Le Hubl○t

Journal des tuteurs — Numéro 10 — Décembre 2000

Dans ce numéro, on achève de présenter les commandes Unix de base, et on se lance dans les choses sérieuses en matière de sécurité informatique. On attaque aussi des sujets plus techniques en LaTeX.

À suivre au mois de janvier, le prochain stage informatique : on y parlera de LaTeX, de forum, du logiciel de dessin `xfig`*... et de ce que vous nous suggérerez !*

Comme dans chaque numéro, les conseils de lecture et les bonnes adresses :

- *Certaines pages de notre site sont mentionnées dans les articles : on suppose que toutes les adresses indiquées commencent par* `http://www.eleves.ens.fr:8080`
- *Certains mots sont suivis d'un astérisque*; ils renvoient au lexique en fin de numéro.*
- *Vous pouvez retrouver les anciens numéros du* Hublot *sur notre site :* `http://www.eleves.ens.fr:8080/tuteurs/hublot/`

Nous sommes toujours à votre disposition pour toutes questions et suggestions : `tuteurs@clipper`*. Vous pouvez aussi mettre un mot dans le casier de Marie-Lan Nguyen. Enfin, n'oubliez pas que quand vous tapez* `qui`*, les tuteurs ont leur nom souligné : n'hésitez pas à nous solliciter !*

En pratique

Cette rubrique porte sur l'utilisation pratique des machines Unix et de LaTeX. Ce mois-ci, on présente les commandes pour renommer, déplacer copier et détruire des fichiers. La section « Config conscrits » explique comment faire du copier-coller. système de fenêtrage. Enfin, la rubrique LaTeX explique ce que sont les packages*, et quels sont les plus courants.*

Commandes Unix (fichiers)

On achève ce mois-ci le tour d'horizon des commandes Unix de base, pour la gestion quotidienne des fichiers. Les mois suivants, on verra divers petits logiciels, ainsi que la façon d'enchaîner des commandes, et les « jokers ». Si vous êtes allergiques au clavier, n'oubliez pas qu'il existe aussi des gestionnaires de fichiers à la souris sur les stations (le spartiate `xfm` et le plus joli et plus lourd `kfm`). Cela dit, si vous êtes connectés à distance, vous ne pourrez utiliser que les commandes clavier ; il est donc utile de les connaître...

— **cp (*copy*)** — Sert à faire une copier d'un fichier, selon la syntaxe suivante :

```
brick-~ $ cp fichier nouveau
```

`fichier` est le nom du fichier existant, `nouveau` est le nom de la copie.

— **mv (*move*)** — Cette commande sert à la fois à renommer et à déplacer les fichiers. Pour renommer, la syntaxe est la suivante :

```
brick-~ $ mv fichier nouveau
```

Le premier nom est le nom actuel du fichier, `nouveau` est le nouveau nom du fichier.

Pour déplacer un fichier, on indique le nom du fichier, suivi du nom du répertoire où l'on veut placer le fichier :

```
brick-~ $ mv commentaires DEA/
```

Cette commande place le fichier `commentaires` dans le sous-répertoire `DEA/` de votre compte.

— **rm (*remove*)** — `rm` sert à effacer un fichier, avec la syntaxe suivante :

```
brick-~ $ rm fichier
rm: remove fichier (yes/no)? y
```

À la demande de confirmation, répondez pas `yes` ou simplement `y`. Si vous répondez autre chose, le fichier ne sera pas effacé.

Mises en garde

✘ On a très vite fait de confondre `mv` et `rm`.

✘ `rm` ne fonctionne pas sur le principe de la corbeille des Macs ou des PC Windows, il ne déplace pas les fichiers dans un « endroit » de stockage avant destruction. Un fichier effacé est *vraiment* effacé (mais le SPI fait des sauvegardes toutes les nuits, adressez-vous à eux si vous effacez un fichier par mégarde : `spi@clipper`).

✘ Dans la config conscrits, les commandes `cp`, `mv` et `rm` sont en fait des alias* sur les commandes `cp -i`, `mv -i`, `rm -i` (ou, pour le dire autrement, par défaut, l'option `-i` est mise). C'est l'option *interactive*, qui demande confirmation avant d'écraser un fichier, ou de l'effacer. C'est donc une sécurité.

Mais cette option n'est pas activée partout : le plus souvent, quand vous vous retrouverez sur un système Unix hors de l'ENS, il n'y aura pas cette sécurité, et `rm` effacera vos fichiers sans demander confirmation. Soyez donc prudents...

Ces trois commandes sont expliquées sur le site des tuteurs : `/tuteurs/cours/unix/fichiers.html`. Vous trouverez également des exercices corrigés sur `mv` et `cp` : `/tuteurs/exos/cp_mv.html`.

Le mois prochain, on présentera l'utilitaire `less`, qui sert à regarder des fichiers texte sans les modifier.

Config conscrits : le copier-coller

— **Principe** — Le copier-coller consiste à recopier un bout de texte pour le coller ailleurs, soit dans le même document, soit ailleurs. Les logiciels manipulant du texte comportent des commandes permettant de faire du copier-coller, avec raccourcis claviers ou menus ; on peut aussi utiliser la souris (ce que la souris sélectionne est gardé en mémoire, le « presse-papier » des Macs et de Windows).

On ne va parler ici que du copier-coller à la souris, sans évoquer les commandes spécifiques des éditeurs et des traitements de textes. D'autre part, il ne s'agit que de copier du texte, et rien d'autre.

— **En pratique** — Pour faire du copier-coller, on sélectionne le texte avec le bouton gauche, et on colle avec le bouton du milieu :

* Sélectionner le texte : cliquez avec le bouton gauche de la souris à l'endroit où commence le texte ; tout en gardant le bouton de la souris appuyée, faites-la glisser. Le texte sélectionné est mis en *vidéo inverse*, c'est-à-dire en blanc sur fond noir (et non plus en noir sur fond blanc). Quand l'ensemble du texte à copier est sélectionné, relâchez le bouton gauche de la souris.
* Coller le texte : en règle générale, le texte sera collé à l'endroit où se trouve le curseur. Pour coller le texte sélectionné par la souris, placez le pointeur de la fenêtre sur la fenêtre où l'on veut coller le texte (ce n'est pas nécessaire de placer le pointeur exactement à l'endroit où vous voulez coller le texte, puisque le curseur s'y trouve déjà). Ensuite, appuyez sur le bouton du milieu de la souris. Le texte est inséré.

Un double-clic sur un mot le sélectionne automatiquement. C'est particulièrement utile pour des noms de fichiers, des adresses électroniques, ou encore des URL*. Un triple clic sélectionne toute la ligne. Ça ne sert pas à grand chose, mais c'est rigolo.

— **Logiciels courants** — Voici quelques conseils pour utiliser plus efficacement le copier-coller avec les logiciels courants :

* `pine`, `pico` : rien à dire de spécial... N'oubliez pas l'existence de la commande `^J` (« justifier ») pour reformater un paragraphe, quand les lignes sont de longueur inégale. Pour l'utiliser, placez-vous au début du paragraphe et tapez `^J`.
* `netscape` : vous pouvez récupérer le texte de pages Web en utilisant le copier-coller. Il est cependant souvent plus simple de télécharger le document, en sélectionnant `Save As` dans le menu `File`.

