maximum les possibilités d'utilisation anonyme ou en root. Pour les mêmes raisons, il ne faut jamais mettre de fichiers de configuration du système dans les partitions NFS (table d'alias mail par exemple).
  Le vol de partition SMB ou d'information NetBeuI se fait en général par la découverte du mot de passe d'un compte administrateur. Le gros défaut de NT dans ce domaine est que les contrôles d'accès par site son quasi inexistants, qu'il est très facile de s'ajouter à un domaine (si on a le mot de passe) et que les partitions systèmes sont exportées par défaut. Tout repose donc sur le mot de passe des comptes qui ont le droit d'ajouter une machine au domaine et sur le mot de passe du compte administrateur.

Deux types de contrôle : Par Site : on ne considère que l'identification de la machine, Par Clef : l'entité demandante doit fournir une clef qui lui est propre (identification RSA ou MAGIC-COOKIE). Le contrôle par clef est beaucoup plus efficace que par site. Le contrôle par site et par clef est le plus efficace.

Les outils de sécurité sont nombreux et il faut distinguer les outils locaux aux machines de ceux qui concernent le réseau. Les principaux logiciels et techniques de sécurisation sont de machines :

- le filtrage local le tcpwraper : c'est un petit programme qui juge si les demandes faites au daemon inet.d correspondent à des sites autorisés.
- L'utilisation d'adresses IP non routables combinée à l'utilisation extérieure via un proxy
- SSh et SSL qui réalisent un cryptage des informations envoyées via les sockets.
- la sécurisation par fausses informations, cela consiste à modifier l'aspect externe du système de façon qu'une recherche de trous automatiques échoue.
- sécurisation par supervision, une machine logue tous les trafics et corromp ceux qui sont anormaux. C'est la technique dite du firewall ou garde barrière en Suisse Romande.
- technique du routeur absent, les machines ne possèdent que les routes dont elles ont besoin et le routeur par défaut est envoyé sur une machine qui filtre les demandes. Ceci permet de recevoir beaucoup de trafic sans pour autant déployer un firewall d'une puissance délirante. Ceci se fait facilement avec un système linux equipé de la suite IPCHAIN.
- Les patchs et hotfix constructeurs sont aussi des choses à surveiller. Ce sont des correctifs

systèmes qui améliorent le fonctionnement des machines et permettent d'éviter les gros trous de sécurité qui ont été signalés partout.

Les mesures de sécurité à adopter pour les systèmes NT et UNIX sont les suivantes :

Mesures pour UNIX

Il existe de nombreuse versions d'UNIX cette partie est donc découpé entre mesure générique et mesures spécifique à un système donné

pour ce qui concerne les fichiers /etc/group et /etc/passwd :

- Modifiez les mots de passe fournis avec le systeme.
- Otez les utilisateurs de service (guest, visitor, tutor, demo, ...). Verifiez que les comptes anonymes (sys, uucp, bin, adm, lp, ...) ont **"*"** comme mot de passe.
- Utilisez les extensions pour que les mots de passe chiffres n'apparaissent pas dans le fichier passwd (shadow password)
- Les droits du fichier /etc/passwd et /etc/group doivent etre 644 (rw-r–r–)

dans le cadre de l'utilisation des NIS (YP) :

- Faire un fichier de passwd pour les NIS diférent de celui du master serveur
- Utiliser le fichier securnet ou les restrictions d'accés au tables
- Interdir la commande ypcat aux personnes autres que les administrateurs
- remplacer + : :0 :0 : : : par + : :65535 :65535 : : : dans /etc/passwd
- Si vous avez un serveur NFS dédié, suprimer l'accès au utilisateur lambda à l'aide des netgroup.

Pour le reste :

- Sécurisation des accés généré par inetd (telnet rlogin ,ftp, ...) par un tcpd
- Sécurisation de NFS, les exportations doivent être srictement borné les partitions le plus possible exporté en readonly et les partition utilisateurs monter en Nosetuid.
- Sécurisation des fichiers .rhosts et hosts.equiv, le mieux est de banir leur utilisation et d'utiliser la suite ssh à leur place.
- Attention au cron ou un petit malin peu insérer des choses qui reste.

