Mini_HOWTO Mettez à jour votre distribution Linux

Greg Louis, glouis@dynamicro.on.ca

v1.11, 6 Juin 1996

Trucs et astuces pour mettre à jour une distribution à partir d'une autre.

Contents

1 IMPORTANT !!! Désistement de responsabilité et propriété légale

Le procédé pour lequel ce document essaie d'être un guide est par nature dangereux pour les programmes et les données stockées sur votre ordinateur. Vous utilisez une telle méthode à vos risques et périls. Les étapes décrites dans ce document ont fonctionné pour l'auteur ; il n'y a aucune garantie qu'elles fonctionnent chez vous, ni que vous pouvez les suivre sans dégâts importants pour les programmes et/ou les données de votre ordinateur. Vous effectuez sous votre propre responsabilité toutes les utilisations des informations contenues ici, et l'auteur (NdT : ou le traducteur) ne peut(vent) être tenu(s) pour responsable(s) en aucune façon, quels que soient les dommages ou les problèmes dont vous pourriez souffrir par ces utilisations.

Les commentaires et les questions peuvent être envoyées à l'auteur. Les rapports de mises à jour de systèmes complexes réussies sont particulièrement les bienvenues, pour être utilisées dans les prochaines versions.

2 Modifications depuis la version 1.1

- Ajout de cette section historique.
- Ajout de la suggestion de Zoltán Hidvégi concernant mtime et ctime. Merci Zoltán !
- Ajout d'une section Remerciements

3 Introduction

3.1 Comment tuer et réincarner votre machine Linux !

Le but de ce document est d'offrir des astuces vous aidant dans la suppression et la réinstallation d'un système Linux. Ce n'est en aucun cas un livre de recettes parfaites, mais j'espère qu'il vous donnera des indications concernant ce à quoi vous devez penser et dans quel ordre vous devez faire les choses. Cela aurait été une aide pour moi si quelqu'un d'autre avait écrit quelque chose comme cela avant que je fasse ma première mise à jour ; c'est pourquoi j'espère que cela sera une aide pour vous si vous avez une machine Linux à mettre à jour.

Ne suivez pas tout à la lettre néanmoins : vous pouvez avoir plus ou moins de «bouteille». Même les noms de répertoires cités dans ce document peuvent être différents de ceux que vous devrez utiliser : par exemple, certains personnes utilisent /usr/home à la place de /home, d'autres l'appellent /u, et certains (petit frisson :) mettent même tous leurs utilisateurs directement dans /usr même ! Je ne peux être spécifique pour votre système, donc j'ai simplement utilisé les noms tels qu'ils étaient sur le mien.

Vous noterez aussi que j'utilise les distributions Slackware, et que je suppose que vous disposez de suffisamment de mémoire vive et d'espace disque pour installer les sources du noyau Linux et compiler votre propre noyau. Si votre système est différent, certaines de mes recommandations ne s'appliqueront pas ; mais j'espère que vous trouverez l'idée générale sous-jacente utile dans votre projet de mise à jour.

3.2 Pourquoi quelqu'un voudrait-il faire cela ?

Bonne question ! S'il est possible de l'éviter, ne le faites pas ! (Ceci est la plus importante recommandation de tout ce guide !!!). Mais il existe des cas où vous devrez le faire.

Par exemple, j'ai installé un disque dur de 4 Go et j'ai alors découvert que le cru Linux Slackware 2.0 ne savait pas gérer un disque dur d'une taille supérieure à 2 Go ; j'ai alors été terriblement embarrassé. Donc je devais faire une mise à jour vers la Slackware 2.3. Cette mise à jour a été une expérience épuisante, et c'est en partie pour cela que j'ai écrit ces notes. J'ai quasiment tout fait de travers, et c'est seulement la chance et le fait que je possède une autre machine sous Linux qui m'ont sauvés du désastre.

Voici un autre exemple : j'ai découvert que je ne pouvais pas réussir à compiler un noyau Linux a.out qui marche dans la série des 1.3 en utilisant une Slackware 2.3 directement après l'installation (sur une autre machine, pas sur celle que j'ai charcutée). J'ai retroussé mes manches, acheté une Slackware 3.0 en cd-rom et je suis passé en format ELF. Cette fois, la réinstallation s'est mieux passée, en partie à cause de la première expérience douloureuse, et cela est l'origine de la plupart des idées que je vous propose ici.

