Installation et Kit de survie de Debian

-

Installation détaillée d'outils et services majeurs

La dernière version de ce document est disponible sur http://michauko.org/docs/debian_testing/

Le blog qui accompagne (complète) ce document est sur http://michauko.org/blog/

Sommaire

1 Avant de foncer

1.1 Pourquoi cette doc ?

Je tenais à faire une doc d'installation pratique de Debian pour initier simplement les débutants à « un système Linux » et plus spécifiquement à la distribution Debian ainsi qu'un kit de survie une fois le beau système installé avec rien dedans. Vous y trouverez aussi la description détaillée de certains sujets parfois complexes (serveur de mails, web, base de données etc).

Par rapport aux versions précédentes de ce document, il y a quelques évolutions :

- Je décris maintenant 3 modes d'installation à la fois : installation de Debian « stable » en mode expert, Debian « testing » en mode expert et Debian « stable » en mode normal. **Le tout via un tableau comparatif de photos d'écrans**. Ainsi, les gens un peu habitués mais qui hésiteraient à utiliser le mode expert pourront être rassurés et cela montre les différences à venir entre la version actuelle (stable) et la prochaine (testing – cf. chapitre 1.5 pour plus de détails sur les différentes versions).

- J'ai supprimé tout ce qui concerne la partie graphique. Mon point de vue maintenant est que Debian devrait être réservée à un usage serveur (donc pas de partie graphique). Pour avoir un ordinateur bureautique convivial sans se prendre la tête, sans avoir des packages un peu vieillot (même en « testing » parfois), autant utiliser Ubuntu. Et tant pis si je perds des fans ☺ Les indécrottables de Debian n'ont de toute manière pas besoin de lire ce guide, sauf pour me corriger et donner des avis pertinents (ie, pas des Trolls ☺)

Jusqu'à certaines versions précédentes de cette documentation, il y avait en vrac des chapitres décrivant tout un tas de sujets (montage de serveurs de jeux online dédiés, installation d'une galerie de photos etc). J'ai supprimé ces sujets pour les reprendre dans un blog accompagnant cette documentation. Vous y êtes cordialement invités :

http://michauko.org/blog/

Ainsi, je vais recentrer cette doc sur le bloc « installation / kit de survie / gros sujets serveurs » et la publication de petits articles à travers un blog sera plus rapide et plus fréquente que la mise à jour d'une documentation monolithique (MAJ moyenne tous les 9 mois ; accouchement difficile, en gros).

1.2 Trolls

Des trolls, il y en a sûrement. Je les tuerai par mon silence ☺ Les plus fréquents comme : « Mandriva c'est moins bien qu'Ubuntu qu'est mieux que blablabla » ; « KDE et Gnome c'est naze, vive Fluxbox » ; « on dit GNU/Linux et pas Linux »…

1.3 Les basiques

Je ne détaille aucun basique, genre « ls », « vi » ou encore le concept des permissions sur les fichiers. Vous trouverez des tonnes de sites Internet le faisant parfaitement. Je considère que c'est acquis. Je veux juste vous initier aux spécificités de ce merveilleux OS. Surtout pour en faire un serveur robuste. Vu le marché qui propose pour pas trop cher par mois des serveurs dédiés

en location, avec bande passante de malade etc, n'importe qui peut prétendre installer un serveur (je n'ai pas dit prétendre le sécuriser…). Autant installer Debian dessus ☺

1.4 Distribution, cékoidon ?

Debian est une « distribution Linux ». Une distribution, c'est (de mon point de vue) :

- Un système basé sur le « noyau » Linux. Le noyau n'est « que » la couche profonde qui permet de gérer l'accès à vos périphériques, votre réseau, les ressources utilisateurs, organiser les processus (programmes qui tournent) etc etc. On dit « j'ai installé linux », mais c'est un abus de langage. Bref, passons, Linux c'est le noyau ; le moteur.

- Un programme d'installation convivial si possible (celui de Debian n'est pas trop mal – largement mieux que le début d'une installation Windows XP… ; bien qu'en mode texte, il y a des « fenêtres » et votre matériel est reconnu à 99%, pas de « disquette F6 » foireuse).

- Un système de « paquets » intelligent qui permet de gérer vos applications installées (ou à installer) de manière agréable et performante.

- L'assurance d'avoir des paquets en abondance et à jour, permettant d'avoir les dernières applications dans tous les domaines. Debian le permet, malgré sa réputation d'être « lent à sortir des versions » (vous verrez plus bas).

Par rapport à du Windows, il faut voir que toutes les applications, de la calculatrice en mode texte à la dernière suite bureautique en passant par le compilateur C, les bases de données ou encore les jeux, sont « packagés » pour votre distribution. Si un programme ne devait pas l'être, vous avez toujours la possibilité de l'installer à la main, mais ça peut devenir compliqué pour un néophyte et vous perdez peu à peu la cohérence sur votre système.

Exemple : installer un serveur TeamSpeak. Ce n'est pas libre, il n'y aura probablement jamais de paquet Debian. Mais ce cas est simple : vous avez un installeur Linux. Parfois, vous n'aurez que la compilation pour vous sauver (make, make install, make it_work_please etc).

Bref, toutes les distributions Linux, notamment Debian, proposent un environnement *complet* et *cohérent*. Vous n'avez pas à vous soucier de grand chose pour installer par exemple OpenOffice (si vraiment vous voulez faire une Debian graphique ☺, vous demandez l'installation du paquet et Debian ira le télécharger pour vous, l'installer, et vous proposera même des options : installation de langues supplémentaires, dictionnaires etc.

Ensuite, lorsque vous voudrez mettre à jour votre système pour rester à la page et profiter de nouvelles versions de logiciels, Debian le fera pour vous, un peu comme Windows Update chez Microsoft, sauf que là, c'est l'intégralité des applications « packagées » qui sera comparée et mise à jour, que ce soit pour raisons de sécurité (bugs, failles etc) ou car une nouvelle version est sortie. C'est un « WindowsUpdate » qui gère non seulement le cœur du système, mais aussi la dernière application, si misérable soit-elle.

Debian a percé dans la masse des distributions basées sur Linux grâce à son système de gestion de « paquets » justement, qui était vraiment novateur. Actuellement, les distributions font un peu toutes

les mêmes choses et vous pouvez installer des paquets Redhat/Mandriva sur une Debian ou l'inverse. M'enfin, pour l'instant, on s'en moque pas mal. Et de toute manière, c'est toujours un peu bancal de faire ça. Toujours est il que Debian, j'adore, alors je vous explique :)

Je nuance juste un peu mes propos concernant certaines « applications web type LAMP » (le blog Wordpress par exemple). Ceux-ci évoluent vraiment vite et chaque nouvelle version est un régal. Ces outils ne nécessitent en général aucun outil connexe (sauf base de données, serveur web etc). Du coup, personnellement, je préfère maintenant utiliser les versions CVS / SVN pour être toujours à jour. Même les mises à jour de sécurité sont un peu lentes à venir ; quand les packageurs Debian n'insèrent pas des bugs en voulant bien faire…

1.5 Notions spécifiques à Debian sur les distributions

Debian est une distribution non commerciale et qui devrait le rester, c'est écrit dans ses statuts, il n'y a donc pas d'entreprise derrière son évolution. Malgré ça, elle reste une des plus connues et utilisées sur le « marché ». Elle est donc sérieuse, complètement libre et le restera.

Il faut savoir qu'il y a toujours 3 versions de Debian en parallèle :

- La version « stable » (depuis début 2007 nommée « Etch », c'est la version 4 de Debian). C'est la version finalisée qui n'évolue plus dans le temps, sauf pour des correctifs majeurs (failles, sécurité etc). Chez Debian, il faut compter une version majeure tous les 2/3 ans. Aucun planning forcé. Normalement, la « stable » est la version qu'on **installe sur une machine productive**, qui n'évolue pas ou peu, sauf pour raisons de sécurité.

- La version « testing » (actuellement nommée « Lenny », version 4.1 de Debian). Elle a l'avantage d'être quasiment au top des sorties logicielles, tout en étant stable. Vous aurez donc quasiment le dernier environnement graphique, la dernière suite bureautique, le presque dernier serveur Apache, le presque dernier noyau Linux etc. Idéale pour un serveur pas trop critique sur lequel vous voulez avoir des logiciels encore dans le coup ☺ Petite remarque : ce n'est pas parce-qu'il y a écrit « testing » que vous allez récupérer un système foireux non testé : la politique de sortie de versions et de validations des paquets est draconienne chez Debian (lire plus bas), et je vous garantis que vous n'allez pas beta-tester grand chose avec la « testing ».

- La version « unstable », toujours nommée « Sid ». Là pour le coup, vous bénéficiez des paquets dès leur sortie. C'est bien, mais vous allez vite passer votre temps à installer des trucs à longueur de temps et quelques fois, oui, y'a des bugs, voire des cauchemars à la clef... « Unstable » veut dire qu'aucune version d'appli n'est stable (dans le sens « figée »), pas que les applis ne sont pas stables (ie, « plantent »). A l'issue de quelques tests, les paquets « unstable » arrivent dans la « testing ».

La politique de sortie des versions est bien connue des adeptes : « when it's done ». Donc pas de version obligatoire tous les 6 mois (la politique chez « Ubuntu »), uniquement lorsque les dirigeants de Debian l'ont décidé. Ces dirigeants sont élus chaque année démocratiquement, sur le web, ils présentent leur programme etc... Un vrai petit monde à part.

Les jolis noms Etch, Lenny et ceux à venir sont tirés du dessin animé Toy Story. Monsieur Patate y est déjà passé, Sarge, Woody, Buzz l'Eclair et le cochon, mais bon, il en reste.

Comment ça se passe lors d'un décalage de version ? La « stable » se retrouve dans les placards, la testing devient la stable et est figée, un nouveau nom est donné à la « testing » qui va continuer d'évoluer et la « unstable » reste la unstable, toujours appelée « Sid ». A ce moment, vous « upgradez » votre système ou non.

Sid, c'est le nom du gamin complètement taré et hyperactif dans Toy Story, ce n'est pas pour rien, y'a pas de raison qu'il se calme ;) Evidemment, la testing reçoit ses paquets lorsqu'ils sont validés par les gens qui utilisent la « unstable ».

1.6 Matériel requis

Pour essayer : n'importe quel PC, même un peu pourri. Pensez à essayer grâce à VirtualBox (un VMWare gratuit à peine moins fonctionnel, amplement suffisant dans notre cas).

Notez que l'installeur Debian est beaucoup moins moisi que l'installeur de Windows, par exemple. Ainsi, vos derniers matériels SATA, SCSI, carte vidéo NVidia, ATI etc sont reconnues de base. Dans le cas d'un serveur, pensez tout de même à vérifier que le contrôleur RAID que vous souhaitez peut-être utiliser est bien compatible ; sur le site du fabricant.

Enfin, côté puissance (nombre de processeurs/cœurs, rapidité, quantité de mémoire), ce n'est pas tellement une réflexion spécifique à Debian. On ne compare pas une machine de « Grid Computing » avec un serveur de fichiers, de mails ou un simple serveur web à quelques utilisateurs par jour. A vous de voir ☺

1.7 Avant de démarrer

1.7.1 Quel CD d'installation ?

Tout est là : http://www.debian.org/CD/. Notez qu'il y a :

- des jeux de CD pour la « stable », pour la « testing » (Le premier CD suffit à lui seul, sauf si vous n'avez pas de connexion web sur le lieu de l'installation ; vous n'aurez alors pas la quantité impressionnante de logiciels à disposition sur un seul CD, mais sur 6 ou 7). En général, ils sont inutiles.
- des CD pour installation par le web (« netinst »), c'est-à-dire un CD avec le minimum vital, le reste sera téléchargé après coup.
- et encore plein d'autres variantes (architectures matérielles différentes : SPARC, IA64…

Si vous êtes perdu, voici quelques exemples courants :

- Les CD de « stable », exemple pour « netinst stable sur architecture x86 ».
- Les CD de « testing », exemple pour « netinst testing sur architecture x86 ».

1.7.2 Installation par le réseau

Qui dit version à installer par le web, dit connexion Internet opérationnelle. Cette méthode a donc ses limites si vous installez une Debian chez vous et que vous utilisez un modem ou une carte réseau non supporté en standard (ce qui a tendance à se raréfier je pense). En général, vous avez donc à vérifier votre modem d'une part et votre carte réseau d'autre part.

Pour le modem, les cas simples sont l'utilisation d'un modem/routeur/firewall/wifi/dhcp/etc (genre un boîtier Netgear branché directement sur votre prise téléphonique) ou un machinBox (freebox, neufBox, cestleBox etc). Ce sont ces matériels qui font la connexion web, et votre machine derrière est sur un réseau local. Dans ce cas, vous n'aurez donc qu'à vous assurer que votre carte réseau est supportée.

Pour la carte réseau, sachez que les cartes réseaux 3com, Netgear, Broadcom (souvent les cartes intégrées aux cartes mères sont des Broadcom), Marvell etc sont supportées par les noyaux Linux récents. Pour les autres, Google vous renseignera.