Cette partie est librement inspirer des oeuvres de Thierry Besançon et Jean-Luc Archimbaud.

Mesures pour NT

Le système WINDOWS NT est assé sur, il ne permet que peu de fonctionalité réseaux, mais l'ors d'une installation par défalut il n'est pas vérouiller pour deux sous. Une introduction extérieur aura donc des conséquences difficile à prévoir.

- **Sécurisation des accés :**

  Vérouiller le compte invité ou guest

  Sécurisation des répertoires :
- **Protection des fichiers et répertoires :**
  Utilisisez des partitions NTFS et appliqué les droits suivants (conseil de tonton Dubois ...) :
  Le réglage standard des permissions sur le système de fichiers et des répertoires fourni un degré de sécurité raisonnable (paraît-il) sans gêner le fonctionnement de lordinateur.

Pour des installations sécurisées, cependant, vous devez ajouter des permissions à tous les sous répertoires et fichiers existants, comme montré dans le tableau ci-dessous, ***immédiatement après l'installation de Windows NT***. Assurez-vous dappliquer les permissions aux répertoires parent avant dappliquer les permissions aux sous-répertoires.

Répertoire	permissions
\WINNT et ses sous répertoires	Administrateurs : Contrôle Total CREATEUR PROPRIETAIRE : Contrôle Total Tout le Monde : **Lire** SYSTEME : Contrôle Total
\WINNT\REPAIR	Administrateurs : Contrôle Total
\WINNT\SYSTEM32\CONFIG	Administrateurs : Contrôle Total CREATEUR PROPRIETAIRE : Contrôle Total Tout le Monde : **Lister** SYSTEME : Contrôle Total
\WINNT\SYSTEM32\SPOOL	Administrateurs : Contrôle Total CREATEUR PROPRIETAIRE : Contrôle Total Tout le Monde : **Lire** Utilisateurs avancés : **Modification** SYSTEME : Contrôle Total
\WINNT\COOKIES \WINNT\FORMS \WINNT\HISTORY \WINNT\OCCACHE \WINNT\PROFILES \WINNT\SENDTO \WINNT\Temporary Internet Files	Administrateurs : Contrôle Total CREATEUR PROPRIETAIRE : Contrôle Total SYSTEME : Contrôle Total Tout le Monde : A. s. à un rép. : **Lire, Ecrire, Exécuter** Tout le Monde : Accès spécial à un fichier : **none Add**

– Affecter les permissions suivantes à ces fichiers :

Fichiers	permissions
\Boot.ini, \Ntdetect.com, \Ntldr	Administrateurs : Contrôle Total SYSTEME : Contrôle Total
\Autoexec.bat \Config.sys	Administrateurs : Contrôle Total SYSTEME : Contrôle Total Tout le Monde : **Lecture**
\TEMP directory	Administrateurs : Contrôle Total SYSTEME : Contrôle Total CREATEUR PROPRIETAIRE : Contrôle Total Tout le Monde : A. spécial à un rép. : **Lire, Ecrire, Exécuter** Tout le Monde : A. spécial à un fic. : **none Add**

To view these files in File Manager, choose the **By File Type** command from the **View** menu, then select the **Show Hidden/System Files** check box in the **By File Type** dialog box.

– mesure spécifique Windows NT Workstation

Répertoire	permissions
\WINNT\SYSTEM32\DHCP	*Détruire ce dossier*
\WINNT\SYSTEM32\DRIVERS	Administrateurs : Contrôle Total CREATEUR PROPRIETAIRE : Contrôle Total Tout le Monde : **Lister** SYSTEME : Contrôle Total
\WINNT\SYSTEM32\RAS	*Détruire ce dossier*
\WINNT\SYSTEM32\OS2	*Détruire ce dossier*
\WINNT\SYSTEM32\WINS	*Détruire ce dossier*

2.3.3 Oui mais !

Quelquesoient les methodes et les outils employés, la sécurité est surtout une affaire de rigueur et de constance dans la surveillance des machines. Si vous exercez un jour le métier d'administrateur, vous serez surpris de la part de temps que cela va vous consommeer et de toute façon, tôt ou tard, un petit malin rentrera sur vos machines. Il faut donc apprendre à réagir et s'y préparer.

