

# *Vue d'ensemble sur FreeBSD*

## *Comparaison avec Linux*

Atelier ccTLD

7-10 décembre 2005  
Dakar, Sénégal

Présenté par Hervey Allen  
Traduit par Alain Patrick AINA



# Quelques points pratiques

- Au moment de l'installation, utilisez le mot de passe "root" donné en classe.
- Pendant toute la semaine, ne changez pas le mot de passe "root".
- *Posez des questions! Beaucoup de questions ! Nous le voulons vraiment.*
- *Si vous avez des zones d'ombre, demandez de l'aide! c'est de cette manière que vous apprenez*
- *Questions?*

# Sommaire

- Le monde du FreeBSD
- Installation de FreeBSD 5.4
- Partitionnement du disque de FreeBSD
- Structure des répertoires de FreeBSD (man hier)
- Comment FreeBSD s'initialise ? (man boot)
- Configuration d'interface réseau
- Arrêt et redémarrage du serveur – runlevels

# Suite du sommaire

- Comment installer un logiciel:
  - paquetages
  - “ports”
  - source
  - cvs
- Résumé
- Plus de ressources

# Linux != UNIX



# Le monde du Free BSD

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

- RELEASE (5.4 et 4.11 legacy)
- STABLE ('beta' code)
- CURRENT ('alpha' code)
- Ports
- Paquetages
- Projet de documentation
  - Manuel du FreeBSD



# Installation de FreeBSD (5.4)

- Comment pouvez –vous l’installer? (section 2.2.6 du manuel du freeBSD)
  - un CDROM ou DVD
  - Disquettes (incluant une installation préconfigurée)
  - Un site FTP à travers un pare-feu ou en utilisant un proxy HTTP, si nécessaire
  - Un serveur NFS
  - Une partition DOS sur la même machine
  - Une cartouche SCSI ou QIC
  - Une connexion série ou parallèle dédiée

# Organisation du disque de FreeBSD

Si vous voulez comprendre comment FreeBSD organise et voit les disques, il faut lire donc la section 3.5 du manuel de FreeBSD qui offre une description excellente et succincte

Si vous venez au partitionnement du disque avec les perspectives de Windows, vous verrez que vous UNIX (FreeBSD, Linux, Solaris, etc.) *partitionne les données facilement et efficacement*

*Dans FreeBSD un “slice” est ce que vous pouvez considérer comme une “partition” sous*

# Entire IDE Hard Drive (20G) / ad0

## Windows

### Slice 1 (ad0s1)

"C: drive" = 5G

ad0s1a

"D: drive" = 5G

ad0s1b

## FreeBSD

### Slice 2 (ad0s2)

/ = 512M

swap = 512 M

/tmp = 512 M

/var = 2 G

/usr = 6.5G

## Partitions

ad0s2a

ad0s2b

ad0s2d

ad0s2e

ad0s2f

# Schémas de partitions FreeBSD

| <u>Partition</u> | <u>Utilisation</u>                        |
|------------------|---|
| a                | partition racine(/)                       |
| b                | partition “swap”                          |
| c                | non utilisé pour les systèmes de fichiers |
| d                | Supposé non utilisé souvent.              |
| e/f              | /tmp, /usr, etc...                        |

Visualiser les informations sur les partitions avec “df -h”  
et “swapinfo”

# “Slices” de disque FreeBSD

Exemple de résultats pour visualiser les “slices” de disques à partir de “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
```

C’est un disque de 40GO avec 3 systèmes d’exploitation déployés à travers 4 “slices”. Les systèmes d’exploitation sont :

Windows 2000 (1), FreeBSD (2), Linux (3) et la 4ème partition est un “slice” de swap DOS pour Windows 2000

# Partitions FreeBSD dans un “slice”

Vous pouvez avoir plus d’informations sur les “slices” de votre disque en tapant simplement “fdisk”

Pour voir les partitions dans un “slice” de FreeBSD, utiliser “disklabel /dev/DEV”:

```
# /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 un “slice” (suite).