  Pour copier-coller une URL dans le cadre `Location`, commencez par effacer l'adresse de la page courante (sélectionnez-là avec le bouton gauche et appuyez sur `Backspace`). Ensuite, coller la nouvelle adresse dans le cadre.

∗ `nedit` : le principe « sélectionner avec le bouton gauche, coller avec le bouton du milieu » fonctionne ; le texte sera collé à l'endroit où vous cliquez avec le bouton du milieu (comme avec `emacs`).

  Cet éditeur propose aussi un menu contextuel accessible avec le bouton droit, avec entre autres `Cut`, `Copy` et `Paste`.

∗ `emacs` : sélectionner le texte avec le bouton gauche. Le texte sera collé à l'endroit où vous cliquerez avec le bouton du milieu. Vous n'avez donc pas à placer préalablement le curseur à l'endroit voulu.

∗ `vim` : il faut être en mode insertion pour coller un bout de texte avec la souris.

Pour les éditeurs en général, n'oubliez pas qu'ils incluent des commandes pour insérer des fichiers extérieurs (`^R` avec `pico` donc `pine`, `^I` ou `Include File` dans le menu `File` pour `nedit`, `:r` *`fichier`* avec `vim`, `^XI` ou `Include File` dans le menu `File` de `emacs`).

Le mois prochain, on parlera des raccourcis clavier de la config conscrits.

LaTeX : les packages

Les *packages* sont des ensembles de commandes qui permettent d'étendre les fonctionnalités de LaTeX ou de changer son comportement. Certains s'appliquent automatiquement à l'ensemble du texte, d'autres rendent simplement disponibles des commandes particulières. Ils correspondent à des fichiers qui ont un suffixe `.sty`.

Note : « package » est un mot anglais, et comme tel, nous devrions le mettre en italiques dans tout ce texte... Mais comme l'alternative francophone « paquetage » est très rare et guère plus explicite, nous assumons l'usage de package en français dans le texte.

— **Utilisation de packages** — Pour charger un package, on utilise la commande `\usepackage{`*`nom`*`}`, où *nom* est le nom du package. Cette commande se place dans le préambule du document, avant le `\begin{document}`.

Certains packages prennent des options ; celles-ci sont écrites entre crochets, avant le nom du package lui-même ; quand il y a plusieurs options, on les met à la suite, séparées par des virgules :

```
\usepackage[option1,option2]{package}
```

— **Les packages indispensables** — Pour garantir que vos accents passent bien d'une machine à l'autre, et que votre DVI aura des accents, et pour franciser vos documents (titres prédéfinis, typographie, etc), voici les trois packages à utiliser dans tous vos fichiers :

```
\usepackage[francais]{babel}
\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
```

`inputenc`, c'est l'encodage des accents du fichier source (le fichier `.tex`). On lui donne l'option `latin1`, car l'encodage standard des accents occidentaux est l'isolatin1. C'est le même principe pour `fontenc` (encodage de la fonte dans le DVI).

`babel` est un système qui permet de gérer une ou plusieurs langues dans un même document ; ici, on lui passe l'option `francais`. Il connaît un certain nombre de langues (`bahasa`, `breton`, `catalan`, `croatian`, `czech`, `danish`, `dutch`, `english`, `esperant`, `estonian`, `finnish`, `galician`, `german`, `greek`, `irish`, `italian`, `lsorbian`, `magyar`, `norsk`, `polish`, `portuges`, `romanian`, `russian`, `scottish`, `slovak`, `slovene`, `spanish`, `swedish`, `turkish`, `usorbian`, `welsh`, ouf).

Note importante : c'est bien `francais` et pas `french` que l'on vous conseille ; `french` n'est pas tout à fait compatible avec le reste de Babel, et il redéfinit des choses, comme l'espacement à l'intérieur des listes, qui n'ont rien à voir avec la francisation. Les Français qui l'ont écrit ont un peu trop intériorisé la notion d'« exception culturelle »...

Ce que gère Babel, ce sont les titres des parties et des tables générées automatiquement, la date écrite dans la langue choisie (commande `\today`), et aussi certaines macros spécifiques. Pour le français, en voici quelques-unes : `1\ier` : 1$^{\text{er}}$; `2\ieme` : 2$^{\text{e}}$; `1\iere` : 1$^{\text{re}}$.

Quand on lui passe plusieurs langues en argument, celle qui domine dans le texte est la dernière indiquée. Des commandes spécifiques servent à passer dans l'autre langue ; par exemple, `\textgreek{}` pour mettre des mots en grec, ou `\cyr{}` pour mettre des mots en cyrillique (en supposant que vous avez indiqué `greek` ou `russian` dans les options de `babel`).

— **Divers packages** — Il en existe un tas, qui font un tas de choses... On va voir ici les plus courants, et les plus utiles :

`amsmath,amsfonts,amssymb` Ces trois packages définissent des commandes vraiment indispensables pour taper des maths avec LaTeX. Et également quelques symboles qui peuvent servir ailleurs. AMS, c'est l'*American Mathematical Society*.

`epsfig` Pour inclure des images au format EPS (PostScript encapsulé), avec la commande `\epsfig{}`. On en parlera au mois de février.

`multicol` Pour faire un document en plusieurs colonnes. Pour cela, on utilise l'environnement `multicols` (avec un « s », piège diabolique). Par exemple, pour un document écrit sur trois colonnes :

```
\begin{multicols}{3}
Le texte...
\end{multicols}
```

`url` Pour afficher correctement des adresses électroniques et des URL.

`vmargin` Pour redéfinir la taille de la page. On le verra plus précisément dans le prochain numéro.

— **Pour en savoir plus** — Bien sûr, il ne s'agit que d'une très petite partie de tous les packages existants. En fait, chaque fois que quelqu'un écrit une fonctionnalité supplémentaire pour LaTeX (par exemple, `supertabular` pour faire des tableaux sur plusieurs pages), il écrit un package.

Le *LaTeX companion*, dont on trouve normalement un exemplaire (souvent en très mauvais état hélas) dans toutes les salles informatiques, contient une énumération beaucoup plus complète, ainsi qu'une description beaucoup plus détaillée des possibilités.

Notons qu'il y a aussi quelques notices pour les packages les plus utiles installés chez *gourous*, dans le répertoire `/usr/local/util/tex/doc`.

Le mois prochain, on expliquera comment régler la taille de la page.

▷ Nicolas George, Émilia Robin

Comment... utiliser les disquettes

Cette rubrique a pour but de présenter diverses utilisations des machines Unix, en indiquant à chaque fois les logiciels disponibles, et en présentant certains d'entre eux. Ce mois-ci, on va expliquer ce que sont les disquettes, et comment on s'en sert sur les stations.

L'obscur objet disquette

Voyons un peu la longue histoire des disquettes, aussi appelées *floppy disk*, c'est-à-dire disques souples. J'en vois déja me dire que les disquettes que vous utilisez sont en plastique rigide, mais ce terme désigne en fait le disque magnétique souple qui se trouve à l'intérieur de l'enveloppe rigide. Il s'oppose aux disques durs (*hard disk* en anglais), dans lesquels les disques magnétiques (appelés des plateaux) sont rigides.

— **La préhistoire** — Il y a bien longtemps, il n´existait pas de CD-ROM, ni de disques durs ou même de disquettes. On utilisait des cartes perforées. C'était un moyen de stockage et de transmission peu efficace et peu pratique, mais il n'y avait rien d'autre.