2.3.4 Que faire en cas d'attaque ?

En cas de problème de sécuité, il y a des questions importantes auxquelles il faut répondre immédiatement :

1. Est-ce que le pirate a acquis LES DROITS SYSTEMES ?
2. Est-ce que le pirate est entré sur d'autres machines de mon réseau ?
3. Est-ce que le pirate est entré sur d'autres machines d'autres réseaux ?

La liste suivante énumére les mesures à prendre suivant les cas :

- A prendre systématiquement : Débrancher la machines concernée du réseau
- Rechercher toutes les traces de connexions suspectes
- vérifier l'intégrité du noyau : les signes : changement de taille, de date ou de signature MD5. Si oui, il faut tout débrancher et réinstaller complètement le système. Pourquoi, parce que le noyau permet de masquer tout ce que l'on veut et donne un accès complet non interruptible
- Vérifier que la machine n'a pas rebooté ce qui laisserai supposer de grosses bidouilles
- Rechercher tous les programmes suspects
- Prévenir la hiérarchie et le fournisseur d'accès,
- Faire changer les mots de passe à tous les utilisateurs
- Couper les accès extérieurs sur cette machine
- Si le cas 1, 2 ou 3 est vérifié, déconnectez votre réseau de l'Internet et du reste de l'entreprise
- Si le cas 1 est vérifié, vérifiez toutes les commandes du système (taille, droit) ou réinstaller
- Interdire provisoirement toute connexion sur cette machine
- Si le cas 2 est vérifié, éteindre les machines concernées ou déconnecter les de votre réseau
- Si le cas 3 est vérifié, supprimer la route vers ces réseaux Prévenir les administrateurs de ces réseaux

Que réparer est toujours la question qui assaille l'administrateur dans ces moments de grande fébrilité. Deux possibilités :

1. Si on réinstalle, c'est une mesure absolue. Il faut réinstaller et créer tous les ex-utilisateurs. Sous NT, c'est une opération faisable mais, du fait de la numérotation automatique des utilisateurs, c'est très chaud.
2. Rechercher une sauvegarde et la remettre.

   Problème : quelle sauvegarde ? (problème de date)

   La procédure à respecter est la suivante afin d'éviter toute contagion sur les bandes :
   - Faire un backup de la machine incriminée
   - Prendre une autre machine
   - Dérouler la sauvegarde
   - Chercher les traces
   - Si plus de trace,
   - recopier la sauvegarde sur la machine craquée
   - remettre les fichiers les plus récents en les contrôlant un par un

   Bon courage ...

voilà ! ! !

2.3.5 Conclusion

La Sécurisation réseau est le plus simple : il suffit d'utiliser du matériel spécialisé : Switch avec Vlan et Routeur/Firewall. Mais elle ne garantit que partiellement contre les attaques intérieures qui sont le gros problème des organismes de formations d'informaticiens. Répétons-le : la sécurité est une affaire de rigueur dans les procédures et de constance dans la surveillance, le tout en faisant que les utilisateurs ne soient pas complètement bloqués par des procédures de sécurité indémerdables et sans devenir totalement paranoiaque.

2.4 Correction des TP

Quelques questions réponses sur UNIX versus linux

les systèmes UNIX se répandent mais pas mal d'opérations de bases sont non triviales. Voilà quelques situations et exercices pour s'en sortir quand ça ne boote plus.

2.4.1 Notion d'Utilisateur

sujet

- *Créer des utilisateurs UNIX pour les 2 personnes qui vous entourent. sans utiliser adduser*
- *Créer des groupes pour chaque paire d'utilisateurs.*
- *Créer des répertoires de travail en plus du home dir pour chaque paire d'utilisateurs*
- *Comment faire en sorte de rendre lisible ces répertoires à coup sûr par les binômes.*

corrigé

Il faut pour cela modifier deux fichiers système (/etc/passwd et /etc/shadoww) , créer des répertoires aux utilisateurs et copier un fichier de login (.bashrc) dans le répértoire de chaque utilisateur.

