

# Introduction à FreeBSD

Atelier ccTLD Dakar

Décembre 2005  
Dakar, Senegal

Phil Regnault  
(Hervey Allen, version anglaise)



# Contenu

- Pourquoi FreeBSD.
- Le monde de FreeBSD.
- Installation de FreeBSD 5.4.
- Ligne de commande vs. Interface graphique
- Fichiers de configuration.
- Partitionnement du disque sous FreeBSD.
- Hiérarchie disque sous FreeBSD.
- Démarrage de FreeBSD (man boot)
- Utilitaires et programmes.
- Créer et supprimer des comptes utilisateur.
- L'éditeur vi.

# Contenu (suite)

- Configuration d'une interface réseau.
- Arrêt et redémarrage du serveur – runlevels.
- Les services, et ce qui est actif.
- Comment installer des logiciels:
  - Packages (paquets)
  - ports
  - source
  - cvs
- Permissions fichiers. Commandes “chmod”, “chown”.
- Résumé
- Ressources supplémentaires.

# Pourquoi FreeBSD?

Une question qui j'en suis sur plusieurs d'entre vous se posent...



==> Voir le commentaire suivant:

<http://ws.edu.isoc.org/workshops/2005/ccTLD-Nairobi/day1/freebsd/whyfreebsd.html>

# Linux != UNIX



# Le monde de FreeBSD

Démarrer ici: <http://www.freebsd.org/>

- RELEASE (6.0, 5.4 and 4.11 legacy)
- STABLE ('beta' code)
- CURRENT ('alpha' code)
- Ports
- Packages (paquets)
- Projet documentation
  - FreeBSD Handbook (manuel FreeBSD)



# Installation de FreeBSD (5.4)

- Comment installer ? (FreeBSD Handbook section 2.2.6)
  - Un CDROM ou DVD
  - Disquette 3" 1/2 (y compris préconfiguration)
  - Site FTP/HTTP, en passant par un proxy HTTP et/ou firewall si nécessaire
  - Serveur NFS
  - Partition DOS sur un disque local
  - Bande magnétique SCSI ou autre
  - Connexion parallèle ou série dédiée

# Ligne de commande vs Interface graphique

- L'administration d'un serveur FreeBSD peut se faire entièrement via la ligne de commande.
- Une interface graphique n'est pas nécessaire (web, mail, impression, serveur de fichiers, SGBD, etc.) quand on utilise FreeBSD (ou Linux/UNIX).
- On peut lancer plusieurs interpréteurs de commandes (shells) en même temps.
- Pour utiliser une interface graphique (GUI) telle que KDE ou Gnome, il faut installer le système de fenêtrage X (X window). Ceci sera couvert plus tard.

# Fichiers de configuration

- Dans le monde Windows, la plupart de la configuration se fait via la base de registre (Registry), qui est une base de données binaire.
- Sous FreeBSD quasiment toute la configuration se fait par le biais de fichiers de configuration.
- Les outils graphiques de configuration se contentent d'écrire un fichier de configuration texte.
- Pour configurer les services, vous aurez souvent besoin d'être l'utilisateur privilégié "root", et devrez rédiger directement des fichiers textes.

# Hiérarchie disque FreeBSD

Si vous voulez comprendre comment FreeBSD organise et voit les disques, alors lisez la section 3.5 du manuel FreeBSD (handbook) pour une très bonne description.