Pour visualiser les informations sur les partitions d’un “slice” sous un format plus compréhensible, utiliser “df -h”. Ceci peut, cependant, être déroutant. Par exemple:

| 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 visualiser la partition swap:

| Device      | 1K-blocks | Used | Avail   | Capacity |
|-------------|-----------|------|---------|----------|
| /dev/ad1s1h | 1022704   | 124  | 1022660 | 0%       |

# Structure des répertoires de FreeBSD

Répéter après moi:

“La commande 'man hier' est votre ami.”

Donc, pourquoi la partition de votre disque FreeBSD se répartit en des “slices”? Grandement pour séparer des systèmes de fichiers importants des uns des autres. Ces systèmes de fichiers sont généralement représentés par des répertoires spécifiques.

Pourquoi ne pas avoir tout en une seule place? C’est à dire, tout sous la racine (/).

- Note: FreeBSD peut optimiser la répartition des fichiers sur la base de l’utilisation des systèmes de fichiers.

# Quelque annuaires de FreeBSD

- Structure des partitions/répertoires:

- / (“racine”)

- /usr

- /var

- swap



- Deux importants répertoires:

- /var/tmp

- /usr/home

# “/” Racine

La partition racine là où résident les fichiers systèmes critiques y compris les programmes nécessaires pour initialiser le système en mode “single user”.

L'idée est que cette partie du système ne se développe ou ne change pas, mais plutôt reste isolée du reste du système d'exploitation.

Si vous donnez assez de place à /usr et /var, alors “/” peut être tout à fait petit (environ 512MO doit être mis de côté maintenant).

Le seul répertoire qui peut s'élargir est /tmp, particulièrement si vous utilisez des binaires Linux qui utilisent /tmp.

# /usr

Est utilisé par les logiciels système comme des outils des utilisateurs, compilateurs, XWindows, et répertoires locaux sous la hiérarchie /usr/local.

Si quelqu'un a besoin d'étendre\* cette partition pour des logiciels supplémentaires, alors l'avoir séparée rend ceci possible.

FreeBSD fait correspondre des répertoires utilisateur à /usr/home.

\*Nous discuterons de ceci. Nous n'installons pas toujours FreeBSD avec une partition /usr séparée.

# /var

C'est là où les fichiers et les répertoires qui changent régulièrement sont stockés. Par exemple, les journaux des serveurs web, répertoires des courriers électroniques, queues d'impression, fichiers temporaires, etc.

Sur un serveur, c'est une bonne idée d'avoir le /var sur une partition séparée pour éviter de le voir remplir les autres systèmes de fichiers par accident.

# “swap”

“Swap” est l’endroit où résident les mémoires virtuelles. “Swap” est son propre système de fichier.

Vous pouvez fonctionner sans le swap, et votre PC peut être rapide, mais ceci est dangereux si vous manquez de mémoire.

Il y a plusieurs points de vue sur la taille optimale du swap. Ceci peut dépendre des types de services qui tournent (les bases de données ont besoin de plus de swap). Par routine la taille de swap doit être quelque part entre votre RAM et deux fois la RAM de votre ordinateur.

/u

## Méthodologie de système de fichier optionnel à considérer:

Rend les choses plus faciles si vous créez “/u” à l’installation.

Les données sont stockées dans /u, et vous pouvez faire un lien symbolique de /home vers /u/home.

Vous pouvez réinstaller FreeBSD à partir de zero sans toucher à /u.

Garder des sauvegardes de /etc/ and /usr/local/etc/ aussi bien que d’autres répertoires dont vous avez besoin. Considérer à garder les “tar” de ceux-ci quelque part sur /u.

# Comment FreeBSD s'initialise ?

## **Le processus d'initialisation:**

- **Après l'initialisation du noyau, lequel se trouve dans “/” (Sous Linux, il se trouve dans /boot), il passe le contrôle au programme /sbin/init.**
- **Si les systèmes de fichiers sont bons, alors, “init” commence par lire les configurations des ressources du système. Ces fichiers sont lus dans cet ordre:**
  - **/etc/defaults/rc.conf**
  - **/etc/rc.conf (passe outre le précédent)**
  - **/etc/rc.conf.local (passe outre le précédent)**
- **“Monte” les systèmes de fichiers qui sont dans /etc/fstab**

# Comment FreeBSD s'initialise ? (suite).

## **Le processus d'initialisation (suite):**

- Dès que les systèmes de fichier sont montés, alors les éléments suivants démarrent :
  - **Services réseaux**
  - **Les démons Système**
  - **Démons de paquetages localement installés (scripts dans /usr/local/etc/rc.d)**

## **Le processus d'initialisation et d'arrêt:**

- Quand shutdown est lancé alors "init" utilise les scripts /etc/rc.shutdown.