On a alors eu l'idée d'utiliser le même principe que pour le stockage de musique, à savoir une surface magnétisée, constituée d'une multitude de minuscules aimants. Ces aimants peuvent être orientés à volonté, et ils gardent alors cette orientation, stockant ainsi de l'information. On a commencé par garder la même forme que pour les cassettes, c'est à dire des bandes magnétiques enroulées sur des bobines, et qu'on déroule pour lire l'information dessus. Ça marchait, mais c'était assez lent, et pour lire les données placées à un endroit précis il fallait dérouler toute la bobine pour atteindre le bon endroit.

Cela avait quand même quelques bons côtés, comme une capacité assez importante. D'ailleurs, le principe est encore utilisé aujourd'hui. Par exemple les sauvegardes des comptes sur clipper sont faites sur des cassettes DAT (Digital Audio Tape, qui sont à la base des cassettes utilisées pour l'enregistrement numérique de musique). D'autres systèmes utilisent des cassettes de caméscope.

— **Et la disquette fut** — Pour pallier ces défauts, on a donc pris cette fois comme surface magnétique un disque souple, qui tournait autour de son axe, qu'on a inséré dans une pochette en plastique souple, laquelle gardait des ouvertures pour accéder à la surface magnétique. De cette manière, on pouvait accéder (presque) instantanément à n'importe quelle partie du disque en question, simplement en le faisant tourner et en déplaçant la tête de lecture le long d'un rayon.

C'est ainsi qu'est apparue la première disquette, qui était alors un mastodonte, connu sous le nom de disquette 8" (lire « 8 pouces », c'est à dire un peu plus de 20 centimètres de large, le pouce mesurant exactement 2,54 centimètres) Ce monstre possédait une capacité absolument délirante pour l'époque, à savoir 80 Ko.

Plus tard sont apparues les disquettes 5"1/4 (cinq pouce un quart), qui avaient à la base une capacité de 180 Ko par face, et qui furent utilisées par la majorité des micro-ordinateurs du début des années 1980 (IBM PC, Commodore 64...).

— **Format actuel** — En 1982, Sony créa un nouveau type de disquette qui devait très vite s'imposer, la disquette 3"1/2 (trois pouce et demi), toujours en usage actuellement, et donc la capacité pouvait alors aller jusqu'à 720 Ko. Outre sa taille réduite et sa plus grande capacité, elle comportait deux innovations :

1. Le disque magnétique était protégé par une enveloppe rigide, ce qui rendait les disquettes moins fragiles.
2. Les ouvertures pratiquées dans ces enveloppes étaient recouvertes par des caches métalliques, qui coulissaient automatiquement quand on introduisait la disquette dans le lecteur. De cette manière, on ne pouvait plus accidentellement abîmer ou salir la surface magnétique.

Du fait de leur plus grande capacité et fiablité (même si la fiabilité d'une disquette n'est jamais très élevée), ces disquettes furent rapidement adoptées par Apple (dont le Macintosh fut la première machine à l'utiliser), puis par Atari et Commodore (dans les Ataris ST et les Amigas), puis arrivèrent sur les PC avec le PS/2 d'IBM, et devinrent le standard.

Les disquettes 5"1/4 et 3"1/2 ont vu par la suite leurs capacités portées à 1,2 Mo et 1,44 Mo (en 1984 et 1987), et ce sont les capacités utilisées actuellement. Il y a eu une tentative infructueuse pour introduire des disquettes 3"1/2 à 2,88 Mo, mais devant le peu d'enthousiasme du public, cela fut abandonné.

Signalons également le format 3", qui ressemble au format 3"1/2, avec une capacité de 180 Ko par face. Très peu répandu, il ne fut utilisé que par quelques machines (Stratos ou Téléstrat, et surtout l'Amstrad CPC), et à cause du succès soudain de l'Amstrad, ces disquettes connurent des périodes de pénurie assez mémorables pendant lesquelles les prix s'envolèrent.

Hors de la disquette, quel salut ?

La première chose à savoir quand on utilise des disquettes est leur manque total de fiabilité : quand on copie un fichier sur une disquette, on n'est jamais sûr de pouvoir le relire. C'est pourquoi il est fortement déconseillé d'utiliser les disquettes comme sauvegarde principale ou comme archive ; et quand l'on transfère des fichiers à l'aide de disquettes, il est préferable de tout transférer en double sur des disquettes séparées pour avoir des chances raisonnables d'avoir au moins un exemplaire lisible de chaque fichier.

Une disquette est en effet très sensible aux conditions extérieures, à savoir température, humidité et surtout champs magnétiques. Bon nombre de disquettes ont succombé à un séjour au soleil, dans une pièce humide, ou simplement à un passage à côté de haut-parleurs mal isolés...

De plus, une capacité de 1,44 Mo peut se réveler insuffisante, et s'il existe des outils pour répartir un fichier entre plusieurs disquettes, cela n'est jamais pratique.

Aussi les disquettes possèdent maintenant un certain nombre d'alternatives :

* Pour les transferts, les **transferts réseau** sont beaucoup plus efficaces et rapides, en particulier pour les portables, susceptibles d'être déplacés facilement et de pouvoir se brancher sur une prise réseau libre (il existe une telle prise en salle S, destinée uniquement à cet usage).
* Le **format Zip**, qui est similaire à une disquette, mais avec une capacité largement supérieure (100 Mo), et une fiabilité largement supérieure.
* La **gravure sur CD**, qui devient de plus en plus courante au fur et à mesure que les graveurs se répandent. Un CD ne peut être gravé qu'une seule fois, mais étant donné sa capacité (660 Mo) et le prix d'un CD vierge, ce n'est pas un gros problème. Signalons l'existence de CD réinscriptibles, mais ils coûtent relativement cher et ne peuvent être lus que par les graveurs et certains lecteurs.

L'élémentaire prudence

On reprend ici un article paru dans le numéro 7 du Hublot*, en mai 2000.*

Les plantages des ordinateurs fonctionnant sous Windows ou MacOS peuvent entraîner des pertes irréparables de données. Sachez que vous ne pourrez pas vous retourner contre le fabricant du logiciel, car la licence vous avez acceptée possède une clause qui exclut toute responsabilité pour les dégâts que pourrait causer le produit. D'autre part, quel que soit le système, un disque dur est un dispositif électromécanique pouvant être sujet à des pannes matérielles. Vous devez donc faire des sauvegardes régulières de vos données, au moins tous les mois[1]. Voici quelques conseils :

☞ Ne laissez jamais les sauvegardes dans la même pièce que l'ordinateur, en cas d'incendie vous perdiez tout (ne riez pas, c'est arrivé au département de physique). Sans parler des inondations venues du plafond, ou de la tasse de café...

☞ Rangez vos supports amovibles (disquettes, bandes, ZIP) loin de toute source de perturbations électromagnétiques, comme c'est d'ailleurs expliqué sur les boîtes de disquettes.

☞ Certains laissent avec confiance leurs données sur les Macs ou PC sous Windows en libre accès à l'École ; outre l'absence de confidentialité, n'importe quelle personne mal intentionnée peut les modifier, voire tout effacer (c'est impossible sur les Suns et les PC sous Linux, sauf si vous oubliez de vous déloguer). Même sans mauvaises intentions, il peut s'agir de quelqu'un qui a besoin de libérer de la place pour utiliser le scanner, par exemple. Ayez donc toujours une sauvegarde de votre travail !