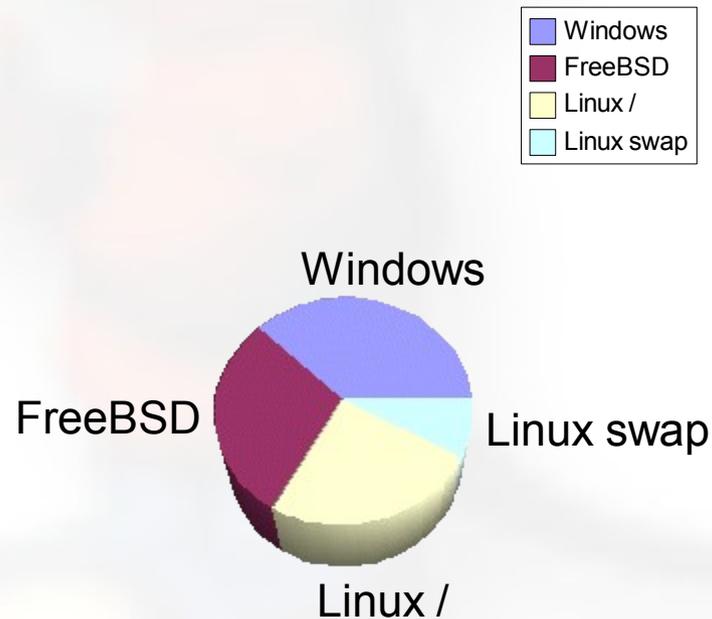
Si vous avez l'habitude de partitionner les disques depuis le monde Windows, vous verrez qu' UNIX (FreeBSD, Linux, Solaris, etc.) *partitionne* les données de manière simple et efficace.

Sous FreeBSD, une “slice” (tranche) est ce que Windows appelle une “partition”.

# Exemple de tranche (slice)

- Partition BIOS = tranche de disque (4 max)

Tranches = part de disque



# Partitionnement FreeBSD

<u>Partition</u>	<u>Utilisation</u>
a	Root partition (/), la racine
b	swap partition
c	le disque entier, non-utilisée.
d	utilisation libre
e/f	/tmp, /usr, etc...

Voir les informations de partitions avec “df -h” et “swapinfo” - voir aussi disklabel.

# Les tranches disques vue par FreeBSD

Exemple de sortie de la commande

```
“fdisk -s”
```

```
/dev/ad0: 77520 cyl 16 hd 63 sec  
Part          Start          Size Type  Flags  
  1:           63      8385867 0x0b 0x80  
  2:      8385930      8385930 0xa5 0x00  
  3:     16771860       208845 0x83 0x00  
  4:     16980705     61159455 0x0f 0x00
```

Disque de 40 Go, avec 3 systèmes

d'exploitations (SE) installés sur 4 tranches. Les SE sont: Windows 2000 (1), FreeBSD (2), Linux (3), et une partition de swap DOS pour Windows 2K

# Partitions FreeBSD dans une tranche (slice)

Vous pouvez obtenir des informations plus détaillées en tapant simplement “fdisk”.

Pour voir les partitions d'une tranche FreeBSD:  
“disklabel /dev/DEV”: (DEV = tranche)

```
# /dev/ad1s1:
8 partitions:
#          size      offset      fstype    [fsize bsize bps/cpg]
a:   524288         0      4.2BSD    2048 16384 32776
b:  2045568   524288        swap
c: 122865057         0      unused         0     0      # "raw" part, don't edit
d:   524288  2569856      4.2BSD    2048 16384 32776
e:   524288  3094144      4.2BSD    2048 16384 32776
f: 119246625  3618432      4.2BSD    2048 16384 28552
```

# Partitions FreeBSD dans une tranche (slice) - suite

Pour voir les informations des partitions montées de manière plus lisible: “df -h”.

Attention: tout n'est pas une vraie partition.

Filesystem	Size	Used	Avail	Capacity	Mounted on
/dev/ad1s1a	248M	35M	193M	15%	/
devfs	1.0K	1.0K	0B	100%	/dev
/dev/ad1s1e	248M	526K	227M	0%	/tmp
/dev/ad1s1f	55G	2.7G	48G	5%	/usr
/dev/ad1s1d	248M	42M	186M	18%	/var
/dev/ad1s2	55G	15G	38G	28%	/data
/dev/da0s1	500M	226M	274M	45%	/mnt/flash

Utiliser “swapinfo” pour la partition swap:

Device	1K-blocks	Used	Avail	Capacity
/dev/ad1s1b	1022784	124	1022660	0%

# Organisation hiérarchique du disque sous FreeBSD

Voir la page de man hier: 'man hier'

Pourquoi diviser la tranche FreeBSD en sous-partitions ? Pour isoler les parties importantes du système de fichiers les unes des autres. Souvent ces systèmes de fichiers sont représentés par des répertoires particuliers.