Si vous avez déjà un Windows sur le PC où vous comptez installer Debian, profitez-en avant d'installer pour vérifier la référence précise de votre carte dans le gestionnaire de périphériques. De même pour la carte vidéo, si vous avez une ATI ou une NVidia, no problem. Idem pour les chipsets Intel intégrés. Si vous avez un vieux clou, pas de problème en général. Enfin, si vous avez un truc complètement inconnu, euh.... Google vous renseignera.

Parmi les autres pré-requis, il vous faut un espace disque libre d'environ 5 ou 6 Go s'il s'agit de tester le système en installant un peu tout ce qui passe « pour voir ». Je reparle du partitionnement plus tard dans la procédure d'installation.

1.7.3 Temps d'installation

Il faut compter 30 minutes, 1 heure au max, pour avoir un système installé, opérationnel, mais presque vide :) Ensuite, le temps de télécharger 2 ou 300 Mo pour installer un environnement complet suivant vos besoins. Puis toute une vie pour peaufiner, suivant votre niveau de geekattitude.

1.7.4 Imprévus à l'installation

Suivant le matériel détecté sur votre machine, certaines étapes supplémentaires peuvent avoir lieu. Exemple, si vous avez des connexions firewire ou plusieurs cartes réseaux, le programme d'installation vous demandera laquelle utiliser. Vous y survivrez, ce n'est pas très compliqué.

En « testing », vu que le « Debian Installer » peut évoluer, vous pourrez aussi voir des différences avec les photos d'écran que je montre. A priori, ce ne sera pas sur des éléments décisifs. Le programme d'installation ne vous demandera pas de trucs ultra-balaises ou alors il vous aiguillera vers une réponse toute faite. Les premières phases de l'installation ne font (modifient) rien sur votre disque dur, vous pouvez rebooter si vous pensez avoir tapé une bêtise. C'est après le partitionnement des disques que ça commence vraiment.

1.7.5 Exemples de code, de configuration etc

Remarque : je donne dans la documentation des exemples de lignes de commande ou de fichiers dans des passages encadrés. Il y a souvent des commentaires en fin de ligne avec des dièses. Si vous

procédez par copier-coller, je vous recommande d'enlever les commentaires car ils pourraient générer une erreur (du fait d'apostrophes dans les commentaires ou ce genre de choses)

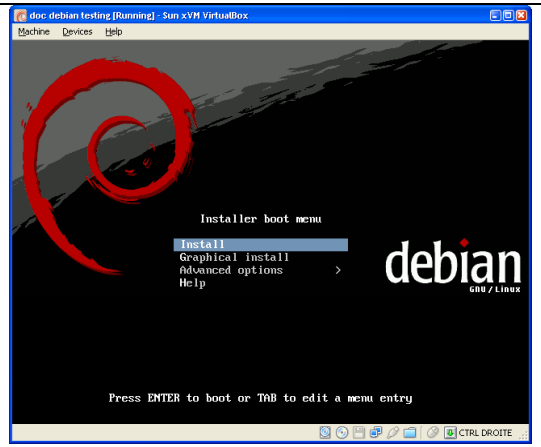
Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)

2 Installation pas à pas

Remarque générale : pour vous déplacer dans l'interface « graphique en mode texte » ;) utilisez les flèches, la touche entrée et la touche « TABulation » pour sauter de bouton en bouton si les flèches ne le permettent pas - ça arrive je crois.

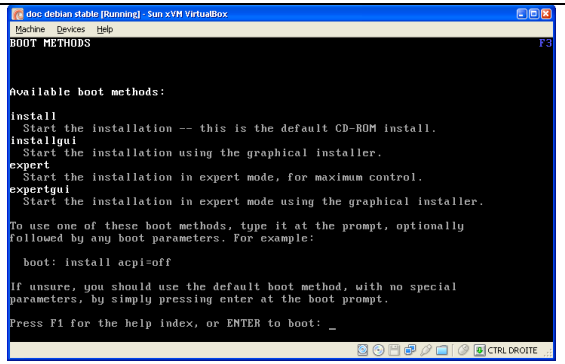
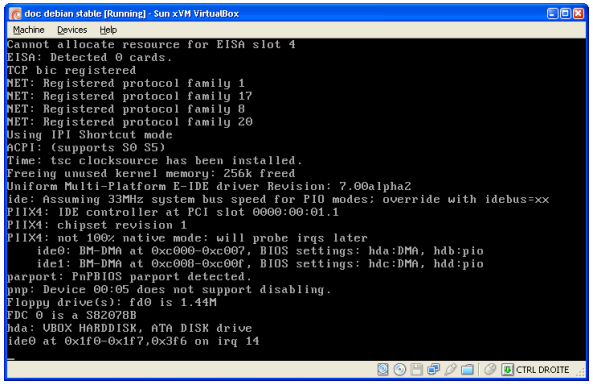
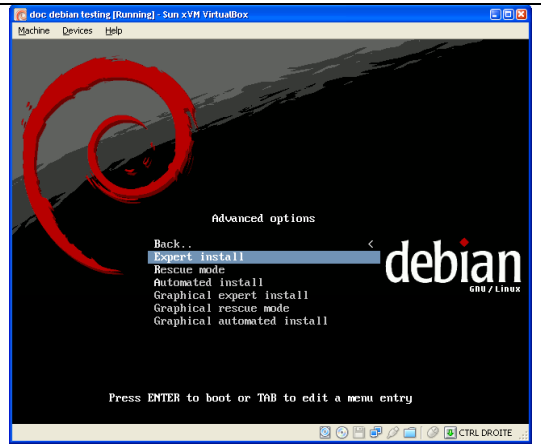
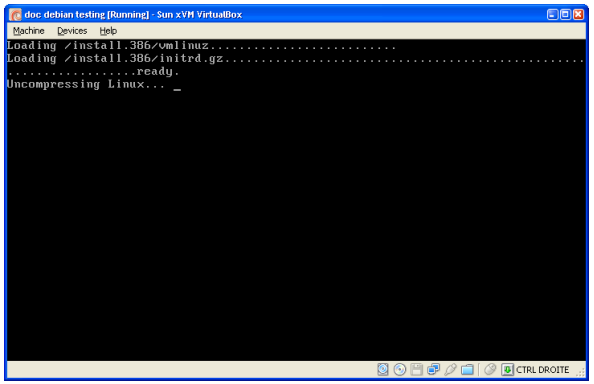
2.1 Démarrage sur le CD, choix du mode « expert » ou « normal »

Pour passer en mode « expert » sur « Etch », il faut taper « expert » puis valider (ou lire la doc avec les touches F1, F2, F3 etc) :



Sous « Lenny », c'est vaguement graphique :

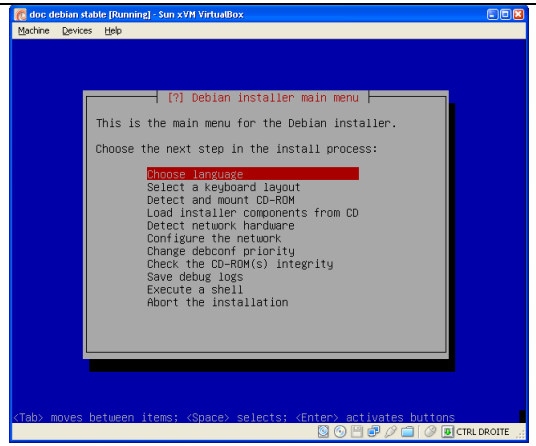
Ici, on prend le mode normal pour comparer, il suffit de valider

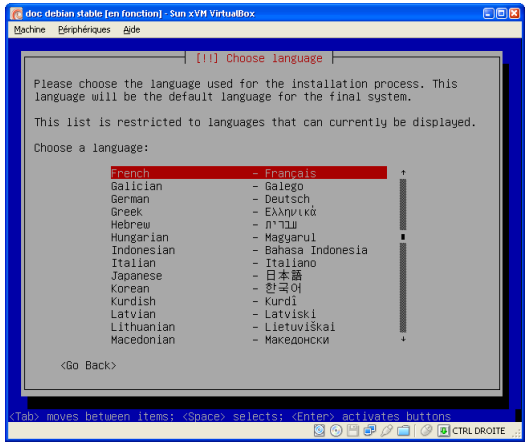
Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)
 Chargement du noyau :  L'installation démarre pour de bon.	 Chargement du noyau :  L'installation démarre pour de bon.	Idem (photo peu intéressante).

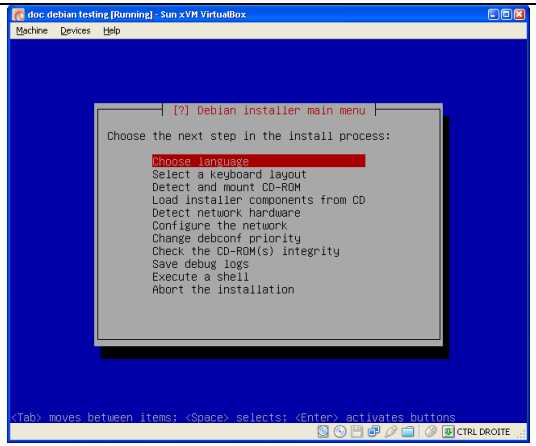
L'écran précédent (chargement du noyau) est une constante lorsqu'un Linux démarre (sauf des distributions un peu plus user-friendly qui masque tout cela par une fenêtre avec un joli logo, genre Ubuntu). Avec un peu d'habitude, on apprend à lire uniquement les éventuels messages d'erreurs.

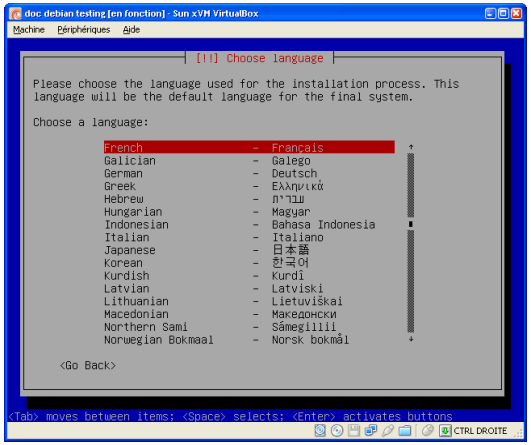
Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)

2.2 Choix de la langue, clavier

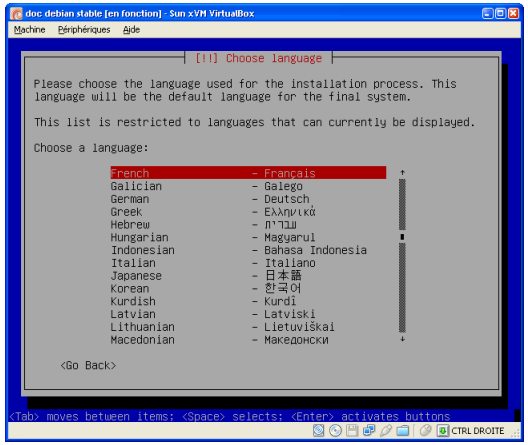


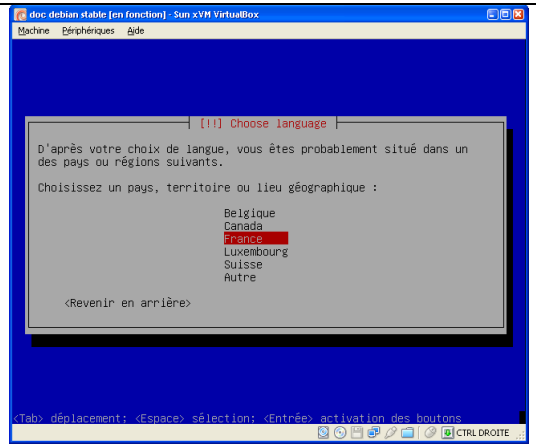
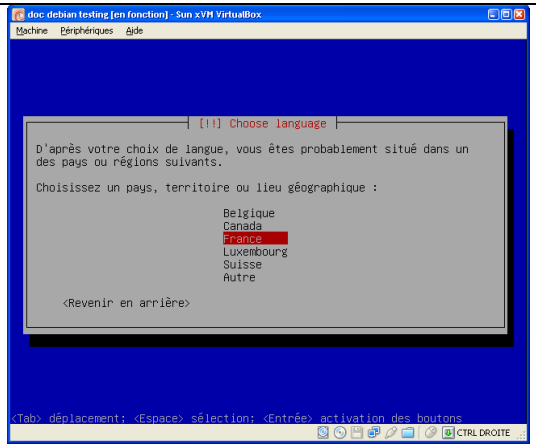
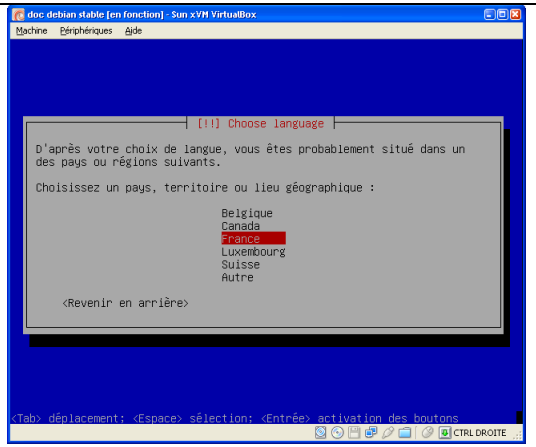
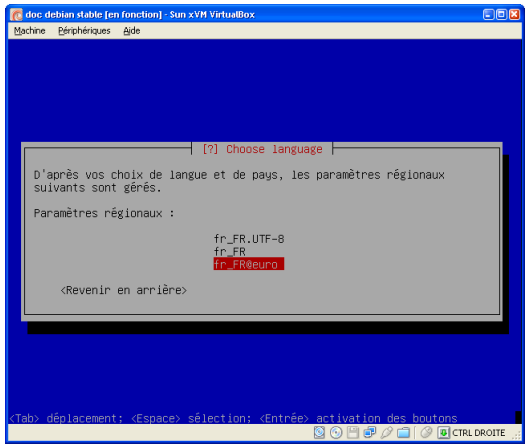
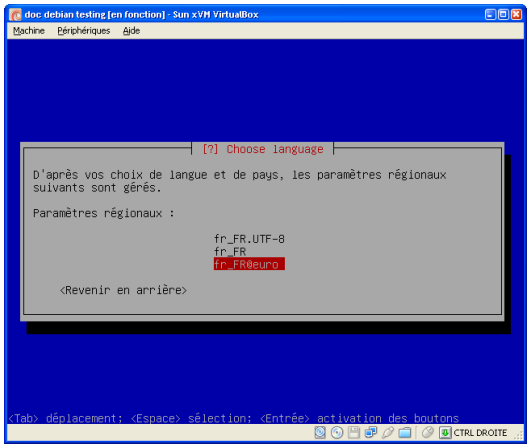


