

TRAVAIL PRATIQUE DE SYSTEME D'EXPLOITATION

Sujet : **Gestion de paquetage sous Debian et Redhat**

GROUPE 17

Membres : NYIMBU KALALA CHRISTELLE

SOMWE LUSANGU NATHAN

BONGO BONGO GAEL

MBENZA PHUATI NATHAN

BOKEME NDUAMI KEVIN

ALENGE ZANDELE LIEVIN

MBENZA BALABALA CREDO

MATONDO MABASUKISA GENIUS

MUTWADI GABALU EDO

OKITUNYUNGU EMONGANYA MICHAEL

1. INTRODUCTION

La gestion de paquetages appelé en d'autres termes « gestion de logiciels ou gestion de package ». C'est un processus qui permet de maintenir un système informatique à jour et sécurisé.

Le plus important à savoir et à retenir est que la gestion de paquetages est un outil qui automatise l'installation, la mise à jour, la suppression, la configuration des logiciels.

Donc en d'autres termes, la gestion de paquets est un processus de gestion des logiciels installés sur un système informatique. Elle permet de rechercher, d'installer, de mettre à jour et de supprimer des paquets de logiciels de manière efficace et centralisée. Les paquets de logiciels sont des fichiers contenant les programmes et les fichiers nécessaires à leur installation et à leur utilisation.

La gestion de paquets est souvent utilisée dans les systèmes d'exploitation basés sur Linux, tels que Red Hat, Debian et Ubuntu. Elle permet aux utilisateurs de maintenir leur système à jour en installant les dernières versions des logiciels et des correctifs de sécurité. Elle facilite également l'installation de nouveaux logiciels en résolvant automatiquement les dépendances entre les différents paquets de logiciels.

En résumé, la gestion de paquets est un processus important pour maintenir un système informatique à jour et sécurisé. Elle permet aux utilisateurs d'installer et de gérer facilement les logiciels sur leur système, en résolvant automatiquement les dépendances entre les différents paquets de logiciels.

Dans notre travail nous nous appesantiront plus sur la gestion de paquetages sous Debian et Red hat.

Tout d'abord, Debian est un système d'exploitation libre et open-source basé sur Linux. Il est connu pour sa stabilité, sa sécurité et sa facilité d'utilisation. Debian est utilisé sur des ordinateurs personnels, des serveurs et des superordinateurs. Il est également utilisé comme base pour d'autres distributions Linux, telles qu'Ubuntu et Linux Mint.

La gestion de paquets sous Debian permet aux utilisateurs de rechercher, d'installer, de mettre à jour et de supprimer des paquets de logiciels de manière efficace et centralisée. Elle résout automatiquement les dépendances entre les différents paquets de logiciels, ce qui facilite grandement l'installation et la maintenance des logiciels sur son système.

En outre, Debian dispose d'un système de gestion de paquets avancé appelé Advanced Packaging Tool (APT), qui permet aux utilisateurs de gérer les paquets de logiciels de manière encore plus efficace. APT permet également de gérer les mises à jour de sécurité et les correctifs pour les logiciels installés, ce qui garantit la sécurité et la stabilité du système Debian.

En résumé, la gestion de paquets sous Debian est un processus important pour maintenir un système informatique à jour et sécurisé. Elle permet aux utilisateurs de gérer facilement les logiciels sur leur système, en résolvant automatiquement les dépendances entre les différents paquets de logiciels.

Concernant Red hat, elle est une entreprise qui développe et commercialise des distributions de Linux basées sur des logiciels open-source. Red Hat Enterprise Linux est l'une des distributions les plus populaires et est utilisée par de nombreuses entreprises.

Tout comme Debian, la gestion de paquet sous Red Hat est un processus qui maintient un système informatique à jour et sécurisé. Elle permet aux utilisateurs de rechercher, d'installer, de mettre à jour et de supprimer des paquets de logiciels de manière efficace et centralisée.