Pourquoi ne pas tout mettre dans une seule partition ? C'est à dire: tout sous la racine ( root - / )

# Organisation hiérarchique du disque sous FreeBSD

## Avantages du système de fichier unique:

- Pas de problème de redimensionnement si une partition est trop petite.
- Conceptuellement plus simple pour certaines applications et personnes (station de travail).

## Avantage des systèmes de fichiers multiples:

- Problème sur un système, les autres continuent:
  - Utilisateur qui remplit une partition.
  - Coupure de courant qui abîme une seule partition
  - Montage des partitions système en lecture seule
- FreeBSD peut optimiser le placement des fichiers en fonction de l'utilisation
- Séparation logique des fonctions, meilleure sécurité.

# Quelques répertoires FreeBSD

- Structure des partitions / répertoires
  - / (“root”, la racine)
  - /usr
  - /var
  - swap
- Deux répertoires importants:
  - /var/tmp
  - /usr/home



# “/” Root – la racine

La partition racine est là où résident les fichiers systèmes vitaux, y compris ceux nécessaire au démarrage du système en mode “single-user” (mono utilisateur).

Le concept est que cette partie du système change très peu et ne grossit pas.

Si suffisamment d'espace est alloué à /usr et /var, alors “/” peut être réduit: aux alentours de 256Mo suffisent.

Un répertoire qui peut grossir est /tmp.

# /usr

Utilisé pour les logiciels tels que les outils utilisateur, compilateurs, X Window, etc...

Les paquets – ainsi qu'un grand nombre de logiciels libres – sont installés par défaut sous /usr/local.

Si le besoin survient\* de redimensionner cette partition pour pouvoir ajouter des logiciels, alors le fait d'en avoir fait une partition séparée rend ceci possible.

FreeBSD place les répertoire utilisateur sous /usr/home.

(\*) Il n'est pas toujours nécessaire d'avoir une partition /usr dédiée.

# /var

Là où résident les fichiers et répertoires qui sont en changement continu. Par exemple, les journaux (logs), répertoires mail, queue d'impression, fichiers temporaires.

Sur un serveur il est recommandé de placer /var sur une partition séparée afin d'éviter de remplir les autres systèmes de fichier par accident.

# swap

La mémoire virtuelle utilise le swap (pagination). Le swap à une partition dédiée.

On peut fonctionner sans swap, mais c'est dangereux si on est à court de mémoire.

En fonction du type de service qui va tourner sur le serveur, on recommande entre 1 fois et 2 fois la taille de mémoire vive en swap.

Par exemple: 512 Mo de RAM -> 1 Go de swap.

# Démarrage de FreeBSD

## Processus d'initialisation:

- Le boot loader est chargé, et à son tour il charge le noyau (/boot/kernel/kernel)
- Après le démarrage, le noyau passe le contrôle à /sbin/init.
- Si “/” est “propre” alors init lance le script de démarrage /etc/rc, qui à son tour va lire les variables de configuration dans:
  - /etc/defaults/rc.conf
  - /etc/rc.conf (redéfinit /etc/defaults/rc.conf)
  - /etc/rc.conf.local (redéfinit /etc/rc.conf)

# Démarrage de FreeBSD (suite)

## Processus d'initialisation (suite)

- “Montage” des systèmes de fichier listés dans /etc/fstab
- Ensuite, les services suivants démarrent:
  - Services réseau
  - Démons système
  - Démons appartenant aux paquets additionnels (/usr/local/etc/rc.d)
  - Démarrage des gestionnaires de session / terminaux (/etc/ttys: getty, login)

# Démarrage de FreeBSD (suite)

## **Arrêt du système:**

- Sur appel de `shutdown` `init` lance le script `/etc/rc.shutdown`.