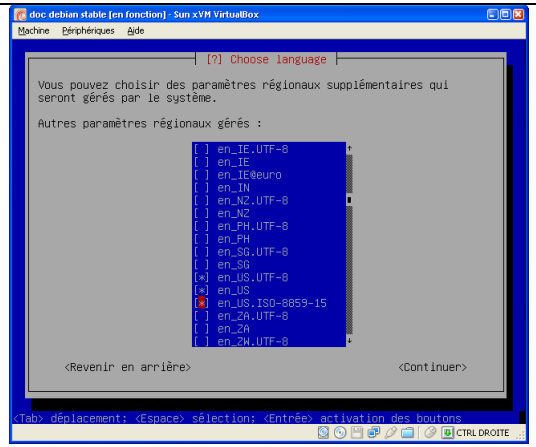
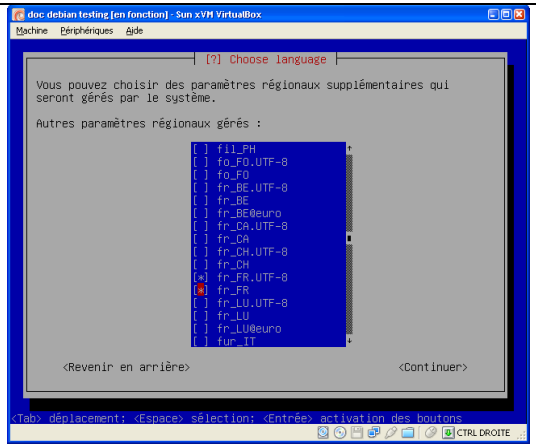
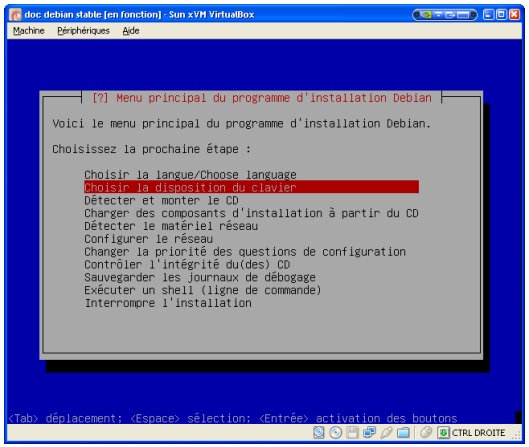
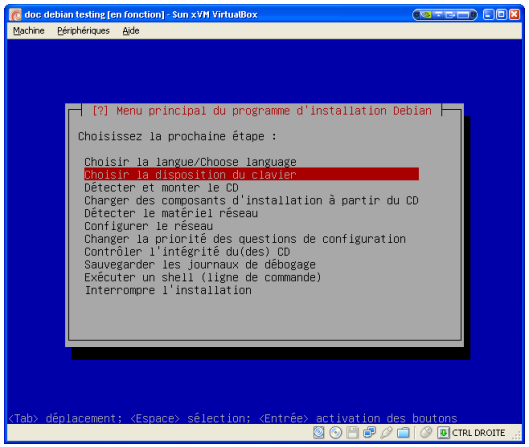


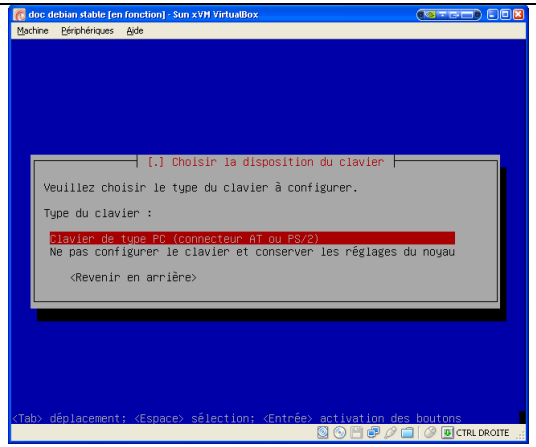
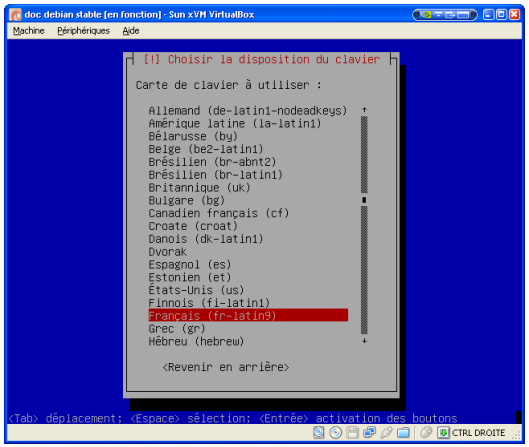
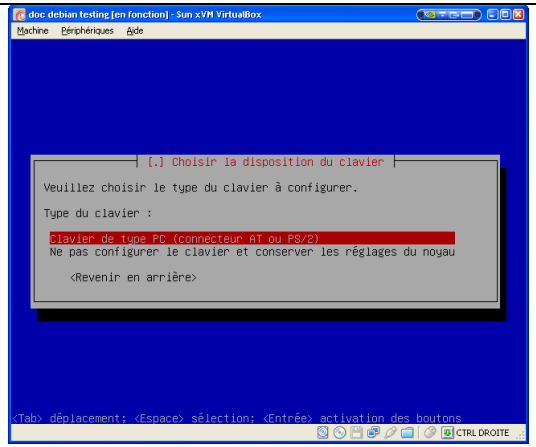
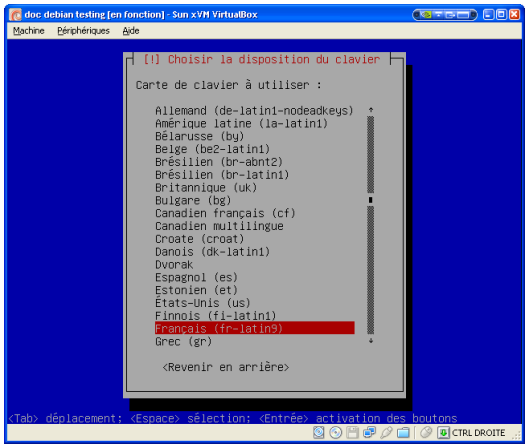
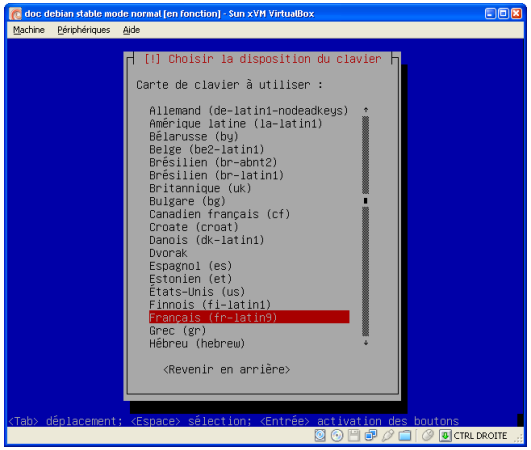


En mode « normal », on ne voit pas le tableau ci-contre qui rappelle chaque étape de la configuration. Les étapes ne font que s'enchaîner.



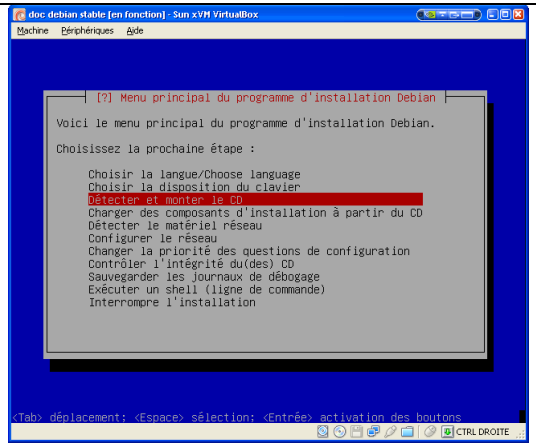
Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)
		
		Dans ce mode, c'est une version allégée, on choisit simplement la langue, pas l'encodage. Ca peut se choisir plus tard en reconfigurant le paquet « locales » je pense (via la commande « dpkg-reconfigure locales »). La seule différence rapidement notable est que de base, les messages systèmes seront en français là où un puriste les voudra en anglais (notamment pour mieux débugger sous Google).

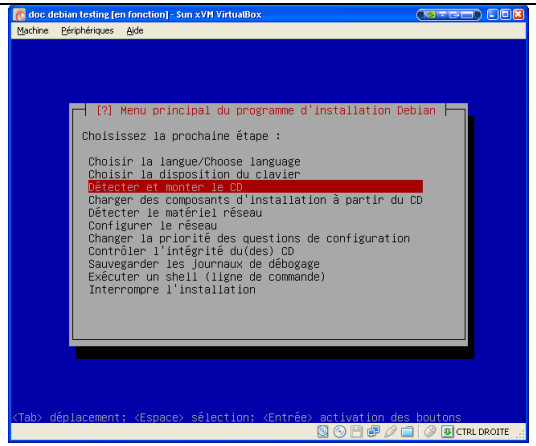
Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)
		
		

Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)
 	 	

Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)

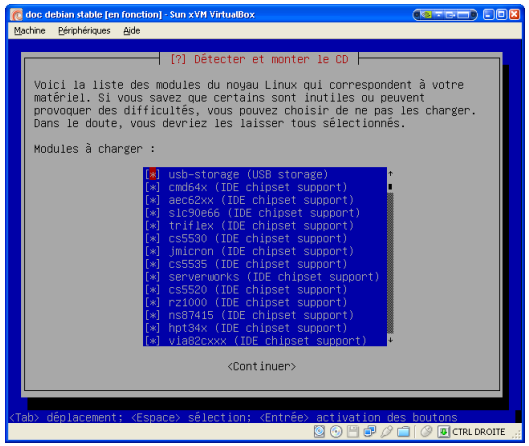
2.3 Détection du CDROM, chargement des paquets de base