Il faut retenir que contrairement à Debian qui utilise l'outil APT, Red hat dispose de yum qui est l'un des outils de gestion de paquets les plus populaires pour les systèmes Red Hat. Il permet de gérer les paquets de logiciels de manière efficace et centralisée en résolvant automatiquement les dépendances entre les différents paquets de logiciels. Il est également capable de rechercher, d'installer, de mettre à jour et de supprimer des paquets de logiciels.

En termes de fonctionnalités, les deux outils sont assez similaires et permettent de rechercher, d'installer, de mettre à jour et de supprimer des paquets de logiciels de manière efficace et centralisée. Cependant, APT est considéré comme étant plus avancé que yum et offre des fonctionnalités supplémentaires telles que la gestion des mises à jour de sécurité et des correctifs pour les logiciels installés.

2. HISTORIQUE

Debian historique ;

Debian a été créée en août 1993 par Ian Murdock, alors étudiant à l'université Purdue. Elle est soutenue par le projet GNU de la Free Software Foundation (FSF), de novembre 1994 à novembre 1995.

Les versions 0.01 jusqu'à 0.90 de Debian sont produites entre août et décembre 1993 ; Ian Murdock écrit alors : « La version 0.91 de Debian sortit en janvier 1994. Elle avait un système de gestion de paquets primitif qui permettait aux utilisateurs de manipuler les paquets mais n'autorisait pas grand-chose d'autre (il ne possédait certainement pas de dépendances ou d'options analogues). À partir de ce moment-là, quelques douzaines de personnes travaillaient sur Debian, alors que je devais toujours assembler les versions moi-même. La version 0.91 fut la dernière version faite de cette manière. [...] »

Une grande partie de l'année 1994 fut consacrée à organiser le projet Debian de façon que les autres puissent plus directement contribuer, comme pour la réalisation de Dpkg (Ian Jackson fut très largement responsable de cette dernière). Si je me souviens bien, il n'y eut pas de version officielle en 1994, bien que nous en eûmes un certain nombre en interne, à chaque fois que nous progressions dans l'avancement de la distribution. [...]

La Debian 0.93, en version 5, sortie en mars 1995, fut la première version « moderne » de Debian : il n'y avait jamais eu autant de développeurs (bien que je ne puisse me rappeler combien), chacun avait maintenant ses propres paquets et Dpkg fut utilisé pour installer et entretenir tous ces paquets après l'installation du système de base. [...]

La Debian 0.93, en version 6, sortie en novembre 1995, fut la dernière version au format a.out. Il y avait environ 60 développeurs pour entretenir les paquets de la version 0.93R6. Si je me souviens bien, dselect fit son apparition dans cette version qui fut ma version favorite de Debian. [...] »

Murdock cesse de travailler activement sur le projet en mars 1996, durant la préproduction de Debian 1.0. Cette dernière est renommée « 1.1 » pour éviter toute confusion avec une version précédente qui avait été nommée par erreur « 1.0 » par un fabricant de disque compact ; cet incident mena au concept d'images ISO « officielles », de façon à éviter ce genre de bévues à l'avenir.

Durant le mois d'août 1995 (entre les versions 0.93R5 et 0.93R6), Hartmnut Koptein commence le premier portage de Debian pour l'architecture m68k (Motorola). Il raconte :

« De très nombreux paquets étaient construits autour de l'architecture i386 (« petit bout », - m486, - O6 et autres options de ce genre de la bibliothèque libc4) et ce fut un travail énorme que d'avoir une base de paquets de départ sur ma machine (un Atari Medusa 68040, 32 MHz). Après trois mois (en novembre 1995), je mis à disposition 200 paquets, sur les 250 disponibles, tous pour la bibliothèque libc5 ! »

Dès lors, le projet Debian s'est développé en incluant de nombreux portages vers d'autres architectures, dont le micro-noyau GNU Mach, qui mènera au développement de Debian Hurd, projet fondé en 1998 par Marcus Brinkmann, avec l'aide de Gordon Matzigkeit (employé de la FSF à l'époque).