L'entrée à ajouter au fichier /etc/passwd est de la forme suivante :

```
siad:x:503:503:Odile      siad:/home/garp/siad:/bin/bas      h
```

ce qui doit ce comprendre :

```
login:x:Uid:Gid:Real      Name  in  Real  Life:home   directory:shell
```

On met donc dans l'ordre : le nom de login souhaité (moins de huit caractères), un x dans le champ passwd (sauf sur les vieilles machines où il y a la chaîne crypter), le numéro d'utilisateur, le numéro du groupe de l'utilisateur, le vrai nom de l'utilisateur et le shell de l'utilisateur.

Pour le fichier /etc/shadow l'entrée est de la forme :

```
siad::11088:0:99999:7:-1:-1       :1345 3790
```

En fait, seuls les deux premiers champs sont importants, il s'agit du login et du mot de passe crypté. On ne met rien dans le deuxième et aussitôt après on tape **pawssd login** pour y mettre un mot de passe.

La description complète des champs de ce fichier est donnée par la commande **man 5 shadow**.

Si besoin on rajoute le groupe dans le fichier de groupe /etc/group de la même façon. pour une description complète **man 5 group**

Les ânneries à ne pas faire sont : oublier **:** comme séparateur ou d'en mettre un de trop, modifier les autres entrées du fichier peut aussi conduire à des situations cocasses.

Ensuite on crée le répertoire de l'utilisateur et on y met les bons fichiers :

```
mkdir /home/siad
cp /root/.bashrc /home/siad/bashrc
chown -R siad /home/siad
chgrp -R 501 /home/siad
```

Quand on souhaite faire travailler plusieurs personnes sur les mêmes fichiers regroupés dans le même répertoire, on créé un groupe commun en rajoutant une ligne dans /etc/group (ex web) :

```
web:*:501:gouinaud,lefortp,degi o,soft,pinson,jma,gallois,verdot,grambou l
```

On change le groupe du répertoire visé avec chgrp (chgrp -R web /home/web) et on le groupe des fichiers créés à l'intérieur du répertoire à l'aide du setgid bit en tapant **chmod g+s /home/web** et c'est tout.

2.4.2 Process Utilisateur

sujet

Taper le programme suivant :

```
#include <stdio.h>
void main()
{
FILE *fp;
fp = fopen("toto","w ");
if (fp=NULL)
  exit(1);
fprintf(fp,"tut u\n");
fclose(fp);
for (;;);
exit(0);
}
```

- *Lancer le programme et trouver à l'aide de la commande ps le PID du process ainsi créé.*
- *Essayer de le tuer sous divers utilisateurs*
- *Quel est le propriétaire du fichier et le groupe de* **toto** *?*

Corrigé

La liste des process s'obtient à l'aide de la commande **ps -aux** ensuite on prend le numéro dans la deuxième colonne de la liste et on utilise la commande **kill -15** pour tuer le processus. Par exemple :

```
[gouinaud@garp sys]$ ps -aux
gouinaud 6961 0.0 1.9 3956 2556 pts/1 S 14:05 0:00 xdvi.bin -na
[gouinaud@garp sys]$ kill -15 6961
```

Si le process ne disparaît pas, il faut réessayer le kill mais avec la valeur -9.

Les possiblités de tuer son voisin sont très limitées sous UNIX. Pierre n'a pas le droit de tuer paul et le root peut tuer tout le monde y compris lui.

Le programme qu'il y a plus haut crée un fichier, on remarque que ce fichier appartient au login qui a lancé le programme et que l'utilisateur qui lance le programme doit avoir les droits d'écriture suffisants pour que le programme se déroule normalement.

2.4.3 gestion des comptes

sujet

Récupérer a l'aide de ftp les fichiers de passwd et de groupe de la machine fc. Faîtes en sorte que les comptes de tous vos collègues fonctionnent sur cette machine.

Rajouter ensuite à chaque utilisateur la possibilité d'accéder à un répertoire supplémentaire commun qui permettra, par exemple, à tous de modifier les pages web de votre serveur.