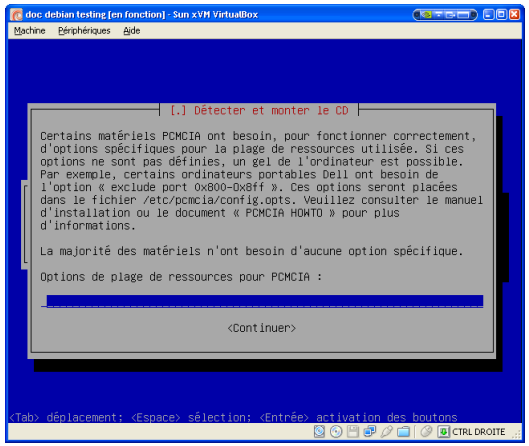
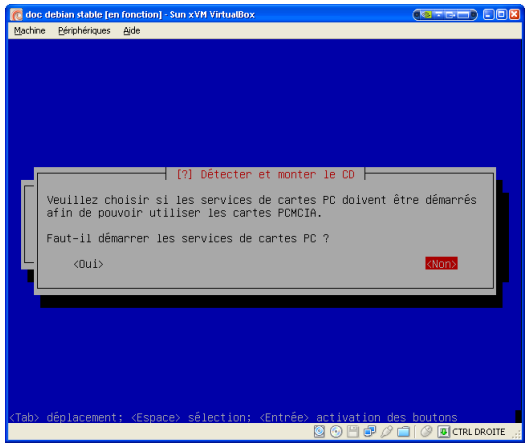
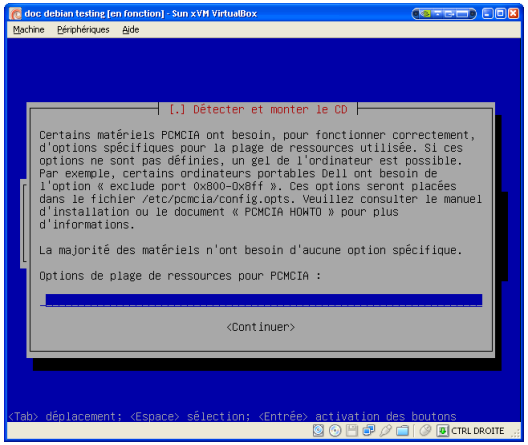
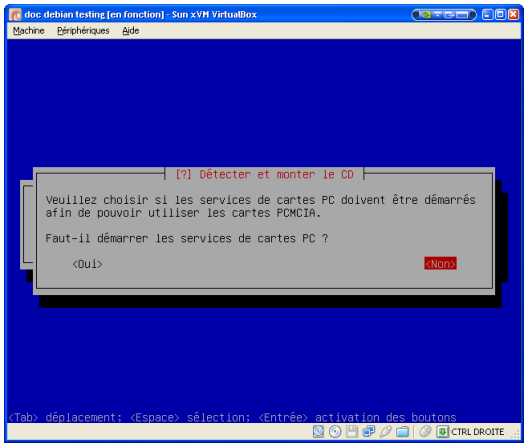
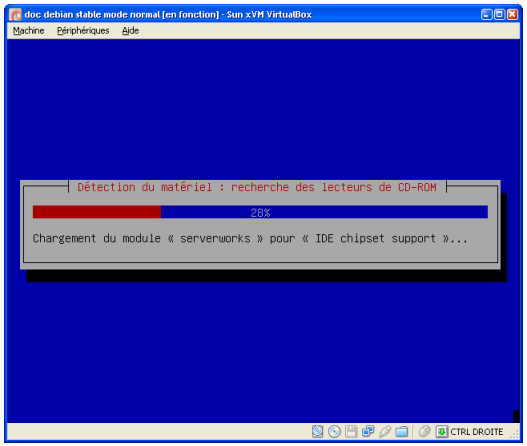


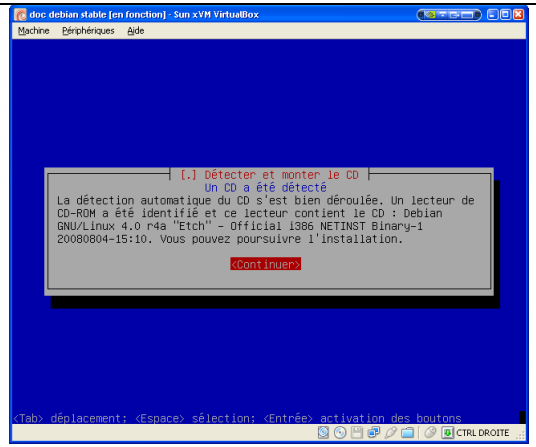
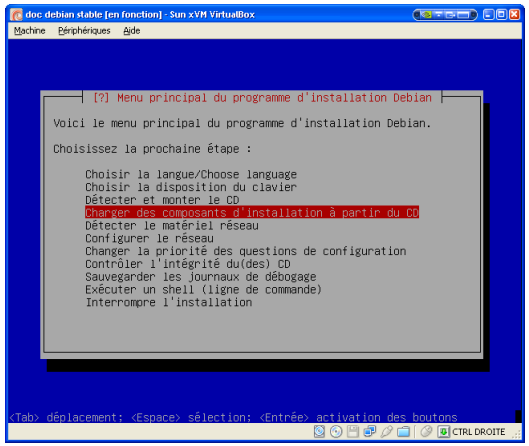
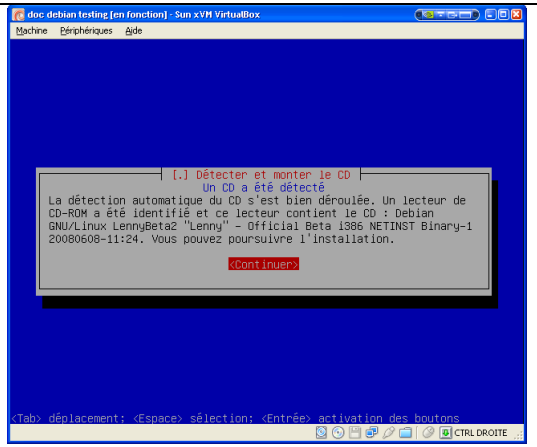
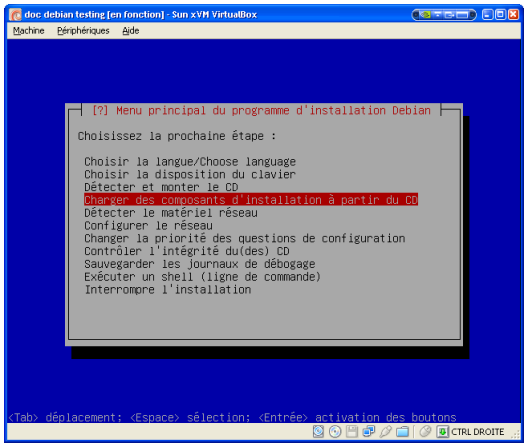


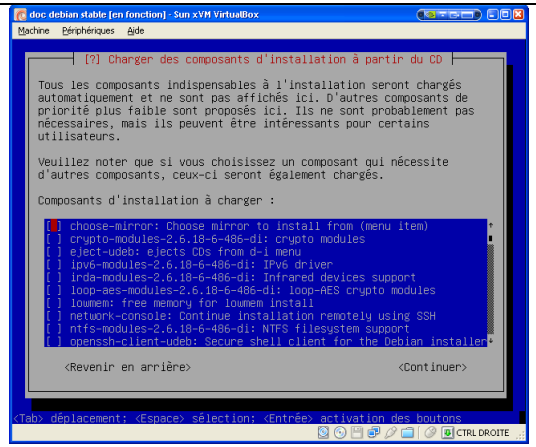
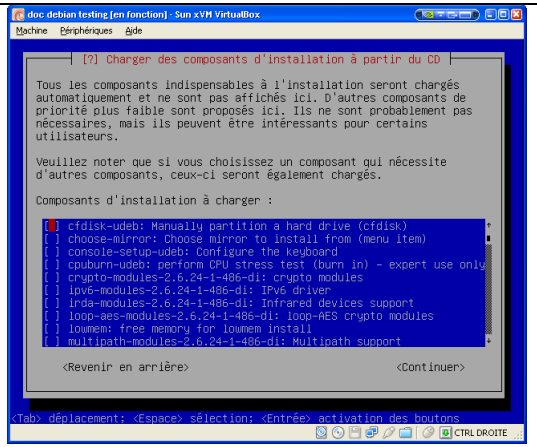
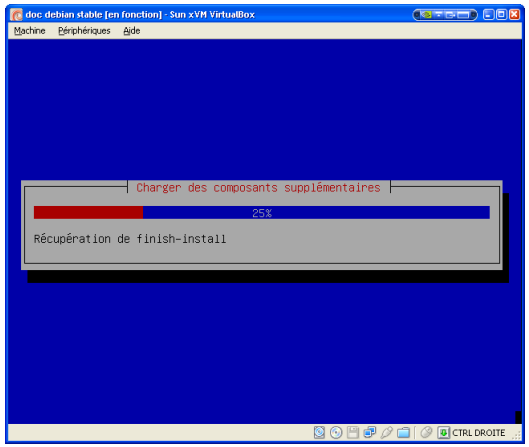
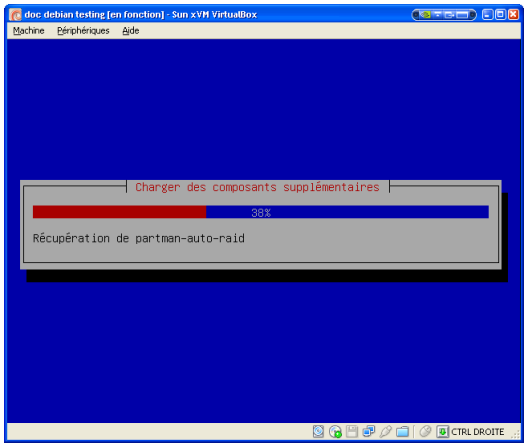
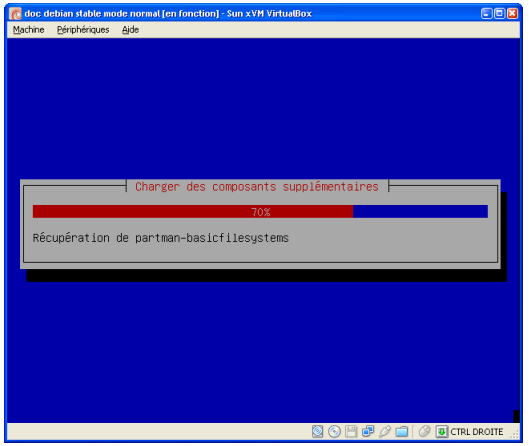
En mode « normal », quelques questions sont zappées et on arrive directement plus bas dans l'installation.

Chargement optionnel de modules pour cette détection, utile si votre lecteur n'est pas reconnu – je doute :



Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)
Ici, on peut utiliser une carte PCMCIA (je doute sur un serveur, c'est pas un portable – et des lecteurs de CD par interface PCMCIA, j'en ai pas vu beaucoup).   	Idem.   	Sur tous les types d'installation, on voit passer le chargement des modules de CDROM : 

Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)
  A priori vous n'aurez besoin de rien :	  Idem ci-dessous :	

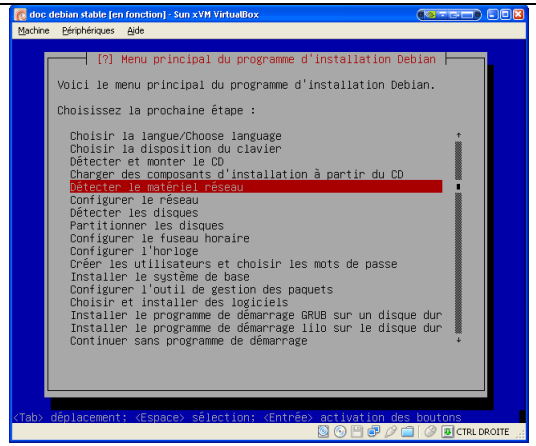
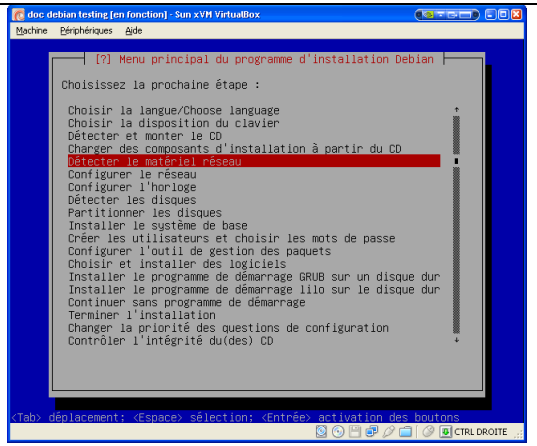
Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)
		Idem
Enfin, on voit passer le chargement de tous les programmes de base pour l'installation : 	Idem 	

Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)

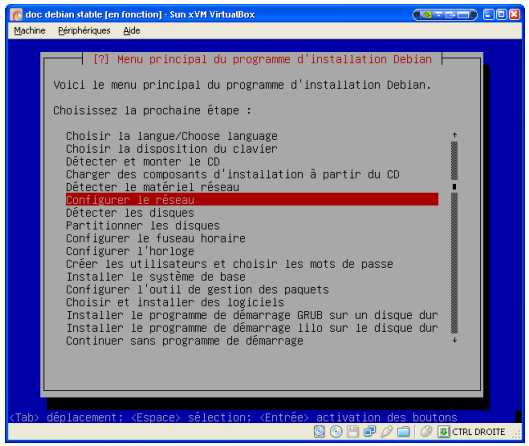
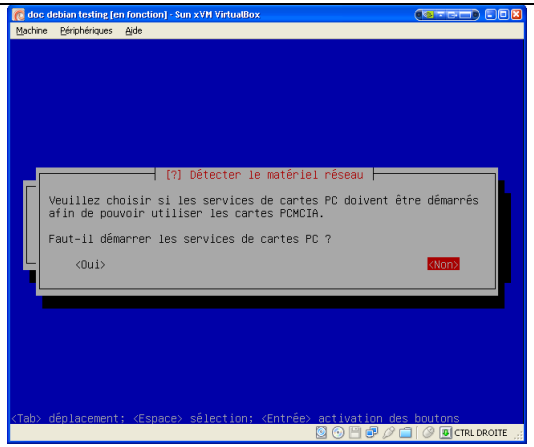
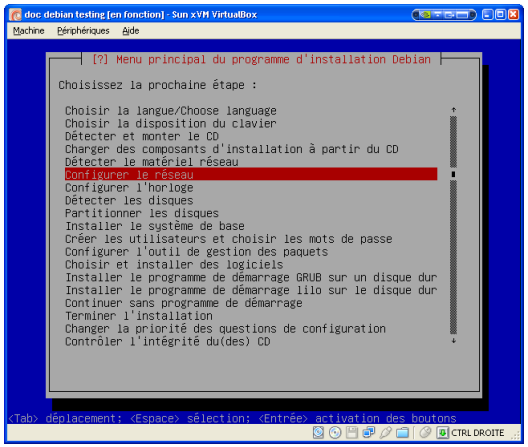
2.4 Configuration du réseau