# Commandes - Outils - Shell – Chemin

Qu'est-ce qu'une “commande” et un “programme”?

Pourquoi ne peut-on pas toujours exécuter toutes les commandes et programmes sur un système ?

Comment peut-on “corriger” ceci ?

Quelle est la configuration par défaut de l'environnement d'un utilisateur ?

- /usr/share/skel
- /etc/profile
- /home/user/.bashrc
- /home/user/.bash\_profile
- set, printenv, export

# Commandes de base

- cp, cd\*, ls, mkdir, mv, rm y man
  - (\*commandes intégrées dans le shell).
- Où les commandes sont-elles placées ?
- /bin, /usr/bin, /usr/local/bin, /sbin, /usr/sbin
  - Différence entre “sbin”, “bin” et “/usr”
- Si vous connaissez DOS:
  - cp = copy
  - cd/chdir = cd/chdir
  - ls = dir
  - mkdir = mkdir
  - mv = move
  - rm = del[ete] et/ou erase

# Création, suppression et mise à jour des comptes utilisateur

(FreeBSD Handbook section 8.6)

## Création des utilisateurs et maintenance

- passwd, pw, vipw

## Quelques fichiers associés

- /etc/passwd, /etc/group, /etc/master.passwd, /usr/local/etc/sudoers (man visudo)
- /usr/share/skel
- /var/mail/\*

# /etc/passwd

Fichier contenant la liste des utilisateurs du système. Le format est le suivant:

```
hervey:x:500:500:Hervey  
Allen:/home/hervey:/usr/local/bin/bash
```

i.e.:

```
user:pw:UID:GID:name:directory:shell
```

Dans /etc/passwd le mot de passe (“pw”) est caché par une astérisque (“\*”). Si l'utilisateur est un utilisateur virtuel (un service), alors son shell sera “/sbin/nologin”.

# /etc/master.passwd

Ce fichier sert à placer les mot de passes chiffrés. Seul l'utilisateur root peut lire ce fichier. Les informations sont stockées dans /etc/pwd.db et /etc/spwd.db sous forme de DB Berkeley pour une consultation rapide.

/etc/master.passwd a le format suivant:

```
hervey:$1$qvAgYWGD$nLf/LpT1r0XXXXXXjMC/:1001:1001::0:0:Hervey  
Allen:/home/hervey:/usr/local/bin/bash
```

**i.e.:**

- Nom de login
- Mot de passe chiffré. Si il démarre par "\$1\$". c'est un chiffrement md5
- ID de l'utilisateur (uid)
- Groupe de l'utilisateur (gid)
- Classe de login (voir /etc/login.conf)

# /etc/master.passwd - suite

```
hervey:$1$qvAgYWGD$nLf/LpT1r0XXXXXXjMC/:1001:1001::0:0:Hervey  
Allen:/home/hervey:/usr/local/bin/bash
```

- Délai de renouvellement du mot de passe (0 = jamais)
- Quand le compte expire-t-il (0 = jamais)
- Informations générales sur l'utilisateur
- Répertoire de l'utilisateur (home dir).
- Interpréteur de commandes (shell)

# L'éditeur VI

- Pourquoi vi? Pourquoi pas emacs, xemacs, joe, pico, ee, etc.?
- vi existe sur pratiquement tous les types d'UNIX et Linux.
- Si vous vous retrouvez sur une nouvelle machine, vi sera certainement disponible.
- Il se peut que vous choisissiez un éditeur plus avancé, mais nous nous en entraineront avec vi après avoir installé FreeBSD.