3.3 Ai-je à «détruire et réinstaller» ?

Chose étrange, cela est plus sûr. Si vous installez par dessus un système Linux, il y a des chances que vous ayez un mélange de nouveaux et d'anciens exécutables, d'anciens et de nouveaux fichiers de configuration, et généralement un fouillis à essayer d'organiser. Faire un nettoyage par le vide et ensuite réinstaller seulement ce que vous savez nécessaire est une méthode violente mais efficace pour avoir un résultat propre. (Bien sur nous parlons ici de l'installation d'une distribution Linux complète et non pas de la mise à jour d'un ou deux paquetages ! La meilleure façon d'éviter d'avoir à faire une réinstallation complète est précisément de garder chaque composant – en particulier gcc et ses bibliothèques, et les binutils – à jour. Si les programmes que vous utilisez sont raisonnablement à jour, et vous pouvez réaliser ceci en récupérant, et en compilant si nécessaire, les derniers sources de temps en temps, alors il n'y a nul besoin d'une mise à jour massive.)

Comme Patrick Volkerding le souligne (lui aussi recommande la procédure du grand nettoyage pour les mises à jour), installer ELF par dessus un système basé sur a.out engendre un désastre ; en tout cas, si vous en savez assez pour essayer cela, vous n'avez pas besoin de lire ce guide !

Même sans cette complication, vous feriez mieux d'installer à partir de zéro.

3.4 Combien de temps cela va-t-il prendre ?

Cela dépend, bien sur, de la complexité de votre système. Mais je suppose que, pour l'installation réussie (pour l' autre ? – ne me le demandez pas ! :) j'ai passé environ 10 heures à faire des sauvegardes, six heures à installer tout le système jusqu'au point à partir duquel j'ai pu autoriser les connexions, et une autre demi-journée, ou quasiment, à restaurer les choses moins importantes. Ensuite, au fur et à mesure, j'ai bien découvert des petites choses qui n'étaient pas exactement comme je les voulais – je les ai corrigées dès que je les rencontrais – mais pour l'essentiel, une vingtaine d'heures devraient suffire pour réinstaller un système raisonnablement complexe. Peut-être moins si vous réinstallez à partir d'un disque dur (j'ai utilisé un cd-rom) ou plus à partir de disquettes. Peut-être moins si vous utilisez un Pentium rapide, plus si vous utilisez un 386. Mais c'est de cet ordre de grandeur.

Cette introduction est finie. Voyons maintenant comment réaliser cela, une fois que vous avez décidé que cela devait être fait. Armez vous de courage, respirez un bon coup, et :

4 Notez tout ce que vous faites.

C'est extrêmement profitable d'avoir un rapport sur tout ce que vous avez fait pour préparer, et réaliser, les modifications. En particulier, la liste de toutes les sauvegardes que vous allez réaliser durant la préparation de la destruction de votre système actuel est très importante.

5 Faites une sauvegarde complète du système actuel.

De façon générale, les sauvegardes sont écrites sur des média à accès séquentiel. Dans ce cas là, vous ne voudrez pas utiliser cette sauvegarde complète pour restaurer un nombre important de fichiers ; il y a trop de fichiers dedans que vous ne souhaitez pas restaurer. Il vaut mieux créer des petites sauvegardes de zones que vous savez que vous restaurerez entièrement. J'ai listé un ensemble d'exemples plus loin.

Pourquoi alors devez-vous commencer par une sauvegarde totale ? Deux raisons simples : d'abord en cas d'échec catastrophique durant l'installation du nouveau système, vous aurez une solution pour retourner au point de départ avec une peine minimale. Ensuite, quelle que soit la méticulosité avec laquelle vous avez préparé la nouvelle installation, il y a une large probabilité qu'un ou deux fichiers importants aient été oubliés. Dans ce cas la contrainte de la restauration de ces un ou deux fichiers de la sauvegarde complète sera préférable à la gêne d'une continuation sans eux.

Pour économiser du temps et de l'espace disque, si vous avez encore le médium contenant la distribution de votre ancienne version de Linux, vous pouvez ne sauvegarder que les fichiers dans les mtime ou ctime sont plus récents que ceux de la date de son installation.