☞ D'autre part, rappelez-vous bien qu'une disquette n'est pas un support fiable : faites toujours vos transferts en double exemplaire, sur deux disquettes ; à fortiori, **faites toujours au moins DEUX sauvegardes**. Ça nous fait froid dans le dos et mal au cœur de voir des affichettes « Perdu disquette orange très importante le 12 mars », et de savoir qu'il y a des gens qui confient à un seul support leur travail de l'année, ou même simplement de la semaine...

Enfin, sachez qu'à l'ENS, le Service de Prestations Informatiques effectue une sauvegarde des comptes des élèves (sur les Suns et les PC sous Linux) toutes les nuits, sur des bandes magnétiques.

Utiliser les disquettes sous Unix

— **Je dois vraiment utiliser ça ?** — On vous montre ici une façon d'utiliser les disquettes qui peut paraître barbare et pas simple. Mais des commandes

[1] Une sauvegarde tous les mois, ça veut dire que vous vous autorisez à perdre, éventuellement, un mois de travail.

au clavier sont le seul moyen de *partager* un lecteur de disquettes entre plusieurs machines, et de pouvoir l'utiliser à distance. Les Suns, traditionnellement, n'avaient pas de lecteur, seule une machine dans la salle en avait un, utilisé par tout le monde.

En revanche, les interfaces graphiques pour Unix, comme KDE, que vous pouvez avoir sur votre PC Linux, permettent une gestion à la souris des disquettes, pour copier des fichiers ou les mettre sur la disquette. Unix ne se réduit pas à ce que vous en voyez à l'École...

— **Les *mtools*** — Pour utiliser les disquettes au format PC sur les stations Sun, il existe une ensemble de commande appelées *mtools*. Il faut savoir qu'à l'ENS les machines Unix ne sont pas toutes équipées d'un lecteur de disquettes, mais on peut se servir du lecteur d'une station sans être physiquement devant. Pour cela, il suffit d'introduire la disquette dans le lecteur (en essayant de ne pas trop déranger la personne devant le clavier), et de se connecter à distance sur la machine en question.

Pour indication, voici la liste des machines équipées de lecteurs de disquettes en salle S (galion, kayak, bireme, brick, jonque, chaland et drakkar), au 46 (belandre et barge), en Infi (jangada, plate, aviso, allege, buyse) à Jourdan (vedette, chaloupe, nave, selandre, gondole), montrouge (ponton, polacre). Cela évolue en fonction des nouvelles machines installées (les Ultra 5 ont toutes des lecteurs).

Une fois la disquette dans le lecteur, voici quelques commandes utilisables (il peut y avoir des fichiers et des répertoires sur une disquette) :

`mdir`	**Affiche le contenu de la disquette**
`mcd toto`	Se déplacer dans le répertoire `toto` de la disquette
`mmd toto`	Crée le répertoire `toto`
`mrd toto`	Supprime le répertoire `toto`, à condition que celui-ci soit vide
`mcopy a:bla .`	**Copie le fichier `bla` de la disquette vers la station dans le répertoire courant**
`mcopy bla a:`	**Copie le fichier `bla` vers la disquette**
`mdel bla`	Supprime le fichier `bla`
`mdeltree toto`	Supprime le répertoire `toto`, ainsi que tous les sous-répertoires et fichiers qu'il contient (à utiliser avec précaution !)
`mformat a:`	Formate la disquette, donc efface tout.

Ceux qui connaissent les commandes MS-DOS les auront reconnues dans les mtools, précédées d'un « m » (pourquoi un « m », personne ne le sait). Le lecteur de disquettes est désigné par `a:`, et la syntaxe pour les répertoires est la syntaxe habituelle, c'est-à-dire que les répertoires sont indiqués par `/`, et `..` correspond au répertoire précédent.

L'utilisation des disquettes est détaillée sur le site des tuteurs, avec des exemples : `/tuteurs/cours/unix/disquettes.html`.

▷ Gilles Radenne

Internet

Ce mois-ci, nous nous étendons sur le Web, ses principes et ses fonctionnement, des conseils pour identifier les bons sites, et une présentation des moteurs de recherche. Dans la rubrique « Forum », on continue à expliquer comment lire forum, en abordant ce mois-ci la façon d'aller de conti en conti.

Le Web

Le Web, c'est le truc dont on parle à la télé : les pages, les machins multimédia, les « savoirs en ligne », etc. On en parle tellement qu'on dit « Internet » à la place de « Web » (« Je surfe sur Internet », « J'ai trouvé un super site Internet »). Pire encore, il y a des gens qui confondent le logiciel utilisé pour naviguer sur le Web, et le réseau lui-même (ce qui donne des « Tiens, je ne connais pas cette version d'Internet »).

— **Un aspect d'Internet** — Le Web est l'un des aspects d'Internet. Internet, c'est la mise en réseau mondiale d'ordinateurs. Ça implique une infrastructure matérielle (câbles, routeurs...), et une série de protocoles pour faire transiter les informations. Par exemple, le Web utilise le protocole HTTP (*hypertext transfert protocol*, protocole de transfert des fichiers hypertexte).

Mais Internet c'est aussi le courrier électronique, le transfert de fichiers, la connexion à distance (telnet), les news... Pour chacune de ces fonctions, on utilise un logiciel : navigateur pour le Web, logiciel de courrier, etc.

Si vous voulez, Internet c'est comme une route que les gens utilisent pour différentes fonctions (transport de marchandises, vacances, trajet boulot-dodo, balade à vélo...), et ces fonctions c'est le courrier, le Web, etc.

— **Principe du Web**[2] — Sur le Web, une adresse s'appelle une URL (*Uniform Resource Locator*). Prenons pour exemple l'adresse du serveur des élèves de l'ENS : `http://www.eleves.ens.fr:8080/`. Le `http` au début indique le protocole à utiliser, et signale qu'il s'agit d'une page Web (cependant, toutes les pages Web ne sont pas servies par HTTP). Le `.fr` est un nom de domaine, qui indique ici que la page se trouve en droit en France (mais il se pourrait que le serveur, en tant que machine physique, matérielle, soit aux États-Unis). `www.eleves.ens` désigne en réalité une machine qui héberge le site des élèves de l'institution ENS. Le `8080` désigne un numéro de port.

Enfin, le `/` (*slash*) qui termine l'adresse indique qu'on se rend au point de départ de l'arborescence du site, en pratique la page d'accueil.

Tout ceci indique la machine à contacter pour rejoindre la page Web. Ensuite, il peut y avoir d'autres informations quelle page précisément on demande à voir. Par exemple, `http://www.eleves.ens.fr:8080/COF/CineClub/` correspond la page d'accueil du Ciné-Club de l'ENS, elle-même un sous-ensemble logique de `COF/`.

Quand vous sélectionnez un lien, votre ordinateur envoie une requête sur le serveur désigné par l'URL. Ce serveur renvoie la page demandée par votre navigateur (« client » du serveur).

L'ensemble de ces données circulent sur des liens physiques, les câbles, qui ont un débit maximal. La réponse d'un serveur physiquement très éloigné mettra un peu plus de temps à arriver ; une grosse quantité de données mettra plus de temps à arriver qu'une petite quantité ; enfin, à certaines heures, le réseau est encombré par l'ensemble des données transférées par tous les utilisateurs connectés à ce moment-là. C'est ce qui explique que parfois, une page soit longue à charger.