Corrigé

Pas de difficultés particulières pour cette partie, on récupère les fichiers passwd et shadow de la machine distante et on copie les lignes qui nous intéressent dans les bons fichiers.

Pour récupérer le fichier shadow, il faut passer par un compte normal mais pensez à en faire une copie lisible par ce compte en faisant un su.

2.4.4 Fichier /etc/hosts et DNS

sujet

Toutes les machines ayant une adresse IP n'ont pas forcément une adresse dans le DNS. Le fichier /etc/hosts permet de leur donner un petit nom.

Le probleme :

- *Taper telnet zut2 sur vos machines (il se produit une erreur) sauf sur zut2 ou l'on tape telnet zut3 pour avoir l'erreur.*
- *taper vi /etc/hosts*
- *ajouter une ligne pour définir zut2 (resp zut3)*
- *retaper le telnet ca marche*

On voit donc le remède ...

Afin de régler le pb récupérer le fichier etc hosts qui est sur le serveur ftp ://gargamel.isima.fr/ (login anonymous)

Corrigé

Pour qu'une machine puisse être invoquée par son nom et pas uniquement par son adresse IP il faut soit qu'elle soit enregistrée dans le DNS soit qu'elle ait une ligne de définition dans le fichier /etc/hosts.

voilà un exemple de fichier /etc/hosts :

```
127.0.0.1       localhost       localhost.localdomain
193.55.95.1     sp
193.55.95.2     flamengo
193.55.95.6     power6
193.55.95.31    gargamel.isima.fr    gargamel
193.55.95.32    garp.isima.fr       garp
```

Dans la colonne de gauche vous aurez reconnu l'adresse IP de la machine, en ensuite les différents noms sous laquelle on veut qu'elle soit connue.

2.4.5 Montage Nfs

sujet

Afin de récuperer les directory de vos collègues utiliser la commande :
mount -t nfs fc :/home /home/fc
essayer de comprendre ce qui se passe .... essayer ce montage à divers endroits.

Corrigé

Pas de difficulté à ce niveau si ce n'est que le système de fichier que l'on veut monter doit être exporté par le serveur (fc). On a le choix du point de montage, donc en général on ne fait pas de folies, on le met au même endroit que sur le serveur : fc :/home en /home.

Ceci, associé à la récupération des entrées des champs passwd et shadow, permet de débordéliser méchamment un réseau de machines en ayant un espace disque unique pour chaque utilisateur.

2.4.6 Serveur apache

sujet

la page d'entrée du serveur apache de votre machine est dans /home/httpd/html.
Remplacer la par celle de gargamel.isima.fr

Corrigé

Pas grand chose à dire si ce n'est qu'il faut utiliser le login **anonymous** quand on veut recupérer des fichiers sur un serveur ftp où l'on a pas de compte. Dans ce cas on donne son adresse **email comme passwd**.

2.4.7 rpm

sujet

On installe des packages à l'aide de Kpackage,

Aller sur le site http ://www.linux.org/ et récupérer un package du logiciel htdig ou autre.

Corrigé

On trouve beaucoup de logiciels sur www.rpmfind.net. Quand on récupère un logiciel il faut faire attention que celui-ci corresponde bien à votre version de linux et à votre architecture (i386 pour un PC).

On peut installer des packages à l'aide de **kpackage** ou de **rpm -i nom-du package.rpm** si on est privé d'interface graphique. Attention à ne pas installer n'importe quoi si on ne veut pas trop bordéliser sa machine.

2.4.8 Creation de parttion (fdisk)

sujet

1. *Détruire la partion 4 du disque.*
2. *Créer une partition étendue en 4*
3. *Créer deux partitions ext2 5 et 6 de tailles équivalentes*

Corrigé

On utilise pour cela la commande fdisk, si on a plusieurs disques il faut taper **fdisk /dev/hdb ou hdc ou hdd** et fdisk tout court si l'on n'en possède qu'un seul. Pour déterminier les bons hd il faut lire ce que marque la machine au moment du boot.

dans fdisk :

- p affiche la table des partitions
- m affiche l'aide
- d permet de détruire une partition
- n d'en créer une nouvelle
- t dans changer le label

Les règles à appliquer sont les suivantes :

- On ne peut créer des partitions que dans un espace vide, il faut donc en détruire si on a plus de place. Dans ce cas les données contenues dans les partitions détruites sont perdues.
- On a au plus quatre partitions primaires utilisables et une partion étendue qui peut contenir d'autre partitions dites logiques. Le maximum toléré par beaucoup de systèmes est une primiare et une étendue. Les logiques sont, par contre elles, en nombre quelconques.
- Attention à ne pas créer des partitions se recouvrant, ce qui est possible sur certain systèmes.