Un des tout premiers membres du projet, Bill Mitchell, se rappelle, au sujet du noyau Linux :

« On devait être entre la version 0.99r8 et 0.99r15 lorsque l'on a débuté. Pendant très longtemps, je fus capable de compiler un noyau en moins de 30 minutes sur une machine dotée d'un 386 à 20 MHz, et j'étais ainsi capable d'installer une Debian dans le même temps avec moins de 10 Mo d'espace disque. [...] »

Je me souviens que l'équipe initiale comprenait Ian Murdock, moi-même, Ian Jackson, un autre Ian dont je ne me souviens pas le nom de famille, Dan Quinlan, et quelques autres personnes dont je ne me souviens pas des noms. Matt Welsh faisait aussi partie du groupe initial, ou l'a rejoint à ses tout

débuts (il a depuis quitté le projet...). Quelqu'un créa une liste de discussions et nous nous mîmes au travail. [...]. Si je me souviens bien, nous ne partîmes pas d'un plan défini, et nous ne partîmes pas sur le fait de créer ensemble un plan avec une approche très organisée. Dès le début, si je ne me trompe pas, nous rassemblâmes aléatoirement les sources d'un certain nombre de paquets. Avec le temps, nous finîmes par finaliser une collection d'articles qui seraient nécessaires au cœur de la distribution : le noyau, un shell, update, getty, de nombreux autres programmes et de fichiers de configuration requis pour initialiser le système ainsi que tout un jeu d'utilitaires. »

Le nom tire son origine des prénoms du créateur de Debian, Ian Murdock, et de son épouse, Debra. Debian a eu plusieurs dirigeants depuis ses débuts en 1993. Ian Murdock a fondé Debian en août 1993 et a mené le projet jusqu'en mars 1996. Bruce Perens a dirigé Debian d'avril 1996 à décembre 1997.

Redaht historique ;

En 1993, Bob Young lance l'AAC Corporation, une société ayant pour but de vendre des accessoires logiciels pour Linux et Unix. Puis en 1994, Marc Ewing a créé sa propre version de Linux, qu'il a appelé Red Hat Linux. Ewing l'a publiée en octobre et elle a été connue pour avoir été la version Halloween. Young a acheté l'affaire d'Ewing en 1995, et les deux entreprises ont fusionné pour donner l'entreprise Red Hat Software[8] dirigée par Young comme Chief executive officer[9].

L'origine du nom de Red Hat remonte au penchant de Marc Ewing pour les chapeaux rouges, en particulier un chapeau lacrosse de la Cornell University qui lui a été offert par son grand-père alors qu'il était étudiant à la hat Carnegie Mellon University[10],[11]. Le rouge du chapeau rouge Fedora est associé symboliquement avec l'idée de libération et de remise en cause de l'autorité hiérarchique[12], voire désigne en anglais quelqu'un qui s'introduit dans une zone sécurisée pour informer ensuite le propriétaire qu'il peut réparer les failles de sécurité s'il l'engage[13].

En 2012, Red Hat est devenu le premier éditeur de technologies Open Source à atteindre un chiffre d'affaires de plus de 1 milliard de dollars. En 2016, ce chiffre d'affaires a dépassé les 2 milliards de dollars.

3. IMPORTANCE

La gestion de paquetage est une fonctionnalité essentielle des distributions Linux. Elle permet d'installer, de mettre à jour et de supprimer des paquets de manière simple et sécurisée.