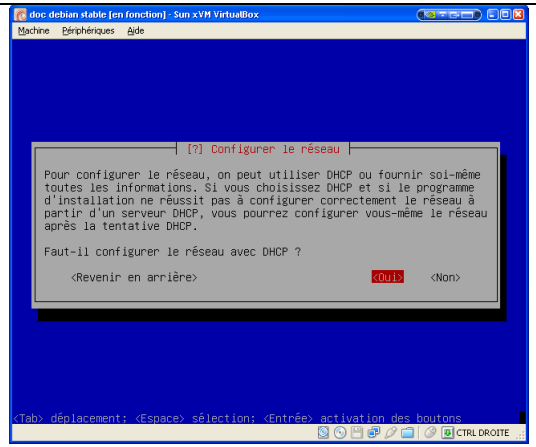
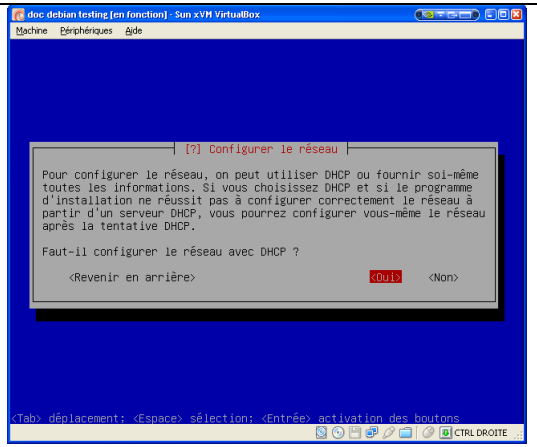
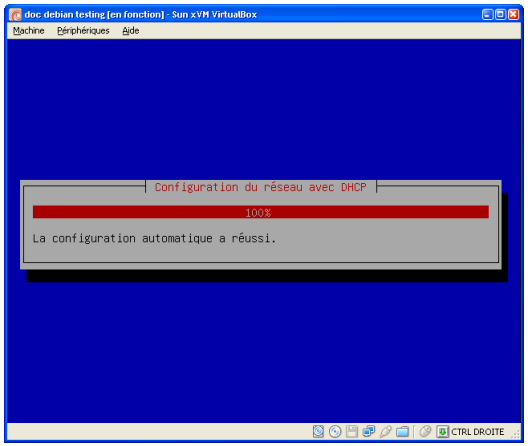
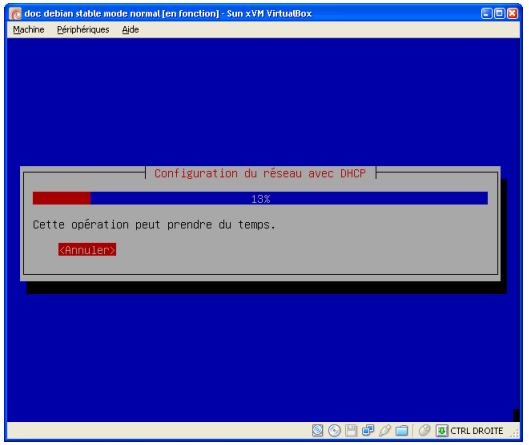
Le cas présenté est un cas simple : une seule carte réseau, un serveur DHCP dans un coin (ça peut être une Freebox ou ce type de matériel)

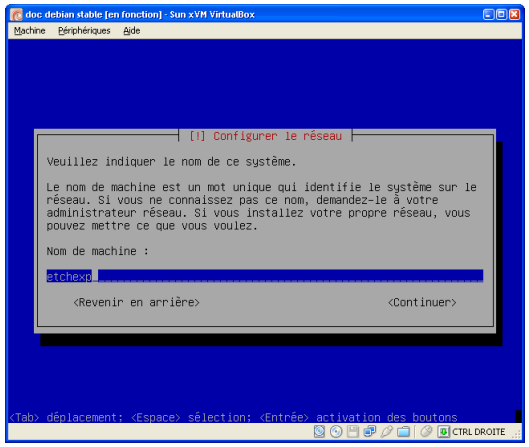
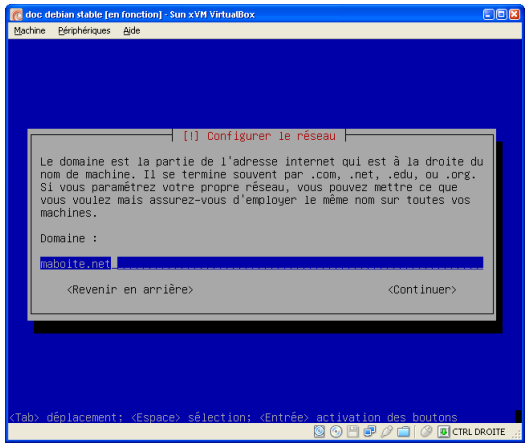
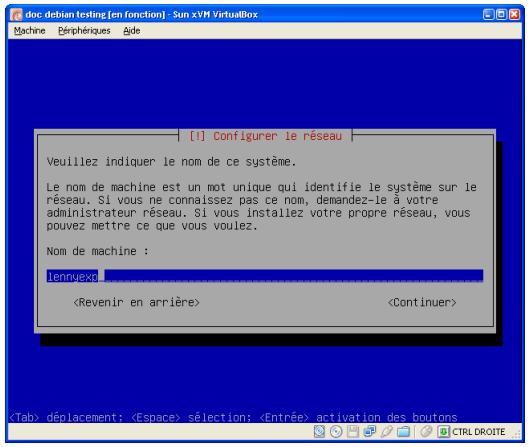
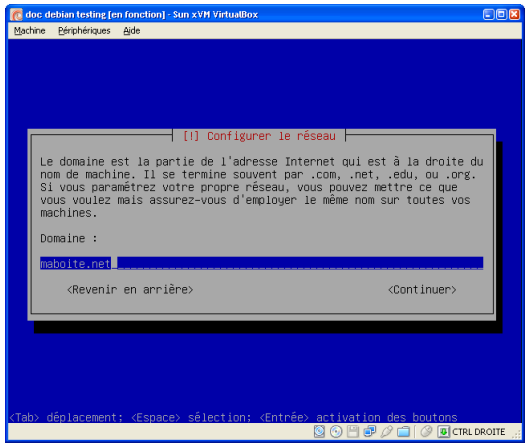
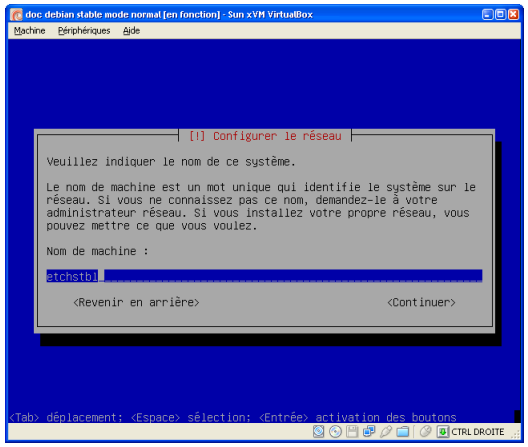
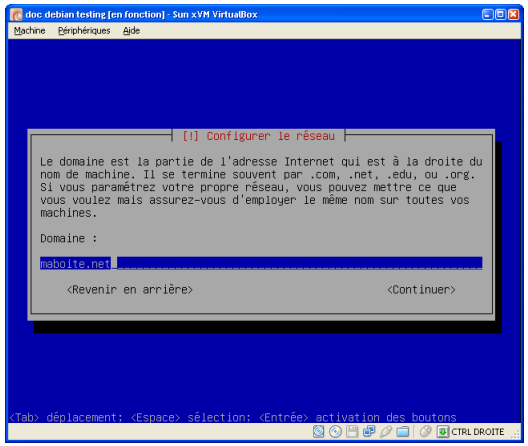
Si vous avez plusieurs cartes réseaux, le programme d'installation vous demandera donc de configurer l'une d'entre elle (idéalement celle connectée au web par un biais quelconque afin de pouvoir récupérer des informations sur Internet). Ce n'est pas super compliqué, je zappe. Voici donc la configuration simple (une carte reconnue, un serveur DHCP accessible).

<table>
<tr>
<td> 
</td>
<td> 

La même histoire pour les cartes PCMCIA. Ce ne sera utile qu'avec une carte réseau sur PCMCIA :</td>
<td>Ici, le programme d'installation tente directement sa chance en DHCP. Voir plus bas.</td>
</tr>
</table>

Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)
	 	

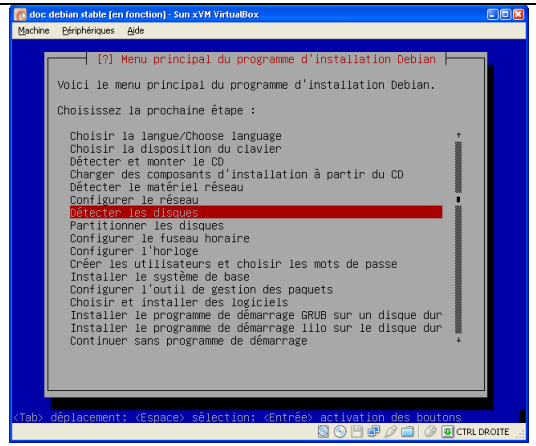
Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)
 Même chose que ci-contre.	 	

Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)
On donne un nom à la machine :  Et un domaine, bidon ou non – à vous de voir suivant votre configuration réseau.  	On donne un nom à la machine :  Et un domaine, bidon ou non – à vous de voir suivant votre configuration réseau.  	On donne un nom à la machine :  Et un domaine, bidon ou non – à vous de voir suivant votre configuration réseau. 

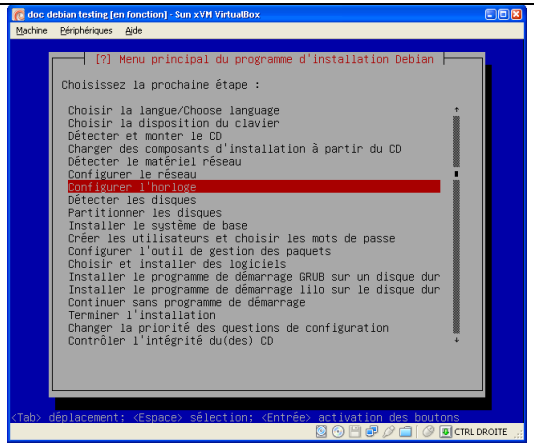
Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)

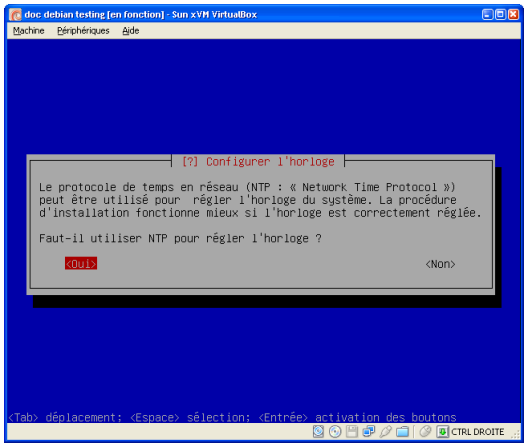
2.5 Configuration de l'horloge / fuseau horaire

En « testing » (« Lenny »), la configuration de l'horloge passe avant la configuration des disques durs. La voici donc décrite, avant la partie configuration des disques durs.



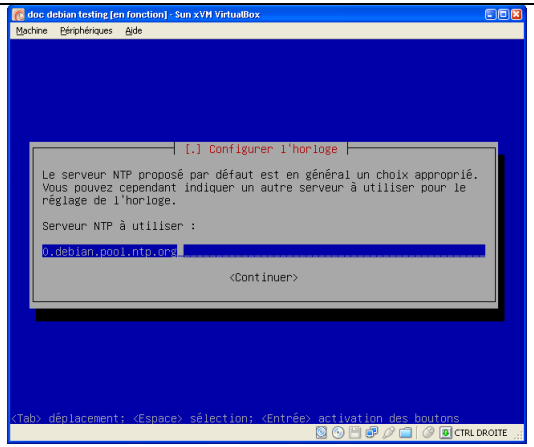
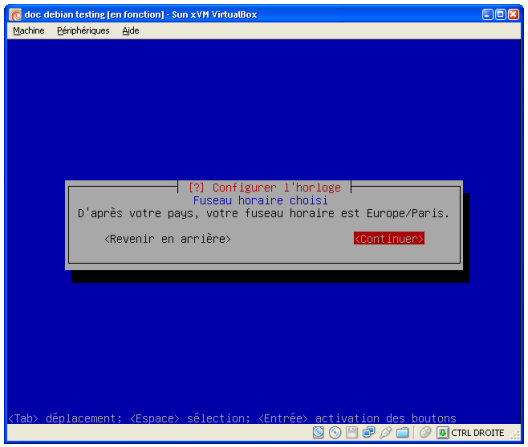
Rendez-vous donc au chapitre « 2.6 Configuration des disques durs pour l'installation ».