Une fois que l'on a fait les modifications que l'on souhaite on sort de fdisk avec w pour enregistrer.

2.4.9 Raccordement au système de fichiers (mkfs, mount, /etc/fstab)

sujet et corrigé

1. *Mettre des systèmes de fichier sur ces deux partitions*
   mkfs /dev/hda5 et mkfs /dev/hda6. mkfs est l'équivalent du format du dos, attention il n'existe aucun moyen d'annuler cette opération.
2. *Monter provisoirement ces deux partitions en /mnt/5 et /mnt/6*
   mkdir /mnt/5 /mnt/6 mount /dev/hda5 /mnt/5 mount /dev/hda6 /mnt/6
3. *Copier les fichiers /var/* dans /mnt/5*

   ```
   cd /var; tar -cvf /vars.tar .
   cd /mnt/5; tar -xvf /vars.tar
   ```

   ou

   ```
   cd /var; tar -cvf - . | (cd /mnt/(; tar -xvf -)
   ```

   utiliser tar pour cela. Pourquoi ?
   Si on utilise cp on change les droits des fichiers.
   En quelle mode de fonctionnement. Pourquoi ?
   En mode 1, pour que les fichiers de log dans /var/log ainsi que tout les fichiers de spool ne soient pas incohérents. On passe en mode 1 à l'aide de la commande **init 1** plutôt que de rebooter (sur un système avec de nonbreux disques ça prend moins de temps).
4. *Rendre le montage de 5 en /var/permanent*
   On modifie le fichier /etc/fstab sur linux et /etc/vfstab sous Solaris pour cela. on rajoute une ligne de la forme :

   ```
   /dev/hda5        /var        ext2    defaults
   ```

5. *Monter la partition 6 sur /home, que se passe t-il ?*
   Les fichiers qui étaient dans le répertoire /home original n'apparaissent plus, il faut donc penser à les copier comme pour /var.
6. *Rendre le montage de 6 en /home/zutXX permanent*
   de même que plus haut on rajoute une entrée dans la fstab.
7. *Deplacer vos utilisateur vers /home/zutXX*
   En fait, il faut copier les fichiers des utilisateurs et mettre à jour le /etc/passwd.

2.4.10 problème d'ecriture dans vfstab (vfstab)

Sujet

1. *Metre un # devant la ligne /usr dans la vfstab*
2. *Rebooter*
3. *réparer*

corrigé

Le mieux est de rebooter en niveau 1 avec la directive linux 1 à taper juste quand lilo apparaît sur l'écran. Une fois que la machine a booté on répare en remodifiant le fichier fstab puis on sauvegarde. Ensuite on reboote de nouveau normalement.

La difficulté dans un cas similaire réel est de retrouver la bourde que l'on a fait. Il faut donc noter tout ce que l'on fait dans un cahier.

2.4.11 supression de la fstab

Sujet

1. *Faire un mv du fichier fstab en fstab.save*
2. *rebooter*
3. *réparer*

utiliser le disque rescue. Pourquoi ? Pourquoi ça ne boote plus ?

corrigé

La machine ne reboote plus parce qu'elle ne peut pas monter les systèmes de fichiers décrits dans la fstab. De plus le / est monté en ro donc on ne pas le corriger.

Il y a deux façons pour sans sortir, soit booter en niveau 1 et remonter en rw le système de fichiers /, soit utiliser une disquette rescue.

La première se pratique à l'aide de mount avec l'option remount et -n qui desactive l'écriture dans le fichier mtab. Par exemple :

```
mount -n -o rw,remount /dev/hda5 /
```

si votre / reside dans la partition 5 du disque dur.