Voici quelques exemples de l'importance de la gestion de paquetage :

- * Un utilisateur qui souhaite installer un nouveau logiciel peut simplement utiliser une commande pour le faire. Il n'a pas besoin de télécharger les sources du logiciel et de le compiler.
- * Un administrateur système peut utiliser la gestion de paquetage pour mettre à jour tous les logiciels sur un système en une seule fois. Cela permet de réduire le temps et les efforts nécessaires pour maintenir un système à jour.
- * Un administrateur système peut utiliser la gestion de paquetage pour créer des images système personnalisées. Cela permet de déployer des systèmes Linux rapidement et facilement.

En conclusion, la gestion de paquetage est une fonctionnalité essentielle des distributions Linux. Elle offre de nombreux avantages, tels que la simplicité, la sécurité, la fiabilité et la conservation des ressources.

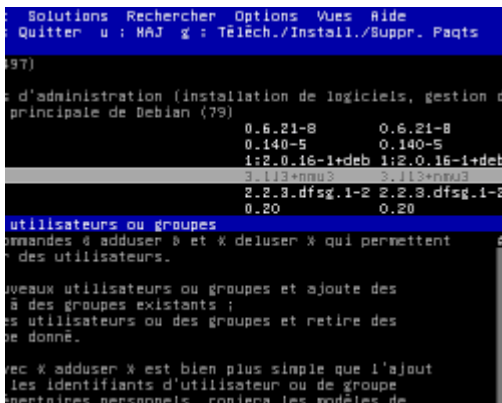
AVANTAGES DE LA GESTION DE PAQUETAGE SOUS DEBIAN ET RED HAT

Simplicité : La gestion de paquetage permet d'installer, de mettre à jour et de supprimer des paquets en quelques commandes. Cela facilite la gestion de logiciels sur un système Linux.



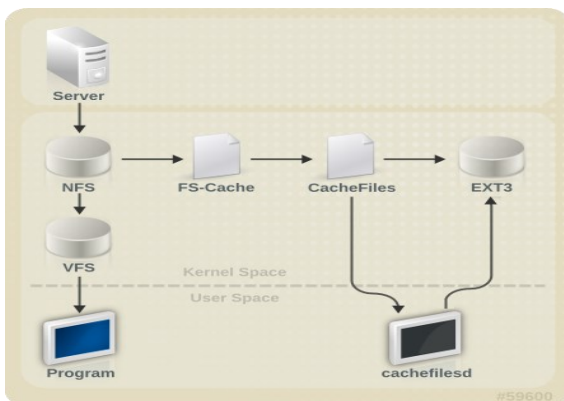
[Image, Simplicité de la gestion de paquetage sous Debian et Red Hat]

Sécurité : Les paquets Linux sont signés numériquement pour garantir leur authenticité. Cela permet de protéger les systèmes Linux contre les logiciels malveillants.



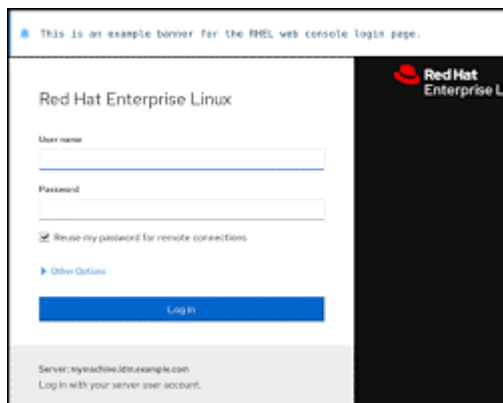
[Image, Sécurité de la gestion de paquetage sous Debian et Red Hat]

Fiabilité : Les paquets Linux sont généralement bien testés et fiables. Cela réduit le risque de problèmes lors de l'installation ou de la mise à jour de logiciels.