— **Noms de domaines** — Il y a un certain nombre de noms de domaine. Une partie indique un pays, dont dépend en droit le serveur : `.fr` pour les serveurs français, `.es` pour l'Espagne, `.ru` pour la Russie, `.jp` pour le Japon, etc.

Il en existe d'autres, comme `.edu` pour les universités américaines, `.org` pour les institutions à but non lucratif, `.gov` pour le gouvernement américain. `.int` est utilisé en particulier par les organisations établies par traités internationaux, par exemple l'Union européenne.

De nouveaux noms de domaines sont apparus avec l'engouement pour le Web. Par exemple, `.com` désignait à l'origine des sites d'entreprises, et `.net` des fournisseurs d'accès au réseau. Mais AOL, fournisseur d'accès, a comme adresse `aol.com` (pour des raisons historiques, certes). D'autre part, des entreprises peuvent avoir un site avec un suffixe `.net`, à cause de leur fournisseur d'accès.

C'est aussi une question de modes : `.com` est à la mode en ce moment, donc les sites veulent absolument un suffixe en `.com`, y compris si c'est aberrant. Je pense ici au Mur pour la Paix (au Champ de Mars). Le monument a un site Web, sur lequel on peut déposer des messages, qui seront ensuite affichés sur les écrans du monument. Et leur adresse est `http://www.murpourlapaix.com/`, alors que c'est un suffixe `.org`, réservé aux organisations à but non lucratif, qui s'imposait...

Un autre exemple : le suffixe `.tv` est théoriquement réservé aux îles Tuvalu ; mais le Web est quelque chose de rentable, et elles ont vendu quelques noms de domaines à des chaînes de télévision qui voulaient une adresse en `.tv`...

Le site des tuteurs propose un certain nombre de cours sur le Web : comment écrire les pages Web, y insérer des images, concevoir son site (`/tuteurs/internet/html/`, comment utiliser Netscape (cours du *Hublot* l'an dernier : `/tuteurs/hublot/1999-2000.html`).

Un « bon » site Web ?

Qu'est-ce qu'un « bon » site ? En laissant de côté le top site méga cool avec mes photos de vacances, on peut essayer de dresser une liste de choses qu'un site sérieux se doit d'avoir.

On peut d'emblée résumer la qualité principale d'un tel site sous le mot : **lisibilité**. Et dans le monde des pages Web, ce mot contient la notion d'accessibilité. On en tire les implications suivantes : la page doit pouvoir être lue par n'importe quel navigateur, les informations doivent être précises, la mise à jour doit être régulière, et enfin il doit être facile de circuler non seulement dans le site mais aussi vers l'extérieur.

C'est pourquoi les sites qui respectent ces principes sont agréables et intéressants à consulter. Les « bons » sites possèdent en général 7 traits distinctifs.

1. Le lien permettant de **revenir à la page d'accueil**, ou « *Home* » : en général, il se trouve sous la forme d'une icône avec le logo du propriétaire du site, et se trouve conventionnellement en haut à gauche de la page Web ; on le trouve parfois en toutes lettres (« Home », « Retour à la page d'accueil ») en bas au centre de la page. Un tel lien permet de commencer une nouvelle recherche dans l'arborescence du site.

2. Le **plan du site** (« *Site map* », « *Site index* ») : très utile pour se repérer, il permet d'avoir une vue d'ensemble du site, et d'orienter sa recherche en fonction des catégories (par ex. : météo, loisirs, arts, sports, info).

3. Un **moteur de recherche** intégré au site (« *searchable* ») permet un accès rapide à un concept, permet d'être fixé rapidement sur nos attentes vis-à-vis de ce site.

[2] On reprend ici un article paru dans le numéro 5 du *Hublot*, en mars 2000.

4. La **rubrique *help*/FAQ** (*frequently asked questions*, « foire aux questions » en français) ou « Présentation » : il peut être utile de commencer la consultation d'un site par cet endroit. On peut y trouver le mode d'emploi du site, son fonctionnement, la périodicité de mise à jour, le nom du propriétaire, les liens.

5. Quoi de neuf ? / « Nouveau » (*What's new*) : indique les dernières modifications de la page, notamment les nouveaux services qui viennent d'être ajoutés, ou encore une réorganisation du site. Il est possible d'être tenu au courant de l'actualité d'un site grâce à une lettre d'information (« *newsletter* ») par courrier électronique. Pour cela, il suffit de s'inscrire en laissant son adresse électronique (voir par exemple le site du Monde diplomatique : `http://www.monde-diplomatique.fr/`).

6. Les **liens** (*links*) : si par malheur vous n'avez pas trouvé ce que vous cherchiez sur un site, allez visiter ses liens, qui normalement vous dirigeront vers des thèmes similaires à ceux qui vous ont amené là.

7. Contact (*feedback*) : c'est la rubrique qui permet de prendre contact avec les concepteurs du site. Attention ! Adressez-vous à la bonne personne et à bon escient. La personne qui a rédigé le contenu de la page (par ex., un article sur les élections présidentielles américaines) n'est pas (pas forcément en tout cas) la même que celle qui a mis la page en langage HTML. À la première personne vous poserez des questions concernant l'article, à la deuxième vous exprimerez vos plaintes ou vos louanges à propos de la mise en page. Dans ce dernier cas, il est recommandé de lire les FAQ au préalable, pour ne pas être le 37e à poser la même question.

▷ Yann Tholoniat

NDLR : on peut ajouter un huitième critère :« lisible » veut le plus souvent dire « sobre »... Pas de débauche de couleurs, pas de trucs qui clignotent, pas de choses écrites en petit dans tous les sens, pas de publicités dans tous les coins, pas de « gratuit ! » ou « pas cher ! » aux quatre coins de la page, etc.

Les moteurs de recherche

Il est presque impossible de trouver une information sur le Web sans utiliser de moteurs de recherche. Ce sont des sites qui permettent d'interroger des bases de données constituées par des robots qui parcourent le Web et référencent les pages visitées. Mais il faut bien être conscient que les moteurs de recherche ne référencent au mieux que 15% des pages existantes.

Voici un exemple d'utilisation du moteur de recherche Altavista (`http://www.altavista.com`). Je demande « Iran », sans plus de détails ; j'obtiens 430 665 réponses. Je demande alors que la recherche se fasse uniquement sur des fichiers en français ; je n'ai plus que 12 085 réponses. Ensuite, je raffine la demande, en la précisant progressivement ; voici les résultats :

`Iran`	: 12 085
`+Iran +1979`	: 1 722
`+Iran +1979 +révolution`	: 616
`+Iran +1979 +révolution +Shah`	: 48

Le signe `+` veut dire que le mot qu'il précède (sans espace) doit obligatoirement figurer dans la page référencée. Vous pouvez également utiliser le signe `-`, pour exclure impérativement le mot des pages recherchées. En règle générale, quoi que l'on recherche, il est utile d'ajouter `-hotel -restaurant`... Enfin, si vous recherchez une expression telle quelle, mettez des guillemets américains autour (`"guerre et paix"` par exemple). Ces signes s'appellent des opérateurs booléens.