# Fichiers de mots de passe FreeBSD

## Il y a 4 fichiers:

### – **/etc/passwd**

fichier des mots de passe au format ASCII, avec les mots de passe supprimés

### – **/etc/master.passwd**

fichier des mots de passe au format ASCII , avec les mots de passe intacts

### – **/etc/pwd.db**

base de données des mots de passe au format db(3), avec les mots de passe supprimés

### – **/etc/spwd.db**

base de données des mots de passe au format db(3), avec les mots de passe intacts

# Fichiers de mots de passe FreeBSD (suite).

- `/etc/master.passwd` joue le même rôle que le fichier *shadow* sous *Linux*
- *Les comptes sans shells sont spécifiés en utilisant le shell de `/nonexistent` vs. `/sbin/nologin`*
- *`/etc/pwd.db` and `/etc/spwd.db` sont des bases de données. Ceci signifie que la recherche de nom d'utilisateur n'est pas linéaire comme sous *Linux*.*

# Configuration des interfaces réseaux

Pendant l'initialisation, si une carte réseau est reconnue, alors le code approprié est chargé pour supporter la carte (un module).

Après l'initialisation, en utilisant “`ifconfig`” vous pouvez voir si la carte réseau existe.  
Rechercher son adresse MAC.

La configuration initiale de la carte réseau peut être faite avec `ifconfig`, or try “`dhclient dev`”

Si la carte réseau fonctionne, éditer `/etc/rc.conf` et mettre des entrées spécifiques pour l'équipement pour le démarrage

# Configuration des interfaces réseaux (suite).

Exemples de ligne dans `/etc/rc.conf` pour  
équipement réseau:

```
hostname="localhost.localdomain"
```

```
ifconfig_wi0="DHCP"
```

Définit le nom de la machine et indique que la  
carte réseau "wi0" va utiliser DHCP pour  
obtenir des informations de réseau.

FreeBSD utilise des noms spécifiques pour  
chaque carte réseau. "wi0" indique la  
première carte "sans fil".

# Configuration des Interfaces réseaux (suite).

- FreeBSD 5.4 vous permet de renommer les interfaces réseaux comme vous voulez.
- Les utilisateurs de Linux qui préfèrent “eth0” à “wi0” pourraient configurer ceci dans /etc/rc.conf avec:
  - `ifconfig_wi0 = "DHCP name eth0"`
- Cependant, certains programmes, espèrent des noms spécifiques, pour des interfaces réseaux

# Arrêt et redémarrage d'un Serveur

Comment arrêtez-vous une machine sous FreeBSD?

- shutdown 1 message
- halt
- init 0

et, pour redémarrer?

- reboot
- shutdown -r now
- init 6

# Niveau de fonctionnement

FreeBSD a le concept de niveau de fonctionnement:

| Niveau | Signal  | Action                                |
|--------|---------|---------------------------------------|
| 0      | SIGUSR2 | Arrêt de l'OS et de l'alimentation    |
| 1      | SIGTERM | Aller dans le mode "mono-utilisateur" |
| 6      | SIGINT  | Réinitialiser la machine              |

Donc, en réalité, soit, vous êtes en mode "mono-utilisateur" avec "tout arrêté" et juste un accès root (run-level 1), ou votre système est complètement fonctionnel en mode "multi-utilisateurs".

Pour aller du mode mono au multi, taper la commande "exit"

# Niveau de fonctionnement (suite).

Ordre de ce qui tourne sous le mode multi-utilisateur:

- /etc/defaults/rc.conf (scripts correspondant dans /etc/rc.d).
- Reparamétrages locales avec /etc/rc.conf.
- Les systèmes de fichiers montés comme décrits dans /etc/fstab.
- Démarrage des services de tierce partie à partir des scripts de /usr/local/etc/rc.d.