# Configuration des interfaces réseau

Si au cours du démarrage du noyau, une carte réseau reconnue est présente, celle-ci apparaît

On peut charger des modules après le boot si l'interface n'est pas reconnue

Après démarrage, utiliser `ifconfig` pour voir si l'interface existe. Chercher l'adresse MAC

La configuration de base se fait avec `ifconfig` ou bien "`dhclient dev`"

Si la carte fonctionne, mettre à jour les paramètres dans `/etc/rc.conf`.

# Configuration des interfaces réseau - suite

## Exemples de configuration dans `/etc/rc.conf`

```
hostname="localhost.localdomain"  
ifconfig_fxp0="DHCP"
```

Configure `hostname` et indique que la carte réseau `fxp0` utilisera le protocole DHCP pour obtenir la configuration réseau. FreeBSD utilise un nom différent pour chaque modèle de carte. `fxp0` indique la première carte Intel EtherExpress.

Voir aussi `ifconfig name` pour changer le nom d'une interface.

# Arrêt et redémarrage d'un serveur FreeBSD

Comment arrête-t-on un serveur FreeBSD ?

- shutdown 1 message
- halt
- init 0

Et pour redémarrer ?

- reboot
- shutdown -r now
- init 6

# Runlevels

FreeBSD a un concept de runlevels:

Run-level	Signal	Action
0	SIGUSR2	Arrêt et coupure alimentation
1	SIGTERM	Mode mono utilisateur
6	SIGINT	Redémarrer la machine.

En pratique, soit 6, soit 1 – soit la machine est en multi-utilisateur, soit en mono-utilisateur.

Voir man init – FreeBSD a un concept de niveaux de sécurité (securelevels)

# Runlevels - suite

Qu'est ce qui est exécuté en mode multi utilisateur:

- Valeurs par défaut dans `/etc/defaults/rc.conf` (cf. scripts dans `/etc/rc.d`).
- Redéfini par l'administrateur `/etc/rc.conf`.
- Systèmes de fichier listés `/etc/fstab` sont montés
- Les services provenant de paquets/ports ont un scripte de démarrage placé dans `/usr/local/etc/rc.d`.

Les paramètres de démarrage pour les paquets et ports vont aussi dans `/etc/rc.conf`

# Qu'est ce qui tourne sur un système ?

- Voir tous les processus actifs:
  - `ps -aux | more`
- Voir un processus particulier
  - `ps -aux | grep "nom"`
- Les services réseau actifs
  - `sockstat`

Noter l'utilisation de “|” pour connecter la sortie d'une commande vers l'entrée d'une autre (tuyauterie).

# Méthodes d'installation des logiciels

Il y a 3 façons d'installer des logiciels sur une machine FreeBSD

- 1.) paquets FreeBSD (packages) et `pkg_*`
- 2.) la ports-collection (`/usr/ports`) et `portupgrade`
- 3.) Installation à partir des sources

Généralement: 1. paquetage, 2. port, 3. source

Il y a des avantages et inconvénients à chaque.

# Les 3 méthodes

- Distribution source
  - Installation à la main
  - `tar xvfz; ./configure; make; make install`
- Installation port via les sources
  - Installation semi-automatique, création de paquetages:
  - `cd /usr/ports/categorie/outil; make install`
- Installation d'un paquetage binaire:
  - `pkg_add paquetage.tgz`
  - `pkg_add ftp://serveur/.../paquetage.tgz`
  - `pkg_add -r paquetage`

# Les outils “pkg\_\*”

Les commandes `pkg_add` et `pkg_delete` permettent d'ajouter et de supprimer facilement des paquetages logiciels.

La commande `pkg_info` permet d'obtenir une liste des paquets installés, ainsi que des détails sur un paquetage particulier.

# Installation de paquetage avec pkg\_add

- Installation des paquetages possible via le CD d'installation ou par FTP/HTTP, voir sur un serveur local monté en NFS.