6 Sauvegardez /etc et ses sous-répertoires sur une ou plusieurs disquettes.

Ceci est l'autre extrémité : vous n'allez pas restaurer ces fichiers (pour la plupart en tout cas) ; vous allez les comparer avec les nouveaux créés durant l'installation. Pourquoi ? Parce que les nouveaux peuvent contenir des informations que les anciens ne contenaient pas, ou exprimer ces informations de façons différentes. Changements de protocoles, ajout de nouveaux outils ou implantation de nouvelles fonctionnalités dans des outils existants peuvent engendrer des changements dans les formats des fichiers de configuration et dans ceux des scripts de lancement que la sous-arborescence /etc contient, et vous devrez sûrement editer vos anciennes données contenues dans ces fichiers pour respecter les nouveaux formats et tirer avantage des améliorations.

7 Faites des sauvegardes séparées pour chaque groupe de fichiers que vous souhaitez garder.

Ceci est la partie la plus variable du travail, et tout ce que je peux réellement faire pour vous aider est de vous décrire ce que je fais sur mon système, dans l'espoir que cela vous servira comme un guide de base. Basiquement, vous devez regarder dans chaque répertoire qui contient :

- des fichiers qui ne font pas partie de l'installation Linux standard

ou

- des fichiers qui sont actuellement plus récents que ceux que vous allez installer lorsque vous ferez votre nouvelle installation de Linux

et isoler seulement ceux que vous voulez garder.

(Une autre stratégie possible est de sauvegarder tous les fichiers dont les mtime ou ctime sont plus récents que le jour de votre précédente installation de Linux, comme mentionné plus haut, et ensuite de les restaurer à partir de là. Si vous faites cela, vous devez prendre en compte le fait que la nouvelle distribution de Linux peut contenir des versions de certains fichiers qui sont plus récentes que celles que vous avez sauvegardées.

Dans mon cas, je suis arrivé à faire un fichier .tgz sur un média de sauvegarde pour chacun des éléments

- /usr/lib/rn
- /usr/lib/smail
- /usr/lib/trn (le reste de /usr/lib sera réinstallé)
- /usr/local/src
- /usr/local/bin
- /usr/local/lib
- /usr/local/lpfont
- /usr/local/man
- /usr/local/sbin
- /usr/local/thot (il y avait d'autres fichiers dans /usr/local dont je n'avais pas besoin)
- /usr/openwin
- /usr/src/lilo-17 (car ma nouvelle Slackware avait encore une version 16)
- /usr/src/linux-1.2.13 (car j'avais fait ma configuration)
- /usr/X11R6/lib/X11/app-defaults
- /usr/X11R6/lib/X11/initrc (le reste de XFree86 sera réinstallé)
- /var/named
- /var/openwin
- /var/texfonts

Ma machine était relativement simple car il n'y avait aucun fichier de spool à prendre en compte. Je n'utilisais pas de spool pour les news sur cette machine, et il y avait seulement deux utilisateurs ; il était très facile de récupérer tout le courrier électronique lu avant l'arrêt. Sinon, les répertoires /var/spool auraient été sauvegardés à la dernière minute. (Et, bien sur, les répertoires contenant la liste de news et des serveurs !)

8 Préparez des disquettes root et boot pour la nouvelle installation.

Des détails pour faire ceci peuvent être trouvés dans le guide d'installation de votre nouvelle distribution.

9 Formattez des disquettes pour le noyau temporaire et l'installation finale.

Vous aurez besoin de deux disquettes, une pour chaque.

Lorsque tout cela est fait, vous êtes prêts pour le Grand Moment. L'étape suivante met le système hors service.

10 Bloquez les connexions et sauvegardez les arborescences /root et /home.

Ceci est la dernière chose à faire sur le vieux système avant que vous ne l'effaciez, pour conserver la version la plus à jour des informations sur les utilisateurs et le super-utilisateur.

11 Démarrez à partir des nouvelles disquettes boot et root.

12 Effacez les partitions Linux avec fdisk et recréez-les.

Le guide d'installation vous expliquera comment faire cela, ce qui va effacer votre ancien système. A partir de maintenant, vous êtes dépendants de la qualité des sauvegardes que vous avez faites aux étapes précédentes ! Vous avez été averti !

13 Lancez la nouvelle installation.

Il y a déjà plusieurs bons documents décrivant comment faire cela, donc je ne vais pas détailler. Reprenez ici quand le nouveau système bootera à partir de son disque dur.