J'utilise ici un « pool » standard de serveurs NTP :

Rendez-vous donc au chapitre « 2.6 Configuration des disques durs pour l'installation ».

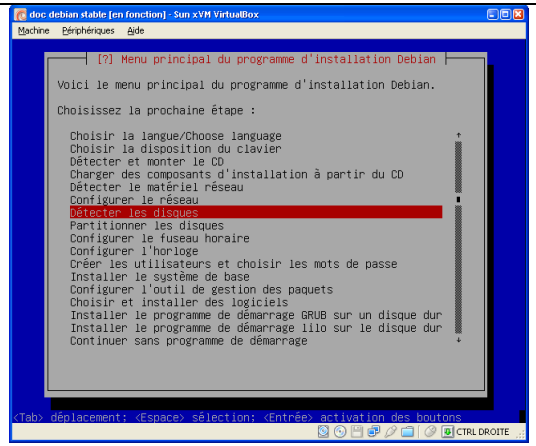
Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)
	 	

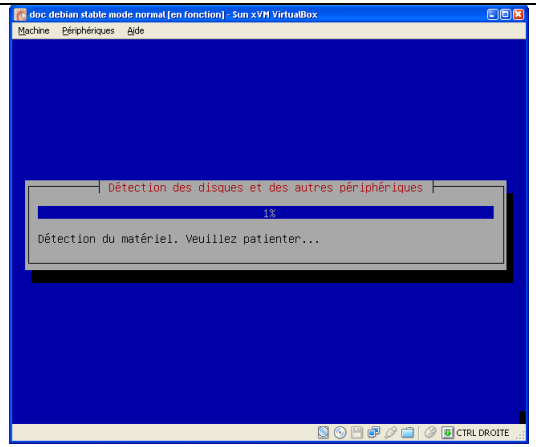
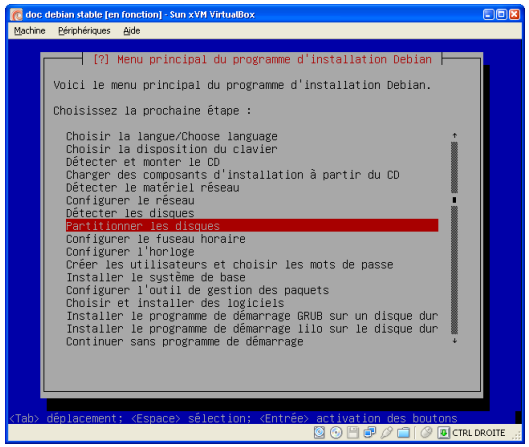
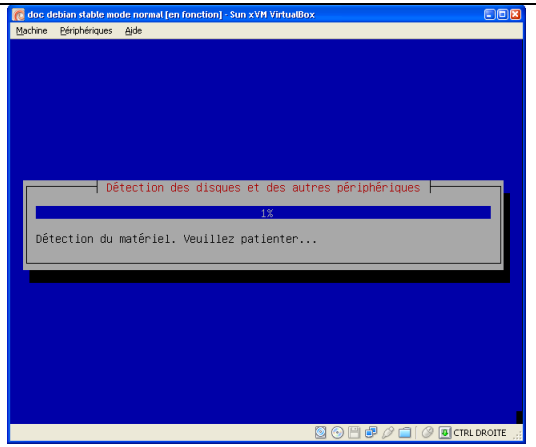
Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)

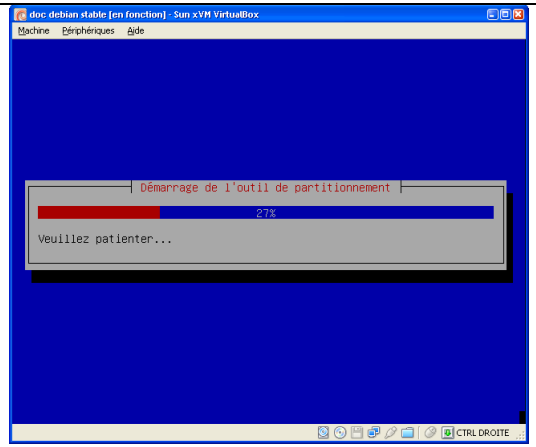
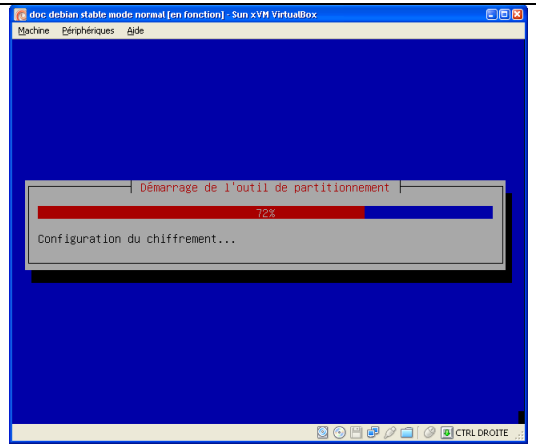
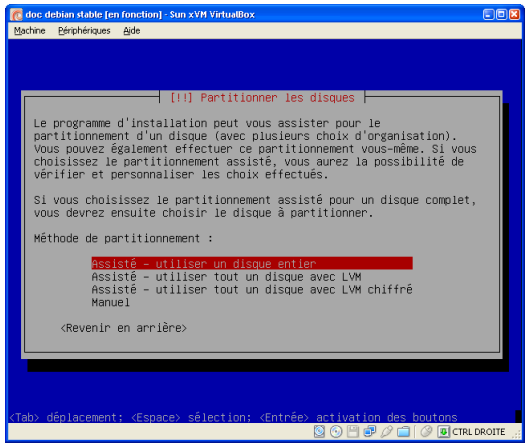
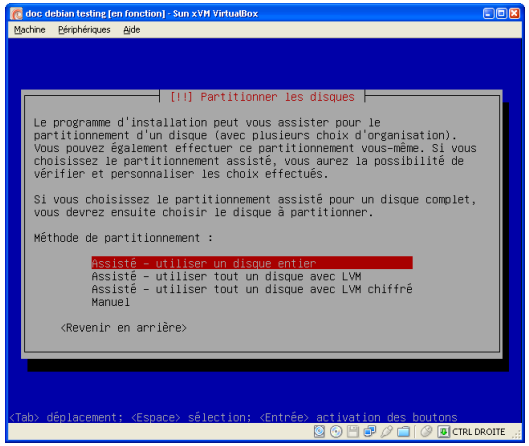
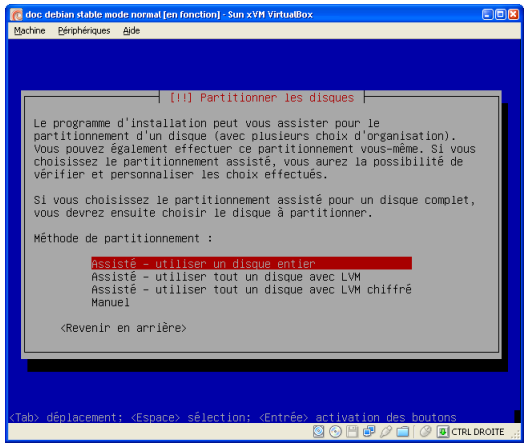
2.6 Configuration des disques durs pour l'installation

Rappel : en « Etch », cette étape arrive avant la configuration de l'horloge. En Lenny, c'est après.

2.6.1 Lancement

	Je zappe un peu les photos d'écran, c'est à peu près pareil qu'en « Etch » mode expert, voir donc ci-contre à gauche.	Encore une fois, en « normal », tout un tas d'étapes sont passées sous silence. Rendez-vous plus bas.

Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)
 		

Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)
		
		

2.6.2 Principes généraux

Je vais décrire 2 choix uniquement parmi les 3 (4 en fait) : mode assisté et mode manuel. Je ne parlerai pas de LVM (« Logical Volume Manager »). LVM aura l'avantage de vous laisser de la marge de manœuvre pour redimensionner à la volée vos partitions. Personnellement, j'en ai jamais eu besoin, du moment qu'on sait à quoi va servir le serveur et qu'on le configure à l'avance correctement.

Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)

Avant les photos d'écran, petits rappels sur les points de montage, partitions etc dans un environnement Linux. La théorie pour une machine bien propre et bien configurée est la suivante :

- Séparer par partitions les points de montage suivant : /var, /tmp, /home, /usr voire même /boot. Et évidemment le swap qui est une partition à part (mais pas « montée »).
- Affecter des tailles raisonnables suivant l'utilisation de votre machine. Du coup, vous pourrez avoir besoin d'affiner vos partitions. Exemple d'un serveur qui hébergera une base de données MySQL, vous pourriez affecter un espace particulier à /var/lib/mysql qui est l'emplacement par défaut des bases de MySQL.
- Si vous optez pour un serveur avec un partitionnement précis comme celui décrit à l'instant, voici les options intelligentes à positionner sur les points de montage. Certains peuvent parfois relever de la paranoïa type « si quelqu'un accède à /boot en root par une faille quelconque, je serais protégé car j'ai empêché l'exécution de programme SetUID dans /boot gniark gniark gniark ». Why not. Ca ne coûte pas grand-chose. A l'inverse, attention, j'avais une fois mis par erreur le « noexec » sur /usr. Le reboot ne se passe pas vraiment comme on voudrait ☺ Et j'ai dû réparer /etc/fstab avec un LiveOS sur clef USB.
    - /boot : noexec,nosuid,nodev => afin de limiter cet espace à sa stricte utilisation : le boot de l'OS. Prévoyez quelques centaines de MB, ce sera suffisant.
    - /home : noexec,nosuid,nodev => afin que les gens ne stockent ici que des données. Discutable suivant votre cas. Côté espace : à voir suivant vos besoins ? si tout le monde a des données sur réseau ou non, par exemple.
    - /tmp : nosuid,nodev => aucune raison de tolérer les scripts SetUID ou les créations de périphériques. N'enlevez pas le droit d'exécution (« noexec ») car beaucoup de programmes d'installations seront KO d'entrée de jeu. Quelques GB de données suffiront, à priori.
    - /var : nodev => exécution et SetUID nécessaires car pas mal d'environnement « chrootés » et de morceaux de système en SetUID. Vu que ce répertoire contiendra de base tous vos logs, espaces tampons en tous genres etc, n'hésitez pas à coller quelque chose comme 10 GB. Et potentiellement beaucoup plus si vous hébergerez une énorme base de données (à séparer en un point de montage spécifique à ce moment là).
    - Le « swap » : généralement, on dit 2 fois la RAM. Maintenant, vu la RAM qu'on peut avoir sur une machine, j'aurais plutôt tendance à ne mettre qu'une fois la RAM, quand on parle de machines en 2 ou 4 GB.

Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)

- Enfin, j'utilise des partitions de type « ext3 » ; l'intérêt des autres type me laisse rêveur.

Au fait, pourquoi s'embêter à cloisonner tout ça ? imaginez votre serveur de mails, un trafic anormal qui vous fait grossir un fichier de log à vitesse grand V. Si vous stockez tout au même endroit, une fois le disque plein, c'est l'OS qui est complètement en vrac. Si vous aviez séparé /var alors vos services font la gueule au moment du problème mais la machine est utilisable, le problème peut être localisé et traité en live. Et pas de crash violent et nécessité d'être présent physiquement pour rétablir la situation.

2.6.3 Mode assisté

Ce mode est assez sympathique. Il vous permet soit d'utiliser tout votre disque pour le système (sauf le « swap » évidemment), soit de prémacher des points de montage différents pour /var, /home, /tmp avec les options qui vont bien pour ces points de montage.

Il y aussi un mode « tout au même endroit », utile pour un serveur de test où on n'a pas envie de séparer les mondes ou dans une machine virtuelle pour faire joujou (VirtualBox ou VMWare par exemple).