- Pour installer depuis un CD-rom:

```
pkg_add /cdrom/dir/nom_du_paquetage
```

- Pour installer depuis un serveur FTP ou HTTP

```
pkg_add ftp://address/dir/package_name
```

```
pkg_add http://address/dir/package_name
```

- Pour installer automatiquement:pkg\_add

```
pkg_add -r package_name
```

# Utilisation de pkg\_info

Savoir si quelque chose est déjà installé:

```
pkg_info      (liste des paquetages installés)
```

```
pkg_info | grep moz      (find all packages  
                          containing "moz")
```

Get more information about an already installed package:

```
pkg_info name\*
```

```
pkg_info -I name\*
```

For example "pkg -I bash\\*" returns:

```
bash-2.05b.007_2      The GNU Bourne Again Shell
```

Savoir à quel paquetage un fichier appartient:

```
pkg_info -W /chemin/vers/le/fichier
```

# Utilisation de `pkg_delete`

Si vous désirez supprimer un paquetage:

```
pkg_delete nom_du_paquetage
```

Si vous voulez supprimer un paquetage ET toutes ses dépendences:

```
pkg_delete -r nom_du_paquetage
```

Mais *soyez prudents* en faisant ceci. Vous pouvez vérifier l'impact du commande avec de l'exécuter en faisant:

```
pkg_delete -n nom_du_paquetage
```

# Installation depuis les ports

Il faut d'abord avoir installé la collection des ports dans /usr/ports par l'une des méthodes suivantes:

- sysinstall: Configure -> Distributions -> Ports.
- cvsup (voir [http://www.freebsd.org/doc/en\\_US.ISO8859-1/books/handbook/cvsup.html](http://www.freebsd.org/doc/en_US.ISO8859-1/books/handbook/cvsup.html))
- portsnap (standard dans FreeBSD 6.0 – voir [http://www.freebsd.org/doc/en\\_US.ISO8859-1/books/handbook/portsnap.html](http://www.freebsd.org/doc/en_US.ISO8859-1/books/handbook/portsnap.html))

Une fois installé, vous pouvez parcourir la collection dans /usr/ports (plus de 12.000 ports)

Permet d'installer un logiciel depuis un minimum d'informations et une seule commande (make)

Voir la section 4.5 du FreeBSD Handbook.

# Installation depuis les ports - suite

Voir si un logiciel existe en tant que port:

```
cd /usr/ports
make search name=paquetage
make search key=mot_clé
```

Exemple pour lsof (LiSt Open Files):

```
cd /usr/ports
make search name=lsof
```

Et la sortie de la commande devrait être:

```
Port:      lsof-4.69.1
Path:      /usr/ports/sysutils/lsof
Info:      Lists information about open files (similar to
           fstat(1))
Maint:     obrien@FreeBSD.org
Index:     sysutils
B-deps:
R-deps:
```

# Installation depuis les ports - suite

Cf. la page précédente, lsof se trouve dans  
`/usr/ports/sysutils/lsof`

Si vous avez une connexion réseau...

Il vous suffit de taper la commande “`make  
install`”

Mais vous pouvez le faire en plusieurs étapes:

- `make`
- `make install`

Pour forcer un serveur FTP particulier pour  
récupérer les distributions:

- `export MASTER_SITE_OVERRIDE="ftp://local.site/distfiles/"`

# Installation depuis les ports - suite

On peut aussi compiler depuis la distribution sur le CD-rom – il suffit de monter le cdrom (“mount /cdrom”) et ensuite:

- `cd /usr/ports/sysutils/lsof`
- `make`
- `make install`

Et le système des ports ira automatiquement chercher la distribution sur le CD-rom, et sur Internet le cas échéant.

# portupgrade

- A mi-chemin entre les paquetages et les ports
- `/usr/ports/sysutils/portupgrade` (s'installe comme tout autre port/paquetage)
- Travaille “au-dessus” de `make` et `pkg_*`
- Meilleure gestion des dépendances lors des installations, mises à jour
- Réparation de `/var/db/pkg/*` - `pkgdb -F`

# portupgrade - suite

- Exemple:
  - `portupgrade bash ->` mise à jour via les ports
  - `portupgrade -P base ->` mise à jour via un paquetage binaire, ou via un port si un paquetage n'existe pas (-PP = forcer binaire, échec sinon)
  - `portinstall apache ->` installation d'apache via les ports
  - `portinstall -P apache ->` installation d'apache via un paquetage binaire, ou via un port si un paquetage n'existe pas...