Avant il est souvent nécessaire de faire un check de la partition incriminée. Par exemple :

```
fsck.ext2 -y /dev/hda5
```

Ensuite on répare les fichiers endommagés et on reboote pour de nouvelles aventures.

Si on décide d'utiliser une disquette rescue il faut suivre les instructions dans le fichier images/rescue/README du cdrom. Une fois la disquette fabriquée on boote dessus puis on monte le systeme de fichier root du disque dur. Cela ce fait ainsi :

```
e2fck -y /dev/hda5
mount /dev/hda5 /mnt
cd /mnt
chroot /mnt /bin/sh
```

ensuite on répare les fichiers puis on sort de tout cela :

```
exit
cd /
sync
umount /mnt
halt
```

On reboote de nouveau pour de nouvelles périgrinations.

2.4.12 Perte du mots de passe root

Sujet

1. *Editer le fichier /etc/shadow*
2. *Mettre un *bug* dans le champ passwd de root*
3. *rebooter*
4. *réparer*

corrigé

Ou le système est bien fait et il faut corriger a l'aide de la disquette rescue, ou le système est mal foutu et on peut corriger en passant en niveau 1.

Moralité, si on a accès à la console la sécurité ça n'existe plus sous unix.

2.4.13 Emelage de rc

Sujet

1. *faire un mv de /etc/rc.d/init.d /etc/rc.d/init.d.bab*
2. *rebooter*
3. *réparer*

corrigé

Là, il n'y a pas trente six solutions, la machine n'a même plus de quoi booter en niveau 1 le plus simple est donc de booter sur disquette et d'aller bricoler dans le disque dur avec la séquence décrite en 2.4.11.

2.4.14 Script de sauvegarde

Sujet

Faire une script qui sauvegarde les informations système sensibles dans un répertoire de la partition utilisateur. Automatiser cette tache grâce à la crontab.

corrigé

On s'apercoit à la lecture de ce qui précède qu'il y a pas mal de fichiers précieux sur la machine, l'avantage c'est qu'ils sont répartis dans deux répertoires :

- /etc qui contient tout le paramétrage de la machine
- /boot qui contient tous les fichiers nécessaires au boot

Pour faire une sauvegarde il suffit donc de faire une script du genre :

```
vi /sauve.sh
------------------      dans vi -------------
#/bin/sh

mkdir /home/sauve

tar -cvf /home/sauve/sauve.`date`.tar       /boot  /etc

<ESC>
:wq
------------------      Fin de vi -------------
chmod a+rx /sauve.sh
```

Pour sauver il suffit de taper une commande :

```
/sauve.sh
```

ou

```
crontab -e
------------------      dans vi -------------
30 12 * * * /sauve.sh
<ESC>
:wq
------------------      Fin de vi -------------
```

pour que cela se fasse tous les jours à midi. Remarquer dans le script l'astuce des cotes ` (accent grave) autour de **date** qui permet de créer un fichier à la date et à l'erreur du lancement de la sauvegarde.

Vous allez me dire sauvegarder c'est bien beau, mais si on veut restaurer on fait comment.

Pour les fichiers qui sont dans /etc on les extrait simplement du tar en se mettant dans /home/sauve puis on les remet a leur place. Pour les fichiers dans /boot (dont le fameux /vmlinuz-*-*) on les remet en place puis on tape **/sbin/lilo** pour remettre à jour le secteur de boot.

2.4.15 arrêt intenpestif

La machine peut refuser de rebooter en cas de panne de courant ou de maladresse boutoneuse.

Dans ce cas, au boot elle tente de réparer elle même les systèmes de fichiers et si elle n'y arrive pas elle passe en mode maintenance (niveau 1 ou mono utilisateur). Dans de cas elle vous demande le mot de passe root.

il faut :

- entrer le mot de passe root
- Faire un fsck de tous les systèmes de fichiers listés dans /etc/fstab
- taper <CONTROL>+D

La machine doit alors rebooter normalement.