La plupart des paramètres locaux iront dans:

`/etc/rc.conf`

# Démarrage / Arrêt des services: Révision

Comment démarrer/arrêter un service ?

- `tuer, /etc/rc.d/service stop`
- `/etc/rc.d/service start` ==> système
- `/usr/local/etc/rc.d/script.sh` ==> tierce partie
- `/etc/rc.conf` ==> système & quelques tierce partie
- `/etc/defaults/rc.conf` ==> intouché
- Vielle école: `/etc/rc.local`
- Lire "man rc" *plusieurs fois! :-)*

# Les méthodes d'installation de logiciel

*Il y a trois méthodes pour installer un logiciel sur un système FreeBSD. Elles sont:*

- 1.) Les paquetages FreeBSD et l'utilitaire `pkg`.*
- 2.) La collection de ports `/usr/ports`.*
- 3.) Installation à partir de la source (`gcc make`).*

*Vous allez probablement installer à partir des paquetages, puis des ports, puis de la source*

*Ils ont chacun des avantages et inconvénients.*

# Les commandes de “pkg”

*En général, les facilités de “pkg\_add” and “pkg\_delete” vous permettent d’installer et de supprimer des logiciels sur votre système d’une manière efficace et logique.*

*La commande pkg\_info vous permet de voir ce qui est installé rapidement, et d’avoir des informations détaillées sur chaque paquetage de logiciel installé.*

# Installation de paquetages avec pkg\_add

- Vous pouvez obtenir des “paquetages” à partir d’une source locale (un CD), des sites de FreeBSD, ou de votre réseau local.
- Pour installer un paquetage à partir d’un CD-ROM:

```
pkg_add /cdrom/dir/package_name
```

- Pour installer à partir d’un serveur ftp, vous pouvez faire:

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

# Utilisation de pkg\_info

Vérifier si un paquetage est déjà installé:

```
pkg_info      (lister tout les paquetages installés)
pkg_info | grep moz(trouver tout les paquetages
contenant "moz")
```

Avoir plus d'information sur un paquetage déjà installé :

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

Par exemple "pkg -I bash\\*" renvoie:

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

# Utilisation de pkg\_delete

Si vous avez un paquetage que vous voulez supprimer, vous pouvez simplement taper:

```
pkg_delete package_name
```

Mais, si vous voulez supprimer le paquetage et toutes ses dépendances, vous feriez:

```
pkg_delete -r package_name
```

Mais, *soyez prudent en faisant cela. Vous pouvez vouloir vérifier ce qui passera en faisant:*

```
pkg_delete -n package_name
```

# Installation à partir des Ports

Vous devriez avoir installé la collection de ports pendant l'installation du système. Autrement, Utiliser /stand/sysinstall après l'installation, puis choisir Configure, Distributions, et puis Ports.

Une fois la “collection de ports” installée, vous pouvez voir l'aborescence entière sous /usr/ports. Il y a plusieurs milliers de paquetages de logiciels disponibles.

Cette collection contient les informations de base vous permettant d'installer rapidement un logiciel, et facilement à partir de CD-ROMs ou un site réseau contenant la source du port.

Confère la section 4.5 du manuel du FreeBSD.

# Installation à partir des Ports (suite).

Pour voir si un paquetage de logiciel existe  
comme un port:

```
cd /usr/ports
make search name=package
make search key=keyword
```

Faisons ceci pour “lsof” (LiSt Open Files):

```
cd /usr/ports
make search name=lsof (ou “whereis lsof”)
```

Et le résultat à partir de ceci est :

```
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 à partir de Ports (suite).

De la page précédente, vous allez noter que le port est dans `/usr/ports/sysutils/lsof`.

Si vous avez une connexion réseau...

Vous pouvez simplement taper “`make install`”

Mais, vous pouvez vouloir faire ceci:

- `make`
- `make install`

Pour avoir automatiquement la source des ports d'un serveur local, vous pouvez faire ceci en changeant une variable système:

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

# Installation à partir des Ports (suite).

Vous pouvez installer à partir de cdrom. Si vous avez un cdrom avec les sources des ports (distfiles), alors montez le simplement . Puis vous pouvez faire:

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

Et le port trouvera le distfile sur /cdrom plutôt que sur l'Internet

# CVS et CVSUP

**Une question se pose, “Comment garder la collection de vos ports à jour?”**

**CVS, ou Concurrent Versions System peut faire ceci pour vous.**

**Premièrement, vous devez installer `cvsup-withou-gui`, puis dire à cet outil d’examiner le serveur qui a les dernières versions de la collection de ports et de mettre à jour votre collection locale avec une seule commande comme:**