Raffinez le plus possible vos requêtes : si vous avez plus d'une centaine de pages à consulter, vous ne vous en sortirez pas. D'autre part, soyez conscients que votre requête sera nécessairement ambiguê, même si vous ne vous en rendez pas compte : une recherche sur « cirque » trouvera « arts du cirque » comme « cirque glaciaire ».

Tous les moteurs de recherche ne s'utilisent pas comme Altavista. Avec Google (`http://www.google.com/`), il est inutile de mettre des `+`. Yahoo (`http://fr.yahoo.com/` pour le site français) classe ses adresses selon une arborescence. Lycos (`http://www.lycos.fr/`) utilise les opérateurs booléens et aussi une arborescence.

Le forum : lire les contis

Nous avons présenté le mois dernier les contis (groupes de discussion) existant dans forum. Ce mois-ci, on va voir comment lire forum, au delà de la commande de base (barre d'espace) expliquée dans le numéro d'octobre.

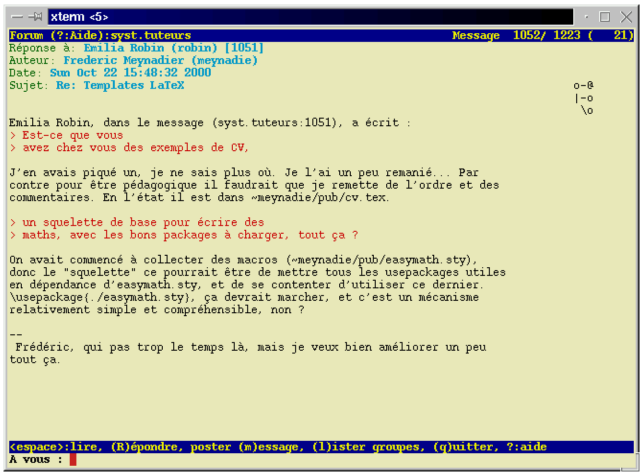
— **Lancer forum** — Pour lancer forum, tapez `forum`, cliquez sur l'icône forum ou sur le bouton « Lire forum » de la fenêtre d'information (`xforumco`, conscrits 2000). Si vous voulez lire un conti spécifique, par exemple `kfet`, il suffit de taper :

```
brick-~ $ forum kfet
```

Forum se présente comme ceci : une barre avec le nom du groupe de discussion, et à droite le numéro du message, suivi du nombre total de messages, et entre parenthèses le nombre de messages non lus dans le groupe de discussion.

En bas, vous avez une autre barre avec les commandes les plus courantes : barre d'espace pour aller de message en message, « `R` » pour répondre à un message, « `m` » pour en poster un, et « `q` » pour quitter. Sous cette barre, vous avez un prompt « À vous », où s'affichent les commandes que vous tapez.

Le message lui-même ressemble à un courrier électronique, avec le nom de l'auteur, la date et le sujet du courrier, éventuellement le nom de la personne à qui l'auteur répond, et parfois d'autres informations :



Une fois le message lu, appuyez sur la barre d'espace pour aller au prochain message. Quand tous les messages du conti sont lus, forum affiche « Fin du newsgroup ». Si vous rappuyez sur la barre d'espace, forum vous place sur le prochain conti auquel vous êtes abonné, et dans lequel il vous reste des messages non lus. Quand tout est lu, forum affiche « Rien de nouveau ».

Vous n'avez plus qu'à taper « `q` » pour quitter forum. Lire forum se résume donc à deux touches : la barre d'espace pour lire, et « `q` » pour quitter.

— **Enfilades et discussions** — Les messages de forum sont des discussions : un premier message a des réponses, d'autres répondent à ces réponses, et cela donne ce que l'on appelle une enfilade (*thread* en anglais). Certains messages ont plusieurs réponses, et de fil en aiguille la conversation dérive souvent sur d'autres sujets.

Le schéma (ou arbre) de la discussion est représenté en haut à droite de l'écran : les « `O` » majuscules représentent les messages encore à lire, les « `o` » minuscules ceux qui sont lus ; l'arobase « `@` » représente le message sur lequel vous vous trouvez. On peut se déplacer dans l'arbre avec les flèches, et taper sur « Entrée » pour lire le message ainsi sélectionné.

Enfin, on peut utiliser la touche « `N` » pour avoir une vue plus grande de l'arbre de discussion (refaire « `N` » pour revenir à la vue normale).

— **Ordre de lecture** — Quand vous lisez forum à la barre d'espace, vous lisez les messages dans l'ordre de la discussion. Quand celle-ci est terminée, vous passez à une autre discussion.

À l'inverse, la touche « Entrée » affiche les messages dans l'ordre de leur publication, indépendamment de leur ordre logique.

— **Aller dans un conti (1)** — Pour se rendre dans un conti donné, tapez « `G` » suivi du nom du conti et de la touche « Entrée ». S'il y a plusieurs possibilités (par exemple, les contis d'annonces), le programme proposera un menu ; utilisez les flèches du clavier pour vous déplacer dans le menu et appuyez sur « Entrée » pour sélectionner le conti à lire.

— **S'abonner, se désabonner** — Une fois dans un conti (ou dans la liste de ceux-ci) on peut utiliser « A » pour s'abonner et « D » pour se désabonner.

Quand on s'abonne à un nouveau conti, on risque de trouver qu'il y a un nombre considérable de messages non encore lus ; les conscrits 2000 sont chanceux, ils n'auront à lire que les messages vieux de sept jours au plus (ne vous laissez pas impressionner par ce qu'affirme `xforumco` sur les 1 420 messages non lus).

Cela dit, si vous ne voulez pas lire tous les messages en retard (surtout si vous n'êtes pas conscrit), utilisez la touche « `z` » (comme zapper), suivie d'« Entrée », pour marquer tous les messages comme lus.

— **Obtenir la liste des contis** — On peut avoir la liste des contis en tapant « `L` » suivi d'« Entrée ». Plusieurs listes s'affichent à la suite :

1. Dans la première, vous avez la liste des groupes auxquels vous êtes abonné, ainsi que ceux où vous avez déjà lu au moins un message. Enfin, les nouveaux groupes créés figurent également dans cette liste.
2. La seconde liste est la liste de tous les groupes existants, sauf ceux de la première liste.

Pour quitter ces listes, tapez `q` jusqu'à récupérer le prompt « À vous » en bas d'écran. S'il y a deux listes, c'est pour avoir plus rapidement accès à celle des contis que vous fréquentez habituellement.

Vous remarquez des lettres devant les noms de certains contis : un « `T` » veut dire que vous êtes abonné (vous lisez <u>t</u>out ce conti), un « `U` » veut dire que vous êtes désabonné (*unsubscribe*)[3]. Quand il n'y a aucune lettre, c'est que vous n'avez jamais mis les pieds dans ce conti. Pour les promotions antérieures, ces marqueurs peuvent être différents.

— **Aller dans un conti (2)** — Maintenant que vous connaissez les deux listes de contis, sachez que vous pouvez les utiliser pour vous rendre dans un conti donné : sélectionnez-le avec les flèches et tapez « Entrée ».

Barre d'espace `N`	Aller au prochain message Afficher/cacher l'arbre de la discussion

`L`
`G` *`conti`*
`z`
	Afficher les listes de contis
	Aller dans ce conti
	Zapper tous les messages du conti

`A` `D`	S'abonner au conti Se désabonner du conti
`R` `m`	Répondre Poster un message
`q`	Quitter

Le mois prochain, on expliquera comment poster dans forum.