2.4.16 At, crontab

Sujet

- *Afin de ne pas perdre vos fichier*
- *Ajouter l'ordre sync dans la crontab du root*

corrigé

Le système crontab permet de faire exécuter à des horaires et dates donnés des tâche récurantes. Il repose sur le fait que tourne le process **crond** et sur un fichier /var/spool/cron/USER pour les tâches du login USER (/var/spool/cron/root pour root).

Les lignes de ce fichier comportent 6 champs séparés par des espaces :

```
# MIN  HEURE  JOURS  MOIS  J_SEMAINE   COMMANDE
  *    *      *      *     *           /root/tous_le_temps
  30   12     1      12    *           /root/a_12h30_le_1_12
  0    1      *      *     0           /root/dimanche
```

Il ne faut jamais éditer ce fichier autrement qu'à l'aide de la commande **crontab -e** sauf si on est vraiment un pro.

2.4.17 Linuxconf

Sujet

- *Mettre à l'heure votre machine*

corrigé

RAS, linuxconf permet de configurer tout sur la machine mais ne permet pas d'automatiser les tâches donc c'est bien mais quand ça marche plus il faut savoir quand même s'en sortir avec les fichiers textes.

2.4.18 Adduser

Sujet

utiliser cette merveilleuse commande pour créer des comptes utilisateur suplementaires

corrigé

ça marche comme suit :

```
[gouinaud@garp    sys]$ adduser -h
adduser: option invalide -- h
usage: adduser [-u uid [-o]] [-g group] [-G group,...]
               [-d home] [-s shell] [-c comment] [-m [-k template]]
               [-f inactive] [-e expire mm/dd/yy]
[-p passwd] [-n] [-r] name
       adduser -D [-g group] [-b base] [-s shell]
               [-f inactive] [-e expire mm/dd/yy]
```

attention, cette commande est en général spécifique à l'UNIX que l'on utilise, ici c'est celle de linux qui est décrite.

2.4.19 installation d'une imprimante réseau

Sujet

A l'aide de printool ajouter une queue d'impresion à votre machine permettant d'acceder à l'imprimante qui est sur FC

corrigé

Lancer printtool, choisir add puis donner le nom de la machine à laquelle est connectée l'imprimante et le nom de la queue d'impression. Choisir dans le menu **restart lpd** et ça doit marcher.

Printtool rajoute en fait une entrée dans le fichier /etc/printcap. Sous Solaris cela ce fait à l'aide de l'outil admintool.

2.4.20 Syslog et les journaux

Sujet

On va aller voir se qui se trame dans /var/log

commande who, what, w et lastog.

recherche des setuid

Sécurisation de la console.

corrigé

dans /var/log :

cron	log du crond	ce qui est fait ou pas fait
dmesg	boot	ou ce qui sort sur la console
httpd/	httpd	répertoire qui contient les logs du serveur web
lastlog	connextions	en telnet,rlogin et ftp
maillog	sendmail	qui écrit où et ce qui arrive pas à partir
messages	les messages du système	ce qui va pas en général
secure	les log de syslog	tout ce qui est connexion su et autres
spooler	les log de lpd	tout ce qui concerne l'imprimante
xferlog	les log de ftpd	tout ce qui se fait, comme put et get en ftp

who donne la liste des utilisateurs connectés, w ou what donne la liste de ce qu'ils font.

Les suid sont des programmes où l'on a fait chmod u+s de façon qu'ils s'exécutent au nom du propriétaire du fichier et non au nom de l'utilisateur courant. Ceux qui sont root sont donc des trous de sécurité potentiels il faut donc y prendre garde.

On les trouve grâce a la commande :

```
find . -perm -4000 -exec ls -al {} \;
\begin{verbatim}

Ce qui donne (par exemple):

\begin{verbatim}
-rwsr-xr-x    1 root     root       471412 May 17  1999 ./usr/bin/kppp
-r-sr-sr-x    1 root     lp          15696 May 11  1999 ./usr/bin/lpq
```

Relevez bien le droit s dans la colonne de gauche. Il est bon d'en faire une liste après l'instal du système et de la refaire de temps en temps pour comparer. Toute présence inexpliquée de ce type de programme est un risque d'atteinte à la sécurité. C'est-à-dire, faut demander conseil !