[Image, Fiabilité de la gestion de paquetage sous Debian et Red Hat]

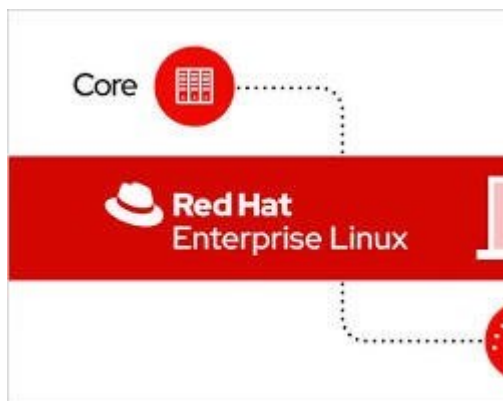
Conservation des ressources : La gestion de paquetage permet d'installer et de mettre à jour les logiciels sans avoir à télécharger et à compiler les sources. Cela permet de conserver les ressources système.



[Image, Conservation des ressources de la gestion de paquetage sous Debian et Red Hat]

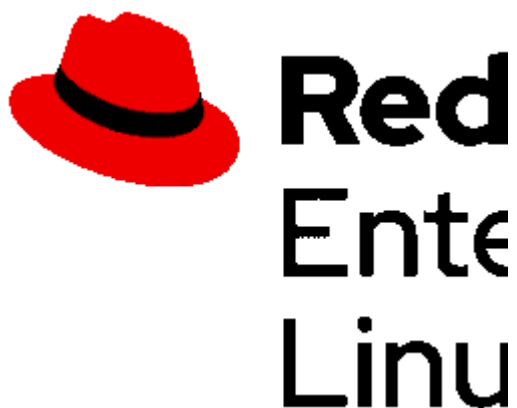
INCONVENIENTS DE LA GESTION DE PAQUETAGE SOUS DEBIAN ET RED HAT

Flexibilité limitée : La gestion de paquetage peut limiter la flexibilité de l'installation de logiciels. Par exemple, il peut être difficile d'installer des logiciels qui ne sont pas disponibles dans les référentiels officiels.



[Image, Flexibilité limitée de la gestion de paquetage sous Debian et Red Hat]

Complexité accrue : La gestion de paquetage peut rendre la gestion des logiciels plus complexe. Par exemple, il peut être difficile de comprendre les dépendances entre les paquets.



[Image, Complexité accrue de la gestion de paquetage sous Debian et Red Hat]

Problèmes de compatibilité : La gestion de paquetage peut entraîner des problèmes de compatibilité entre les logiciels. Par exemple, il peut être difficile de faire fonctionner des logiciels qui ont été compilés pour une version différente de Linux.



[Image, Problèmes de compatibilité de la gestion de paquetage sous Debian et Red Hat]

En conclusion, la gestion de paquetage est une fonctionnalité importante des distributions Linux. Elle offre de nombreux avantages, mais elle peut également avoir quelques inconvénients.

4. GESTION DES PAQUETS

1. Sous Debian ;

Pour ce faire on a besoin **apt** (Advanced Packaging Tool). Ce dernier nous permet de faire de recherches, d'avoir des informations sur des logiciels et aussi les installer, les supprimer et les mettre à jour.

Les principales commandes de l'outil de gestion des paquets apt sous Debian sont :

1. **apt update** :

Cette commande met à jour la liste des paquets disponibles dans les sources de paquets configurées sur votre système. Elle est utilisée pour synchroniser les informations de paquets avec les dépôts en ligne, afin de s'assurer que vous disposez des dernières informations sur les paquets disponibles.

2. **apt upgrade** :

Cette commande permet de mettre à jour les paquets installés sur votre système Debian. Elle télécharge et installe les nouvelles versions des paquets disponibles, en tenant compte des dépendances. Elle est généralement utilisée après la commande apt update pour appliquer les mises à jour disponibles.

3. **apt install** :

Cette commande permet d'installer un ou plusieurs paquets sur votre système Debian. Vous pouvez spécifier les noms des paquets que vous souhaitez installer en les séparant par des espaces. apt résoudra automatiquement les dépendances et téléchargera les paquets requis depuis les sources configurées.