Pendant ce processus, pensez à faire une disquette de boot, car le noyau que l'installation de Linux installe doit être remplacé, et des accidents peuvent survenir durant cette étape. Pensez aussi à installer les paquetages de développement et les sources du noyau.

14 Configurez le swap.

Avec le nouveau système Linux démarré à partir du disque dur, éditez /etc/fstab et ajoutez votre partition de swap. Ensuite lancer la commande "swapon -a". Je ne sais pas pourquoi, mais l'installation de la Slackware ne vous propose pas ceci lorsque la partition de swap existe déjà. Donc, lorsque vous démarrez votre nouveau système et que les scripts rc.S essaient d'activer le swap, la partition ne peut être trouvée dans le fichier fstab et le swap n'est pas activé. Cette étape corrige cela.

15 Restaurez la configuration du répertoire /etc et de ses sous-répertoires.

Comme décrit plus haut, vous ne pouvez pas juste recopier tous les vieux fichiers dans /etc et espérer que tout va marcher correctement après. Avec certains fichiers, vous pouvez faire cela ; par exemple /etc/XF86Config (aussi longtemps que vous utilisez la même version de Xfree86 – et le même matériel vidéo – dans l'ancienne et la nouvelle installation). Pour la plus grande partie cependant, il vaut mieux utiliser diff pour comparer les anciens et les nouveaux fichiers avant de faire les copies. En particulier, surveiller les changements significatifs dans les fichiers de /etc/rc.d, qui peuvent nécessiter de rétablir votre ancienne configuration en éditant à la main, plutôt que de recopier vos anciens scripts rc à partir de votre sauvegarde. Une fois que tout est fini, rebootez.

16 Configurez et recompilez le noyau Linux.

Même si vous n'avez absolument pas besoin de faire cela pour obtenir un noyau qui supporte votre matériel, il est important de faire cela pour avoir un noyau qui ne contienne pas des quantités de pilotes inutiles pour votre machine. Pour plus de détails, regardez le Kernel HOWTO. Installez le noyau recompilé sur une disquette d'abord ; une fois que vous avez vérifier qu'il démarre, ok, installez le sur le disque dur, lancez lilo si vous l'utilisez, et redémarrez.

17 Restaurez le contenu des sauvegardes faites précédemment.

Certains exécutables peuvent nécessiter d'être réinstallés à partir des répertoires des sources ; j'ai eu à faire cela pour lilo par exemple, car ma version était plus récente que celle de l'installation de la Slackware et que je ne m'étais pas préoccupé de sauvegarder l'exécutable placé dans /sbin. Vous devrez vérifier les programmes restaurés et confirmer l'existence et l'exactitude des fichiers de configuration, des bibliothèques, etc. Dans certains cas, vous pourrez avoir à restaurer des choses dans un ordre précis ; vous aviez pris des notes pendant la sauvegarde, n'est-ce pas ? ;-)

18 Revoyez la sécurité.

Vérifier les permissions sur les fichiers et les répertoires pour être sur que l'accès n'est ni trop restrictif ni trop libre. Je trouve que la Slackware laisse un environnement trop ouvert à mon gout, donc je me promène en changeant les 755 en 711 pour les exécutables dans les répertoires bin/ et autres choses du genre. Ou même en 700 pour ceux des répertoires sbin/. Une attention particulière est nécessaire si vous avez un serveur FTP ; mais vous y avez alors probablement déjà pensé. :)

19 Autorisez les connexions.

Ça tourne et ça fonctionne. Pendant un petit moment, il y aura probablement des détails à régler, mais le gros du travail est fait. Réjouissez-vous !

20 Désolé, mais une fois encore :

VOUS UTILISEZ CES INFORMATIONS À VOS RISQUES ET PÉRILS !

(Voir le désistement de responsabilité au début du document.)

21 Remerciements

Je remercie Zoltán Hidvégi pour sa contribution au contenu de ce mini-HOWTO.

22 Traduction

Ce document a été traduit par Benoît Sibaud (pas d'adresse fixe, cherchez dans les news). NdT : N'hésitez pas à user et à abuser des HOWTO et Mini-HOWTO, ils sont là pour ça. Certaines parties de ce document sont décrites en détails dans d'autres HOWTO et Mini-HOWTO.