```
cvsup -g -L 2 -h cvsup.freebsd.org \  
/usr/share/examples/cvsup/ports-supfile
```

# Résumé

- Visant la stabilité et non les bureaux utilisateurs.
- Très, très bonne historique de stabilité et de sécurité.
- Evolutif vers des services à très grande échelle.
- Importante collection de logiciels (13,000+ ports en Juin 2005), incluant la possibilité d'utiliser les paquetages Linux.
- Installation des logiciels par plusieurs méthodes
  - l'utilitaire pkg de FreeBSD est indéniablement supérieur au rpm car il peut résoudre des dépendances.

# Plus de ressources

Cette présentation se trouve à:

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

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



Sujets supplémentaires...

...si il y a du  
temps

# Le noyau du FreeBSD

- Vous pouvez reconstruire le noyau pour ajouter le support de matériel, support de systèmes de fichiers additionnels, etc...
- Ou, supprimer des pilotes extra.
- La source du noyau, si installée, est dans /usr/src/sys
  - “Si il n’y a pas un répertoire /usr/src/sys sur votre système, alors la source du noyau n’a pas été installée. Le moyen le plus facile pour faire ceci est d’utiliser /stand/sysinstall atant que super utilisateur, en choisissant Configure, puis Distributions, puis src, et sys.” (manuel FreeBSD 9.3)
- Pour reconstruire votre noyau, vous utilisez le fichier de configuration par défaut, reparamétré au besoin, et puis recompiler votre noyau, et l’installer dans le /boot.

# Recompilation du noyau FreeBSD

Confère la section 8.3 du manuel de FreeBSD

- Fichier de Configuration dans : `/usr/src/sys/arch/conf`
- Exemple (vieux style):
  - `cp GENERIC /root/kernel/MYNEWKERNEL`
  - `ln -s /root/kernel/MYNEWKERNEL`
  - Editer le fichier MYNEWKERNEL pour définir les options  
voir `/usr/src/sys/arch/conf/NOTES`

*Après avoir édité pour les options*

- `cd /usr/src`
- `make buildkernel kernconf=MYNEWKERNEL`
- `make installkernel kernconf=MYNEWKERNEL`

## Recompilation du noyau (suite).

- Noyau installé comme `/boot/kernel/kernel`
- Ancien noyau dans `/boot/kernel.old/kernel`
- Si le nouveau noyau ne s'initialise pas, aller au menu de démarrage et taper:
  - `unload`
  - `boot /boot/kernel.old/kernel`

## Recompilation du noyau (suite).

Le fichier de configuration du noyau a beaucoup d'options. Pour des explications plus complètes des différentes options, voir (e.g. sur un PC avec un CPU Intel):

- /usr/src/sys/i386/conf/NOTES  
ET, pour des options spécifiques non liées à une architecture confère:
- /usr/src/sys/conf/NOTES

Ou regarder à la section 8.4 du manuel du FreeBSD  
Pour plus d'exemples.

# Noyau et support matériel

FreeBSD est entrain d'aller vers le support matériel en modules. Ces “pilotes” (kernel loadable modules) sont chargés au moment de l'initialisation pour supporter le matériel de votre système.

Certains matériels sont toujours supportés par un logiciel statique directement chargé dans le noyau.

L'utilisation de certains matériels est optimisée en définissant des paramètres d'état du noyau avec `sysctl`.

# Modules chargeables par le noyau et Modules statiques

- Statique (en configuration) – intégré pendant la recompilation  
vs.
- Modules chargeables par le noyau (kld) / boot/kernel.
- Autochargement en utilisant les directives de /etc/rc.conf et/ou en utilisant /boot/loader.conf, qui passe outre /boot/defaults/loader.conf
- Commandes `kldload`, `kldstat`, `kldunload`