

## Portupgrade

Portupgrade a été créée pour simplifier la mise à jour et la suppression des Ports. Comme la majorité des utilitaires de gestion des ports vous pouvez le trouver dans la catégorie "port-mgmt". Pour l'installer, exécutez la commande :

```
# cd /usr/ports/port-mgmt/portupgrade && make install clean
```

Avant la mise à jour des ports listez les Ports déjà installés avec `# pkgdb -F` pour identifier d'éventuels problèmes et les résoudre. Si vous exécutez

```
# portupgrade -a
```

la mise à jour automatique des Ports s'effectue si une nouvelle version de l'application est disponible. Utilisez `'-i'` pour activer le mode interactif et avoir un meilleur contrôle:

```
# portupgrade -ai
```

Si vous souhaitez seulement mettre à jour un application spécifique au lieu de tous les ports, utilisez la commande

```
# portupgrade pkgname
```

Utilisez l'option `'-R'` pour le mode récursif et aussi mettre à jour toutes les dépendances. L'option `'-P'` permet d'utiliser les Packages au lieu des Ports. Avec cette option portupgrade recherche le répertoire définit dans la variable d'environnement `PKG_PATH` pour les Packages. S'il n'y en a pas de disponible sur la machine locale, Portupgrade tente de le télécharger à travers le réseau. s'il n'est disponible ni sur la machine locale, ni sur le réseau, alors le port est utilisé. Portupgrade est un outil puissant, lisez attentivement la documentation des pages man avant de l'utiliser !

## Portaudit

Pour identifier des failles de sécurité l'application `security/portaudit` peut vous aider.

```
# portaudit -Fa
```

liste toutes les failles de sécurité connues pour une application installée sur la machine vous exécutez portaudit. Bien sur, vous pouvez planifier l'exécution de cette tâche avec le cron. Vous pouvez aussi utiliser portaudit avant l'installation d'un Port ou un Package et connaître ses éventuelles vulnérabilités.

## Libérer de l'espace disque

La collection des Ports (Portstree) peut finir par occuper beaucoup de place. Si vous n'utilisez pas toujours 'clean' lors de l'installation d'un Port, vous pouvez nettoyer l'ensemble des Ports de votre système en une seule passe en utilisant portsclean qui fait partie de portupgrade.



```
# portsclean -C
```

Avec le temps, vous pouvez avoir de nombreux fichiers dans le répertoire 'distfiles', ces fichiers des paquets de fichiers source compressés. Pour supprimer les fichiers distfiles dont vous n'avez plus besoin, exécutez :

```
# portsclean -D
```

Si vous voulez seulement effacer tous les fichiers distfiles non requis par un Port installé, exécutez :

```
# portsclean -DD
```

## Désinstaller des Ports

Vous pouvez désinstaller des Ports avec `pkg_deinstall`, qui fait partie de Portupgrade. Vous pouvez aussi utiliser `pkg_delete`. Si vous cherchez une méthode plus simple, vous pouvez utiliser le Port `pkg_cutleaves` qui trouve les packages qui ne sont plus utilisés par une dépendance d'une autre application et qui suggère la désinstallation de chacune de ces applications pour une meilleure maîtrise.

## Gestionnaire graphique de Packages

Il existe bien entendu des interfaces graphiques pour les utilisateurs qui ne souhaitent pas utiliser la console. A côté des applications basées sur les ncurses telles que `pkgfe` et `portbrowser` il existe aussi de remarquables outils bureautiques basés sur KDE et disponibles en standard sur DesktopBSD et PC-BSD. Avec ces outils même des débutants peuvent installer et gérer des Ports avec aisance.

## Autres Informations

Si vous souhaitez obtenir plus d'informations à propos de Ports, nous vous suggérons de consulter à l'excellent manuel de référence FreeBSD :

<http://www.freebsd.org/doc/fr/books/handbook/>

D'autres sources sont ici:

<http://www.freebsd.org/doc/en/books/porters-handbook/>

<http://www.lpthethe.jussieu.fr/~talon/freebsdports.html>

<https://fug-fr.org/>

# FreeBSD

## Ports et Packages

### Introduction

Un système d'exploitation tel que FreeBSD est un programme permettant l'utilisation d'un ordinateur. Il gère la mémoire, les entrées et sorties, et contrôle l'exécution des applications. Sans applications telles que des navigateurs, des éditeurs de texte, des applications de courrier électronique, etc, un système d'exploitation n'est que de peu d'utilité.