▷ David Madore, Émilia Robin

[3] On l'a déjà dit ailleurs, on le redit : il y a une certaine part d'arbitraire, et aussi d'habitudes antérieures, dans le choix des commandes de forum, de l'emplacement des contis, ou encore ici des marqueurs.

Qu'est-ce qu'un ordinateur ? (3)

Ce cours explique progressivement ce qu'est un ordinateur. Le mois dernier, on a vu en quoi consistait la mémoire, et comment l'information circulait à l'intérieur de la bête. Ce mois-ci, on va voir à quoi sert le processeur.

« Grand Maître, maintenant que nous savons où est l'information, et comment elle voyage, où donc va-t-elle ? »

« Elle affronte son destin, jeune apprenti. Le sort de toute information est d'être répétée, amplifiée, déformée, cuisinée et pressée jusqu'à ce que sa substantifique moëlle en soit extraite. Et le morceau responsable de cela, c'est le processeur. »

Vous me ferez vingt pompes

« Il n'est point de domaine de la pensée qui échappe à la logique militaire. C'est en vertu de cet adage que l'informatique est universelle et le champ du manipulable par ordinateur est étendu (et, réciproquement, les choses qui échappent aux militaires sont aussi celles à propos desquelles l'ordinateur n'est d'aucun secours). Car le processeur est le vrai tâcheron de l'affaire, le composant qui sait faire le plus de choses et qui a le moins d'initiative.

Le processeur, souvent dit central, ou encore CPU (*Central Processing Unit*), est une puce de format classique (c'est-à-dire un peu de sable dans une boîte en plastique), et il est caché sous un amoncellement de ventilateurs et de circonvolutions métalliques (structure destinée à le refroidir, car il chauffe beaucoup). Son boulot : le règlement, tout le règlement, rien que le règlement. Il comprend des ordres simples, ordres qui sont codables par des 0 et 1, et que le processeur va chercher en mémoire.

Chaque ordre, il l'exécute à la lettre, sans chercher à comprendre le sens de cet ordre, si tant est qu'il en ait un. La machine n'a pas d'objection a afficher que deux et deux font cinq. Elle peut le faire plusieurs millions de fois par seconde, si on lui demande. Elle peut aussi calculer que deux et deux font quatre si un tel calcul est ordonné. Elle n'est ni psychotique, ni névrosée.

Les ordres compréhensibles par le processeur sont de deux types : ceux qui demandent d'effectuer un vrai travail (additionner deux nombres, par exemple), et ceux qui servent à faire tourner la machine, c'est-à-dire de l'administration interne ; et, comme sa grande sœur en uniforme, le processeur passe beaucoup de temps à s'occuper de lui-même. Plus précisément, voyons au juste quels sont les ordres possibles.

Le processeur peut aller chercher des données en mémoire ; c'est-à-dire envoyer sur un bus en direction de la mémoire une requête disant « envoie-moi le contenu à l'adresse numéro tant » (car chaque bit de la mémoire a une adresse, qui l'identifie uniquement).

En réponse, la mémoire renvoie la donnée requise, ou un code de protestation si les données demandées n'existent pas (comme cela peut arriver si le processeur demande le bit 1 934 675 208 et que la mémoire ne contient que 805 306 368 bits). Dans la pratique, le processeur demande ainsi des suites de quelques dizaines de bits consécutifs (guère plus, car ils ne tiendraient pas tous dans le bus).

Le processeur peut envoyer des données à la mémoire, pour stockage. Le mécanisme est similaire à celui de la lecture.

Le processeur peut additionner, soustraire, multiplier, voire diviser et trifouiller de quelques autres manières des nombres. Ces nombres sont représentés par des suites de 0 et de 1, et ne peuvent pas dépasser certaines valeurs (car le processeur ne peut pas manipuler des suites trop longues d'un seul coup – il reste limité).

Le processeur va chercher ses instructions en mémoire, à la suite : il peut recevoir pour ordre d'aller dorénavant les chercher à une autre adresse. On appelle ça un saut, c'est comme un renvoi dans un formulaire de la Sécurité Sociale (« Si vous n'êtes pas demandeur d'emploi, ne remplissez pas le cadre suivant et allez en 11 »).

Toutes ces fonctions ont des ordres spécifiques, représentés par des suites bien définies de 0 et de 1. Et une suite d'ordre, c'est un programme. Parfois, on appelle le programme un *logiciel* si on veut faire smart, ou *software* si on veut faire smart et british à la fois. »

Article 1 : le règlement doit être respecté

« Mais, Guide Suprême, si les données et les ordres sont des suites de 0 et de 1, comment le processeur distingue-t-il les données des ordres ? »

« Et comment fais-tu la différence entre une averse et un crachin fortement vivifiant ? »

« C'est une question de contexte ; le crachin n'apparaît qu'en Bretagne et Normandie. Dans le Midi, là, il pleut. »

« Ben là c'est pareil, petit scarabée. Si le processeur a demandé une instruction, ce qu'il a obtenu, c'est pour lui une instruction. S'il a demandé des bits censés coder un nombre, il a obtenu un nombre. D'ailleurs, les différents ordres exécutés par le processeur passent leur temps à utiliser des informations pour ce qu'elles ne sont pas ; par exemple, un nombre devient rapidement une adresse d'un bit en mémoire, et vice-versa.

Et si un jour le processeur tombe sur quelque chose qu'il ne comprend pas (par exemple, une donnée qu'il prend pour un ordre, mais ces 0 et 1 ne codent aucun ordre valide), alors le règlement prévoit des instructions alternatives d'urgence, à exécuter.

Ainsi travaille le processeur, toujours obéissant, sans initiative, il est l'exécutant irresponsable qui amplifie toute erreur jusqu'à l'absurde, éventuellement jusqu'au suicide ; le processeur n'a pas de capacité propre d'auto-destruction[4] mais peut se stopper et ne plus fonctionner jusqu'à ce qu'on

[4]Enfin, quelques-uns si, mais on préfère éviter, parce qu'après ça fait des histoires.

coupe et remette le courant. Et paf le rapport de vingt pages. Et c'est la faute du programmeur, qui a conçu une suite d'ordres absurdes.

Une telle erreur de programmeur, ça s'appelle un *bug*. Tout logiciel un tant soit peu utile à un quelconque travail comporte des dizaines de bugs[5], voire même beaucoup plus. Ainsi va le Monde. »

▷ Thomas Pornin

Sécurité : le chiffrement

Le problème de base du chiffrement, c'est la confiance... De même qu'on fait confiance à des gens pour envoyer des jolis programmes, et pas des virus. Ou qu'on fait confiance à son destinataire pour ne pas faire n'importe quoi avec les messages qu'on lui envoie. Ou qu'on fait confiance à l'administrateur système pour ne pas regarder n'importe où. Quand on se rend compte que tout le monde il est pas Beau et tout le monde il est pas Gentil, on se dit qu'on voudrait bien un peu de confidentialité... Ça s'appelle le chiffrement. NDLR : ce qui suit est parfois dit sur un ton ironique ; ne vous laissez pas abuser, son auteur est un spécialiste de cryptographie...

Les mots pour le dire

Chiffrer : rendre inintelligible une donnée à toute personne ne connaissant pas un certain secret, nommé « clé ».