4. **apt remove** :

Cette commande permet de supprimer un ou plusieurs paquets installés sur votre système Debian. Elle supprime les paquets spécifiés ainsi que leurs fichiers de configuration, mais ne supprime pas les dépendances qui ne sont plus nécessaires à d'autres paquets.

5. apt purge :

Cette commande a le même effet que apt remove, mais en plus de supprimer les paquets, elle supprime également les fichiers de configuration associés. Cela garantit une suppression complète des paquets et de leurs paramètres.

6. apt autoremove :

Cette commande permet de supprimer automatiquement les paquets qui ont été installés en tant que dépendances, mais qui ne sont plus nécessaires à aucun autre paquet installé sur le système. Elle libère de l'espace disque en supprimant les paquets obsolètes.

7. apt search :

Cette commande permet de rechercher des paquets disponibles dans les dépôts configurés sur votre système Debian. Vous pouvez spécifier un terme de recherche et apt affichera une liste de paquets correspondants.

8. apt show :

Cette commande affiche les informations détaillées sur un paquet spécifique, telles que sa description, sa version, les dépendances requises, les fichiers qu'il contient, etc. Vous pouvez spécifier le nom d'un paquet pour obtenir ses informations.

9. apt list :

Cette commande affiche une liste de tous les paquets installés sur votre système Debian. Vous pouvez utiliser des options supplémentaires pour filtrer les résultats et afficher uniquement les paquets spécifiques, les paquets obsolètes, etc.

10. apt full-upgrade :

Cette commande effectue une mise à niveau complète du système en installant les nouvelles versions des paquets, en supprimant les paquets obsolètes et en résolvant les éventuels problèmes de dépendances. Elle est souvent utilisée lors des mises à niveau majeures du système.

Toutes ces commandes apt couvrent les fonctionnalités de base pour la gestion des paquets sous Debian. Il existe également d'autres commandes et options disponibles pour des tâches plus avancées comme :

apt edit-sources, apt reinstall, apt satisfy,

Ces dernières ne sont pas incluses dans la liste de commande car elles sont utilisées pour éditer les sources de paquets pas pour faire la gestion des paquets.

Il est important de mettre à jour généralement vos paquets pour garantir la sécurité et la stabilité du système.

2. Sous Red hat ;

Pour faire on va utiliser l'outil **yum**, qui est l'outil de gestion des paquets utilisé dans les distributions Red Hat, CentOS et aussi Fedora :

1. **yum install package** : Installe un paquet et ses dépendances.
2. **yum remove package** : Supprime un paquet installé.
3. **yum update** : Met à jour tous les paquets installés avec leurs dépendances.
4. **yum upgrade** : Met à jour les paquets installés, en installant les nouvelles versions si disponibles.
5. **yum search keyword** : Recherche un paquet correspondant au mot-clé spécifié.
6. **yum info package** : Affiche des informations détaillées sur un paquet.
7. **yum list** : Affiche la liste de tous les paquets disponibles.
8. **yum list installed** : Affiche la liste de tous les paquets installés.
9. **yum provides file** : Affiche le paquet qui fournit un fichier spécifié.
10. **yum deplist package** : Affiche les dépendances d'un paquet spécifié.
11. **yum clean all** : Supprime tous les caches et les métadonnées des paquets, pour les reconstruire lors de la prochaine utilisation de yum.
12. **yum repolist** : Affiche la liste des dépôts de paquets configurés.
13. **yum history** : Affiche l'historique des transactions de yum, y compris les installations, les mises à jour et les suppressions de paquets.
14. **yum groupinstall group** : Installe un groupe de paquets spécifié.
15. **yum groupremove group** : Supprime un groupe de paquets installés.
16. **yum check-update** : Vérifie si des mises à jour sont disponibles pour les paquets installés.
17. **yum provides command** : Affiche le paquet qui fournit une commande spécifiée.

Il est important de mettre à jour généralement vos paquets pour garantir la sécurité et la stabilité du système.