Une question fréquemment posée est si FreeBSD à plus d'applications qu'un autre système d'exploitation librement disponible, y compris Debian (veuillez noter que Debian est très différent des Ports). Avec **plus de 16'000 applications** nous sommes certains que vous trouverez celle que vous cherchez. N'oubliez pas qu'à très peu d'exceptions toutes les applications Linux fonctionnent aussi sur FreeBSD.

Vous devez vous demander à présent comment ces applications sont installées, mises à jour, auditées et gérées sur FreeBSD.

### Ports et Packages

Pour installer des applications, FreeBSD offre deux méthodes complémentaires : La collection des Ports pour une installation à partir des sources et les Packages pour une installation de binaires pré-compilés. Vous pouvez utiliser les deux méthodes conjointement pour installer vos applications depuis un média local ou à travers un réseau.



Le Package FreeBSD d'une application consiste en un simple fichier que vous devez télécharger. Le Package contient toute les commandes de l'application, les fichiers de configuration additionnelles et la documentation. FreeBSD propose les outils `pkg_add`, `pkg_delete`, `pkg_version`, et `pkg_info` pour gérer les packages. Avec la commande `pkg_add` vous installez automatiquement les applications que vous souhaitez.

Le port FreeBSD d'une application est un ensemble de fichiers qui automatise la compilation de l'application. Les fichiers d'un Port exécutent toutes les étapes nécessaires à l'installation de l'application pour vous. Avec seulement quelques commandes les fichiers sources d'une application sont téléchargés, dépaquetés, patchés, compilés et installés. Les ports peuvent aussi être utilisés pour créer vos propres packages.

Les Packages et les Ports respectent les dépendances entre les applications. Disons que vous souhaitez installer une application qui dépend d'une librairie et que tous deux sont disponibles dans les Ports ou les Packages FreeBSD. Lorsque vous utilisez `pkg_add` ou le système des Ports vous remarquez que la librairie est installée d'abord si elle n'est pas déjà présente sur votre système.

Pourquoi deux systèmes? Tous deux ont leurs avantages et ce choix ne dépend que de vous.

#### Avantages des Packages

Le package compressé d'une application est normalement plus petit que l'archive compressée de ses sources.

Les packages n'ont pas besoin d'être compilés, ce qui est un gros avantage avec de gros programmes comme KDE, GNOME, OpenOffice et d'autres.

Si vous utilisez des package vous n'avez pas besoin de comprendre comment compiler les logiciels sur FreeBSD.

#### Avantages des Ports

Comme les packages doivent fonctionner sur la majorité des systèmes, les options de compilation sont utilisées avec parcimonie. Lorsque vous installez une application avec les Ports vous pouvez optimiser l'installation avec des options de compilation.

Parfois il y a plusieurs packages pour une application donnée. Pour `ghostscript` il y a `ghostscript-package` et `ghostscript-nox11-package` sans support pour X. Cette distinction est possible avec les packages mais rapidement cela devient facile à manier quand l'application à plus de deux options.

Les licences de certaines applications n'autorisent pas la distribution sous une forme binaire. Ces applications doivent donc être distribuées à partir de leur code source.

Certaines personnes ne font pas confiance aux distributions binaires. Lorsque vous disposez du code source, vous pouvez l'auditer avant de le compiler et être certain qu'il ne contient aucune méchante surprise. Les changements locaux et améliorations rendent aussi la possession du code source nécessaire.

#### Comment trouver les applications?

Si vous ne vous souvenez pas du nom d'une application vous pouvez aller sur des sites tels que FreshMeat (<http://www.freshmeat.net/>) pour chercher l'application appropriée. Puis regardez aussi sur [www.freshports.org](http://www.freshports.org) où vous pouvez rechercher par catégorie et voir quelles nouvelles applications ont été ajoutées, quelles applications ont actuellement des vulnérabilités ou ont d'autres problèmes. Jetez un coup d'oeil, mais attention, on en devient accroc.