2.6.4 Mode manuel

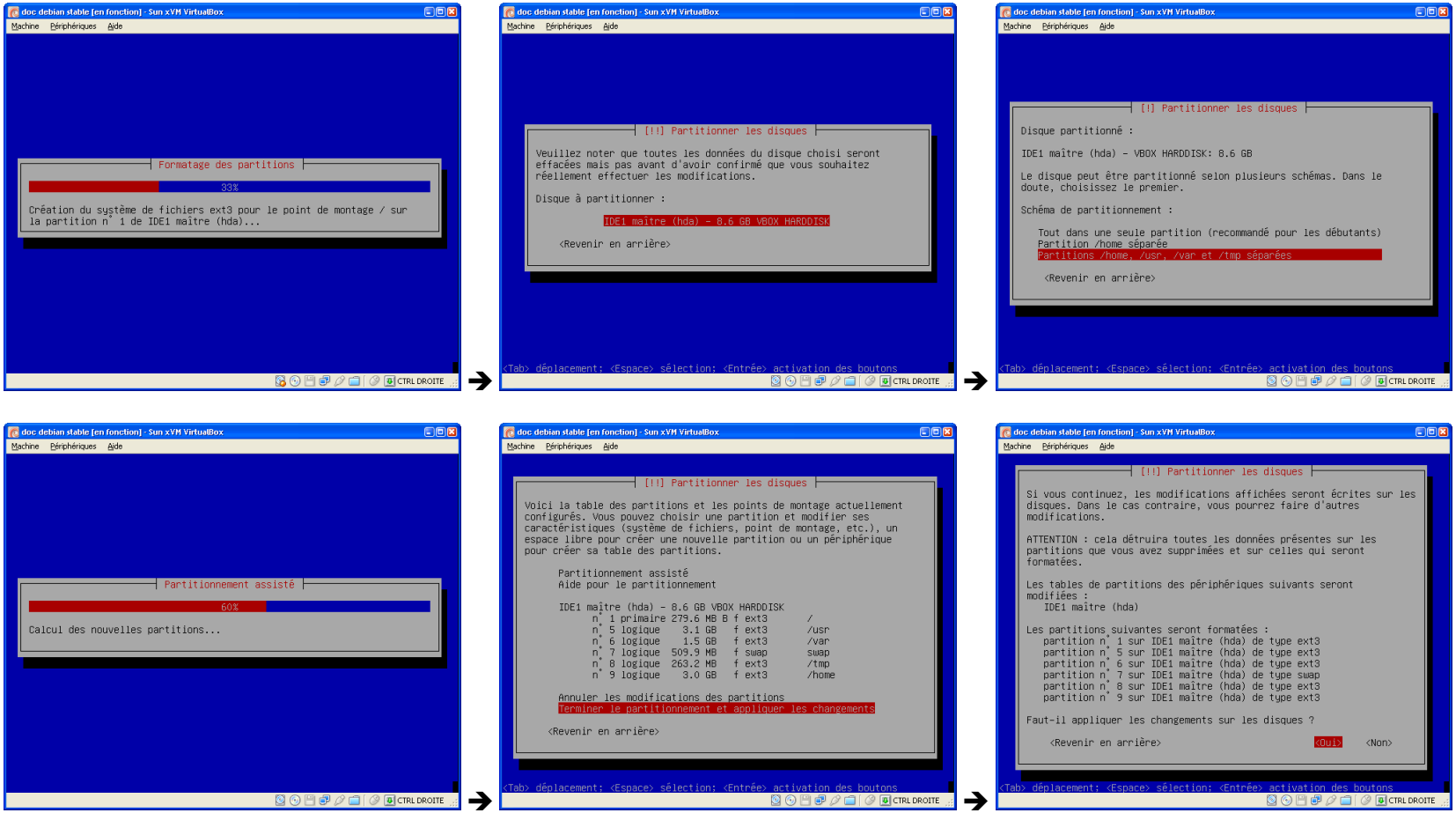
Si votre configuration est spécifique – exemple le cas cité ci-dessus où il faut séparer la future base de données énorme – alors vous en viendrez au mode complètement manuel.

2.6.5 En images : mode assisté, plusieurs points de montage

Ces images sont tirées du mode expert de Etch. Pas la peine de faire un comparatif entre les versions et les modes, tout se ressemble fortement.

Attention à la 3ème image, c'est le moment où l'on choisit le mode « tout au même endroit » ou « prédécoupage automatique ».

Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)



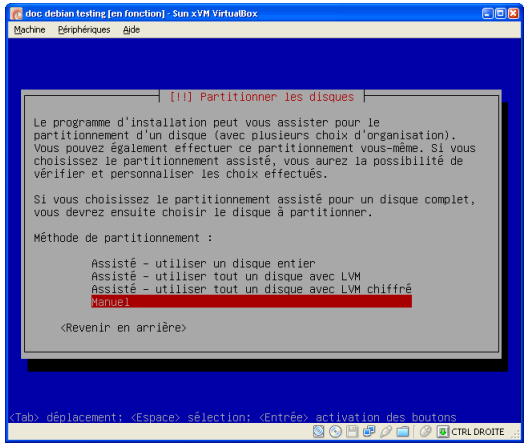
Les partitions créées, sur cet exemple bidon (VirtualBox de 8 GB, pas beaucoup de RAM), ont des tailles correctes en proportion de l'espace disponible.

2.6.6 En images : mode manuel, plusieurs points de montage

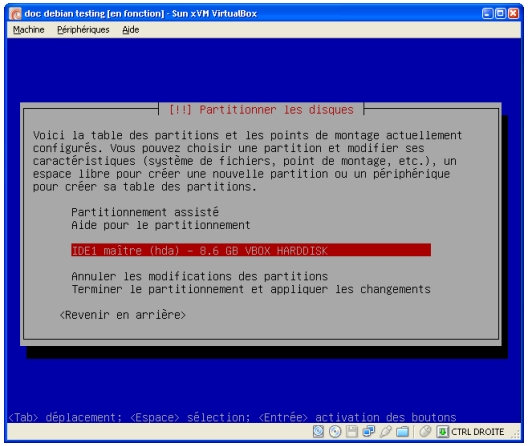
Ces images sont tirées du mode expert de Lenny. Pas la peine non plus de faire un comparatif entre les versions et les modes, tout se ressemble fortement.

Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)

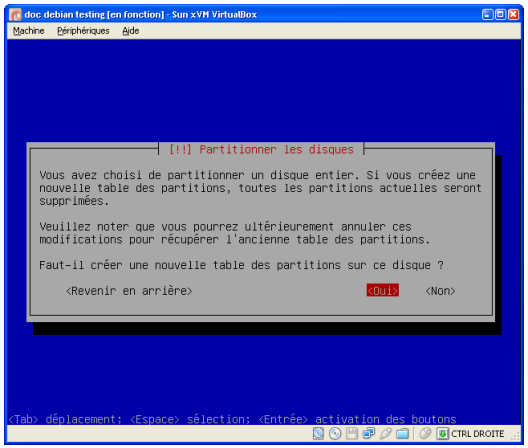
Choix du mode manuel :



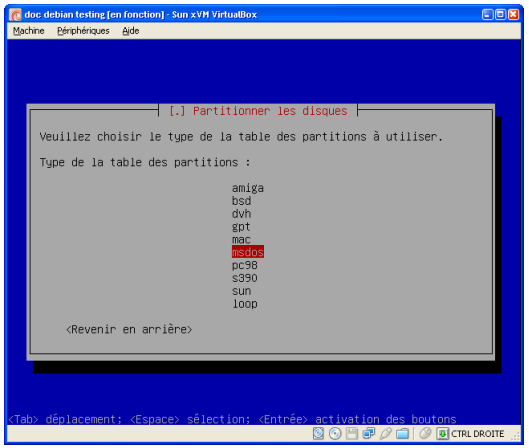
➔



Les 2 prochaines étapes n'arrivent que rarement : disque jamais initialisé (ou disque virtuel) ou grappe RAID fraîchement créée :



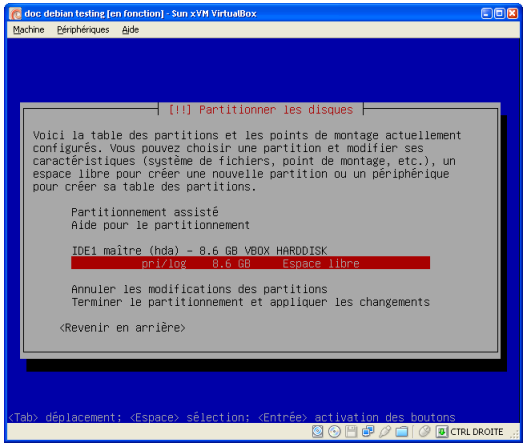
➔



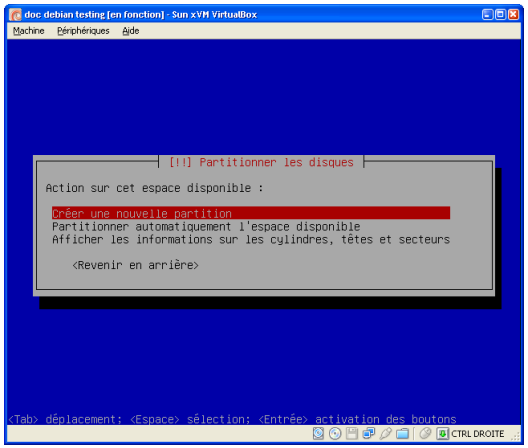
A ce propos, le choix par défaut est normalement le bon. Notez l'existence de « gpt ». Je me suis fait avoir avec un gros disque de 2.5 To (une grappe RAID, peu importe). Voyez sur mon blog cet article.

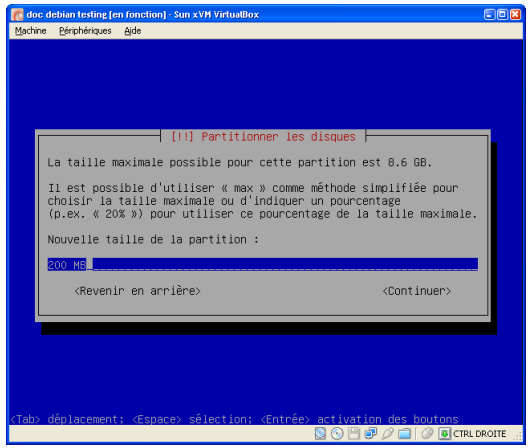
Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)

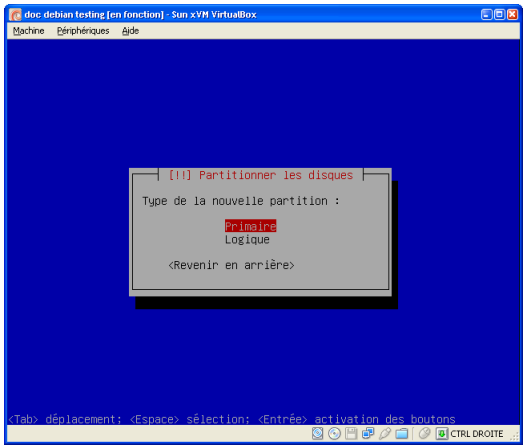
Ceci est le point de départ du partitionnement :



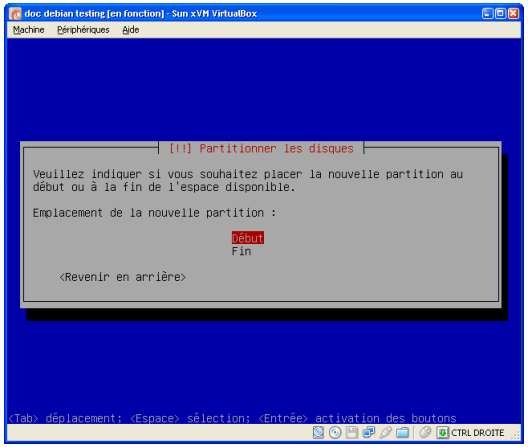
Le principe est d'itérer sur les créations de partitions. Il faut : créer, choisir le type, choisir le point de montage, définir les options, valider ; pour chaque partition. Voici un de ces cycles, ici pour une partition /boot de 200 Mo, avec des options de sécurité (noexec, nodev, nosetuid) :

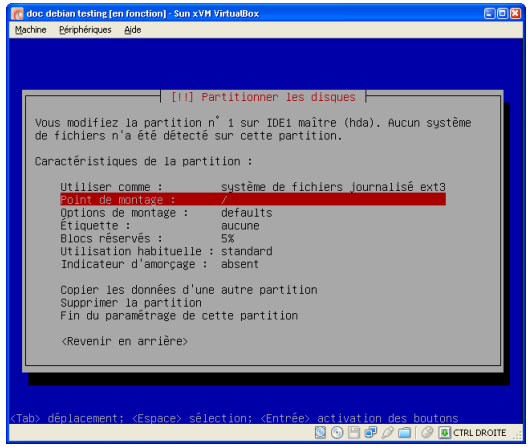


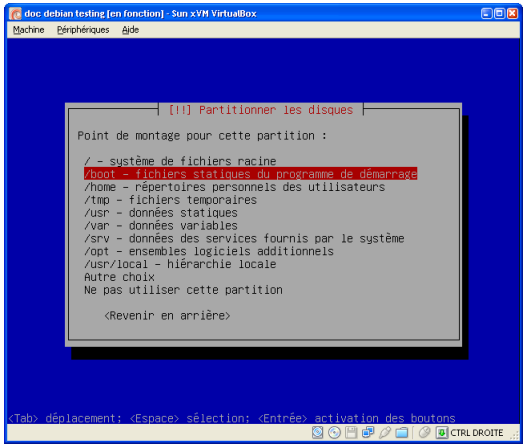




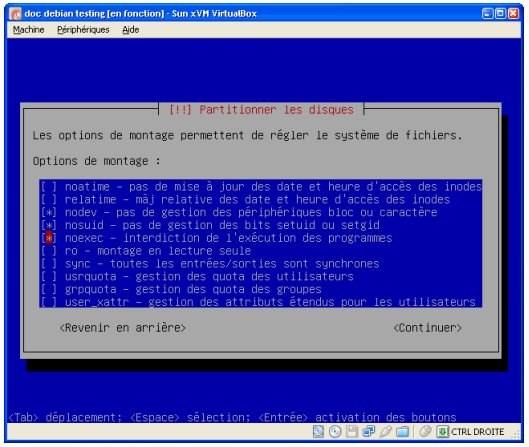
Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)