# Paquetages ou ports ?

- Avantage du paquetage binaire
  - Rapide d'installation
- Inconvénients: moyen de configurer l'installation limitée, pas de possibilité de changer les paramètres de compilation
- Avantage du port
  - Possibilité de “customisation” avant la compilation (make extract; **vi ...**; make configure; make
- Inconvénients: plus de temps à compiler

# Installation à partir des sources

Si vous avez absolument besoin d'installer un logiciel à partir des sources (pas recommandé)

Dans ce cas, il faut rapatrier la distribution source à la main

Typiquement, les sources sont “livrées” en tant que `.tgz` ou `.tar.gz`

Exemple d'installation à partir des sources...

# Installation à partir des sources

- Rapatriement d'un fichier [fn.tar.gz](#) dans /usr/src.
- `tar -xvzf /usr/src/fn.tar.gz`
- `cd /usr/src/fn-version`
- `./configure`
- `make`
- `make install`

Ceci dans le meilleur des cas – mais maintenant il devient difficile de tenir compte de ce qui est installé, et de comment le désinstaller...

# CVSUP et Portsnap

Question: comment tenir ses ports à jour ?

CVSup est une méthode.

D'abord installer cvsup:

```
pkg_add -r cvsup-without-gui
```

Ensuite lancer cvsup avec un fichier de configuration fourni, en choisissant un miroir cvsup:

```
cvsup -h cvsup.freebsd.org \  
/usr/share/examples/cvsup/ports-supfile
```

# CVSUP et Portsnap - suite

On démontrera l'utilisation de CVSup et de Portsnap.

(Portsnap est standard dans FreeBSD 6.0, et disponible dans `/usr/ports/sysutils/portsnap` sinon)

# Permissions fichiers et droits

- 5 catégories et 3 types de permissions.
- Les permissions par défaut d'un utilisateur sont contrôlées par la commande `umask`
- 2 catégories de permissions liées au droits de l'utilisateur lors de l'*exécution* d'une commande (`setuid`, `setgid`).
- Les droits d'accès en lecture/écriture/exécution sont: “r” (lecture), “w” (écriture), and “x” (execute).
- Vous pouvez affecter des permissions à l'utilisateur (u), au groupe(g), et au reste (o).

# Permissions fichiers et droits - suite

- Un fichier, ou un répertoire, appartient à un utilisateur. Vous pouvez, en tant que root, assigner le fichier à un autre utilisateur à l'aide de la commande `chown` (“CHange OWNeR”)
- Vous pouvez changer les droits que l'utilisateur et le groupe ont sur un fichier, ou un répertoire, avec la commande `chmod` (“CHange MODe”)
- Nous utiliserons les commandes `chmod` et `chown`

# Résumé

- FreeBSD vise principalement le serveur, pas la stations de travail (même si cela fonctionne parfaitement)
- Historiquement, grande solidité, sécurité et fiabilité.
- Adapté à de grosses charges de travail.
- La collection des ports donne accès à un grand nombre de logiciels, y compris la compatibilité binaire avec Linux.
- Pas besoin d'interface graphique – configuration à l'aide de fichiers textes simples.
- Plusieurs façons d'installer des logiciels

# Ressources supplémentaires

Vous trouverez cette présentation ici

<http://ws.edu.isoc.org/workshops/2005/ccTLD-Dakar/jour1/freebsd/>

- <http://www.freebsd.org/>
- <http://www.google.com/>
- <http://www.freebsd.org/support.html>
- Livres O'Reilly (<http://www.oreilly.com/>)
- <http://www.freshports.org/>
- <http://www.freebsdjournal.org/>