#### Installation de la Collection des Ports

Avant d'installer des Ports vous devez ajouter la Collection de Ports (Portstree), qui consiste en un ensemble de fichiers de Makefiles, des patches et de description. Ils sont généralement installés dans `/usr/ports` mais vous pouvez choisir un autre chemin si vous le souhaitez. Vous pouvez utiliser `Sysinstall` (l'outil d'installation FreeBSD), `cvsup` ou `csup`. Nous vous recommandons d'utiliser plutôt `portsnap` qui existe depuis FreeBSD 6.1. Il est extrêmement facile à utiliser.

1. La première fois que vous exécutez `portsnap`, il doit télécharger un snapshot compressé.

```
# portsnap fetch
```

2. Une fois le snapshot téléchargé, une copie "live" de l'arbre des ports peut être extraite dans `/usr/ports/`. Ceci est nécessaire même si un arbre des ports à déjà été créé dans ce répertoire (par ex. en utilisant `cvsup`), car il établit une base à partir de laquelle `portsnap` peut déterminer ultérieurement quelle partie de l'arbre des ports doit mise à jour.

```
# portsnap extract
```

3. Après le téléchargement et l'installation du snapshot initial de l'arbre des ports dans `/usr/ports/`, la mise à jour de l'arbre des ports consiste en deux étapes : la récupération des mises à jour du snapshot et leur utilisation pour la mise à jour des la version live de l'arbre de ports. Ces deux étapes peuvent être effectuées par `portsnap` en une seule commande:

```
# portsnap fetch update
```

Si vous voulez mettre à jour automatiquement votre arbre des ports, vous pouvez bien entendu utiliser cette commande avec le cron.

#### Installer des Ports

Si vous avez mis à jour la Collection des Ports et des droits de root alors vous pouvez installer des Ports. Les Ports supposent que vous avez une connexion à internet en état de marche. Disons que vous souhaitez installer votre shell Bash bien aimé que vous connaissez de votre vieux système Linux. Déplacez vous dans le répertoire du Port, construisez le avec `make`, installez le et nettoyez immédiatement les fichiers contenus dans le répertoire de travail. Vous êtes effrayés? Ne vous inquiétez pas c'est très simple !

```
cd /usr/ports/shells/bash/ && make install clean
```

C'est aussi facile que ça, aller dans le répertoire, construire, installer et nettoyer. Mais qu'en est il de l'installation avec les Packages ?

```
pkg_add -r bash
```

ça ne peut être plus simple. Certains Ports ont aussi des fonctions et des options de sécurité additionnelles. Les options additionnelles peuvent être ajoutées par ex. pour `www/mozilla` et `mail/sylpheed-claws`. Lorsque un Port à des options additionnelles alors ces dernières sont affichées lors de la construction du Port.

#### Mise à jour des Ports

Les nouveaux utilisateurs de ce système sont enclins à constamment sentir le besoin de mettre à jour leurs Ports. Ceci n'est pas nécessaire.

Après la mise à jour de l'arbre des ports avec

```
# portsnap fetch update
```

**Soyez absolument certain que vous avez lu et compris `/usr/ports/UPDATING` avant de mettre à jour un Port.**

Ce fichier décrit les problèmes potentiels et leur solution pour la mise à jour d'un Port. Beaucoup de problèmes avec des applications qui ne fonctionnent plus après une mise à jour imprévue peuvent être évités en lisant ce fichier à l'avance. De grandes applications telles que KDE et GNOME peuvent aussi être mises à jour avec

```
# pkg_delete -r suivi de # pkg_add
```

Une liste des Ports installés et l'état de leur version peuvent être affichés avec:

```
# pkg_version -v
```

Bien sur, il existe de nombreux autres outils disponibles pour mettre à jour ou gérer vos Ports, tels que `portmanager`, `portmaster` et **portupgrade**. Manquant de place pour les traiter tous, nous nous sommes concentrés sur le dernier.