Déchiffrer : utiliser la clé pour rendre à nouveau intelligible une donnée chiffrée.

Décrypter : rendre à nouveau intelligible une donnée chiffrée, sans connaissance préalable de la clé (action normalement impossible si le protocole de chiffrement est correct).

Crypter : ce terme n'existe pas ; il est néanmoins souvent employé en lieu et place de chiffrer. Au bûcher les crypteurs.

Cryptographie : art du chiffrement, de la signature, de l'authentification et de la vérification d'intégrité.

Cryptanalyse : art du décryptage, de la production de faux, de la tromperie et de la corruption silencieuse des données.

Cryptologie : science regroupant la cryptographie, la cryptanalyse et l'analyse scientifique de ces deux domaines.

En quoi ça me concerne ?

— **À quoi ça sert ?** — On a envie de chiffrer ses données dès que l'on transfère des choses privées : informations personnelles, mots de passe, numéro de compte en banque, etc. Sans chiffrement, tout ceci circule comme des cartes postales, lisible par qui veut s'en donner la peine.

Le chiffrement assure un peu de confidentialité aux données sensibles. Il permet aussi de se la jouer superhéros défenseur de la liberté quand on n'a pas de données sensibles à protéger.

Signer/authentifier/vérifier servent à faire des opérations légales, et surtout du commerce, sur un support informatique aléatoire, espionnable et falsifiable à loisir par des hordes de méchants dont on n'imagine pas, d'habitude, le nombre ni l'acharnement (exemple : Internet).

— **Est-ce bien utile ?** — Le chiffrement est utile pour certaines entités. Un exemple-type est un réseau informatique entre une entreprise et ses filiales délocalisées.

Signature/authentification/vérification : oui. Ça permet de lire son courrier sereinement depuis l'université de Pétaouchnok-les-oies.

Mais la cryptographie ne fait pas tout ; par exemple, la meilleure cryptographie n'empêche personne d'espionner un écran à 200 mètres de distance avec un télescope.

— **Est-ce légal ?** — Ça dépend du pays. En France, pour la plupart des systèmes, oui. Le décret du 17 mars 1999 règle les modalités d'usage, de fourniture, d'importation et d'exportation des systèmes cryptographiques ; il en ressort qu'un particulier a accès, légalement, à des outils dont la sécurité est à l'épreuve de l'espèce humaine (dans son intégralité), au regard du niveau technologique atteint actuellement par ladite espèce.

Une loi libéralisant totalement l'usage de la cryptographie a été promise par Lionel Jospin pour la fin 1999 (là c'est raté) puis repromise par Laurent Fabius pour « courant 2001 ».

Dans la plupart des pays, l'usage est libre, l'import/export réglementé. Dans certaines contrées (Chine, Iran, Syrie...), la possession et l'usage sont sévèrement interdites.

— **Est-ce efficace ?** — En termes de sécurité, on sait fabriquer des systèmes inviolables. Il existe aussi plein de systèmes parfaitement nuls. D'autre part, la cryptographie établit des tuyaux sécurisés entre deux points de l'espace-temps, mais ne protège pas ces points. Si l'information existe, en clair, quelque part à un moment donné, elle y est vulnérable. Attention aux gens qui regardent par dessus votre épaule. N'écrivez

[5] Tous sauf un : car Knuth est Grand.

pas vos mots de passe sur un post-it collé sous le clavier.

En termes de performances, c'est raisonnable pour lire son courrier à distance *via* un ssh ou chiffrer un courrier avec PGP[6]. C'est encore trop lourd pour faire une sécurité complète et transparente dans certains cas (chiffrement automatique d'un disque dur, par exemple).

— **Est-ce facile ?** — Pour l'utilisateur de base, oui, ce n'est pas difficile. Sinon, la cryptographie rentre dans le cadre de la sécurité informatique, ce qui est un métier et ne s'improvise pas.

— **En pratique, je fais comment ?** — Les deux prochains numéros détailleront l'utilisation de logiciels sécurisés : en janvier, PGP (pour encrypter le courrier électronique), et en février ssh (pour sécuriser les connexions à distance).

L'indispensable théorie

— **Chiffrement** — On mélange tout, suivant une méthode précise, un peu comme une recette de cuisine. La recette, tout le monde la possède, mais les proportions exactes utilisées, c'est la clé. On peut faire, quand on est très fort, un chiffrement dit « asymétrique », où la clé pour chiffrer et celle pour déchiffrer ne sont pas la même (elles sont liées ensemble, forcément, mais on ne peut pas deviner l'une en fonction de l'autre). Ça permet de rendre une des clés publiques (on la publie dans un annuaire, par exemple) tout en gardant l'autre secrète. Ceci permet de réaliser des opérations intrinsèquement asymétriques :

* Envoi d'un courrier sécurisé à un inconnu : on regarde dans l'annuaire, on chiffre avec sa clé publique. Seul le destinataire peut déchiffrer le message (car lui seul possède la clé de déchiffrement correspondant à la clé de chiffrement qui a été publiée), mais on n'a pas besoin de se mettre préalablement d'accord avec lui sur un secret partagé (accord qui nécessite une transmission un peu plus sûre, on ne crie pas un mot de passe dans la rue).
* Signature : un seul peut signer, mais tout le monde peut vérifier que la signature correspond et au message, et au signataire. Ce dernier signe en utilisant sa clé privée, tout le monde pouvant faire l'opération inverse et constater que ça marche grâce à la clé publique.

Les méthodes de chiffrement asymétriques demandent des constructions mathématiques un peu lourdes ; dans la pratique, ça va nettement moins vite que les méthodes de chiffrement symétriques, où la même clé sert pour le chiffrement et le déchiffrement. Donc des outils tels que PGP (chiffrement et signature de mails) et SSL (établissement d'une connexion authentifiée et chiffrée) utilisent une mixture : de l'asymétrique pour arriver à transmettre de façon sûre un secret commun, utilisation de ce secret commun pour chiffrer efficacement les données.

▷ Thomas Pornin

Lexique

— **Alias** — On appelle *alias* une traduction que vous définissez pour un mot que vous tapez en ligne de commande ; c'est le plus souvent un raccourci. Par exemple, si vous tapez `rm`, la machine comprendra `rm -i`. Les alias qui sont définis de façon permanente le sont dans le fichier `.zshrc` de votre compte (config conscrits). La commande `alias` permet d'avoir la liste des alias actuellement définis.

— **URL** — *Uniform Resource Locator.* C'est ce que les médias appellent une « adresse Web », C'est un bout de texte, possible à taper au clavier, qui indique comment accéder à un document ou une autre donnée. Les plus courantes sont en « `http ://` », ce qui indique le protocole (pour ce mot, voir le numéro 9 du *Hublot*) à utiliser, et précisent un ordinateur à contacter et le fichier à lui demander.

Le *Hublot* est le journal des tuteurs informatiques de l'ENS. Il paraît chaque mois à 350 exemplaires. Il est également disponible sur le Web :
`http://www.eleves.ens.fr:8080/tuteurs/hublot/`
Rédaction : Émilia Robin.
Ont collaboré à ce numéro : Marc Espie, Nicolas George, David Madore, Thomas Pornin, Jérôme Plût, Gilles Radenne, Yann Tholoniat.
Merci à Olivier Verzelen pour ses relectures.

[6] Patience, ces termes mystérieux seront expliqués dans les prochains numéros.