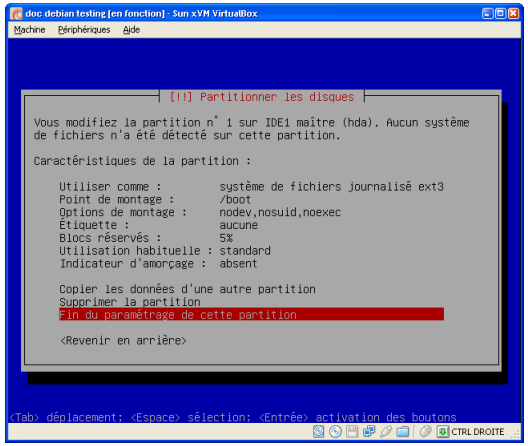




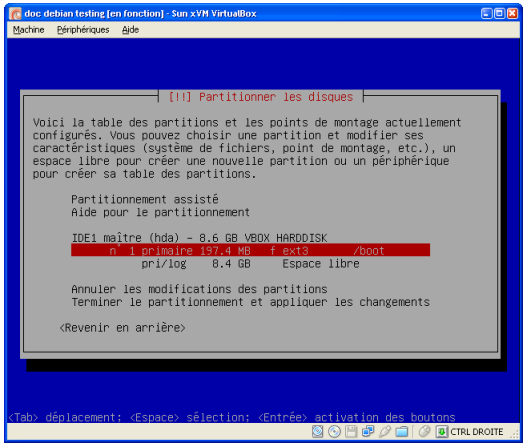
Le menu « Options de montage » donne :



Et enfin on valide cette partition :

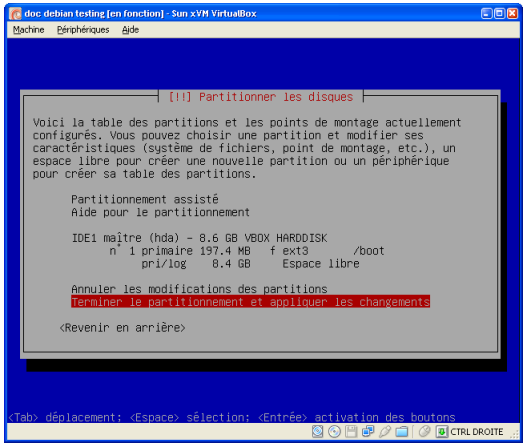


Voici le résulat :

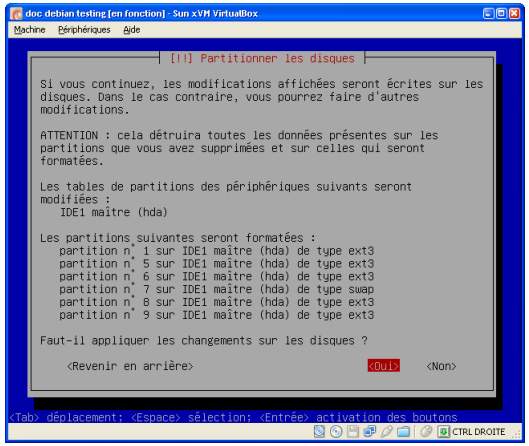


Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)

Une fois toutes vos partitions définies, dernière confirmation :



➔



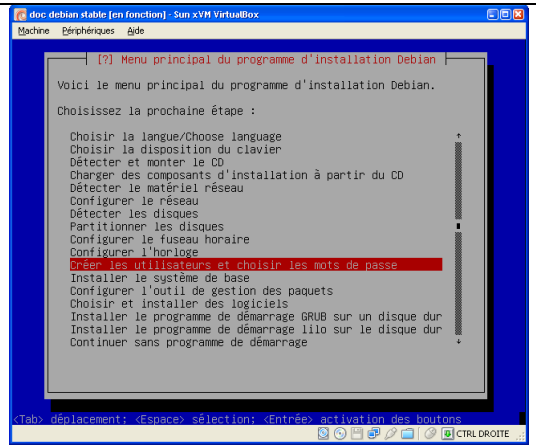
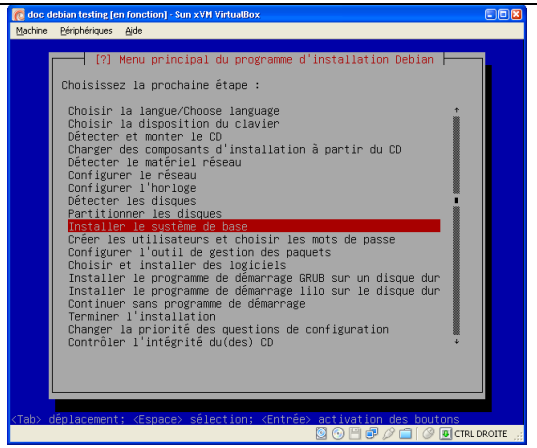
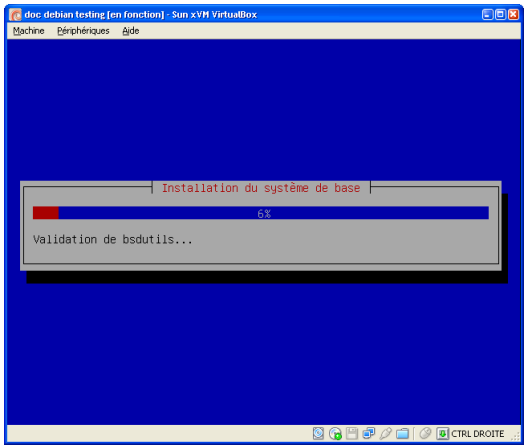
Le programme d'installation applique les modifications et l'installation continue.

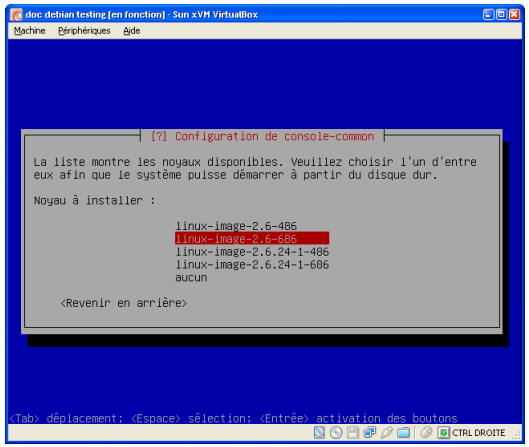
2.7 Configuration de l'horloge / fuseau horaire

En « Etch », la configuration du fuseau horaire a lieu à ce moment là. C'est similaire à la configuration en « Lenny », reportez-vous au chapitre « 2.5 Configuration de l'horloge / fuseau horaire ».

2.8 Installation du système de base

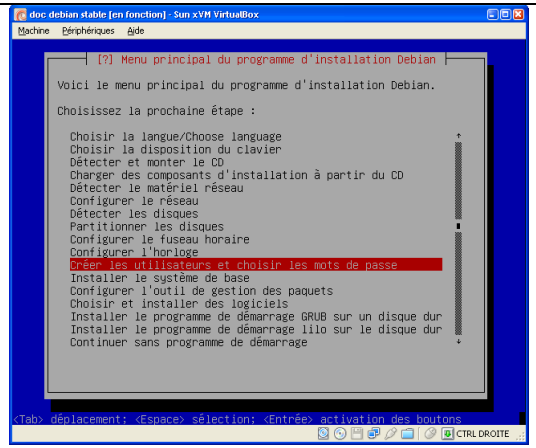
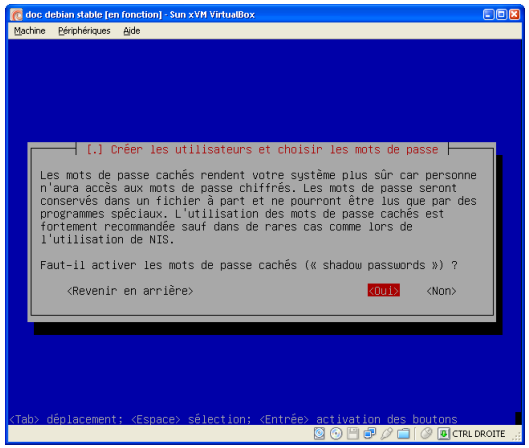
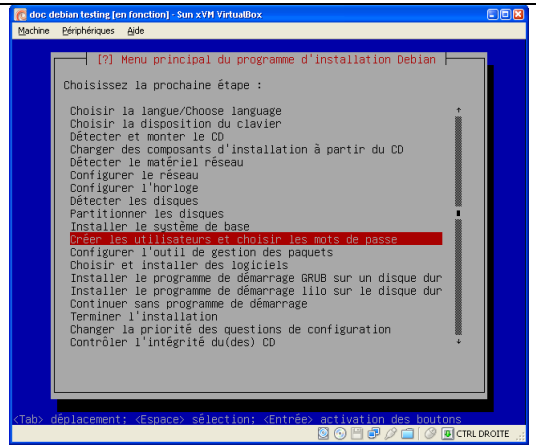
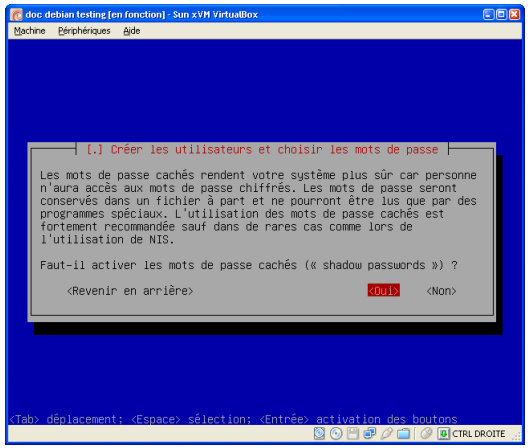
Cette fois-ci, en « Lenny », cette étape intervient avant la création des mots de passe root et d'un compte nommé. C'est l'inverse sur « Etch » ; inversez les chapitres.

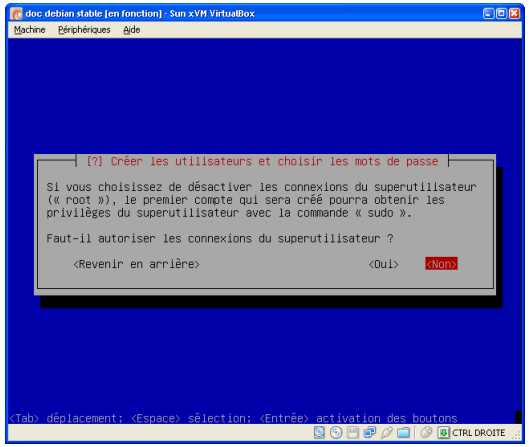
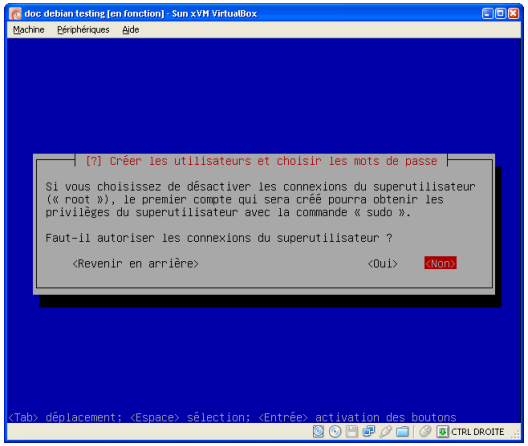
Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)
 Même image en Etch	 Copie des paquets de base du système :  Puis choix du noyau Linux, le nombre de possibilités variera suivant votre CPU. Sachez juste que :	Même image en Etch

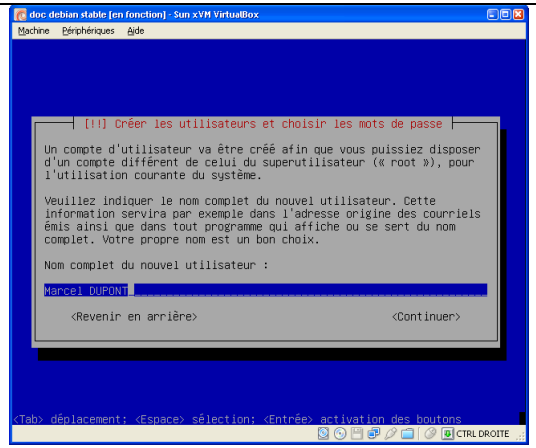
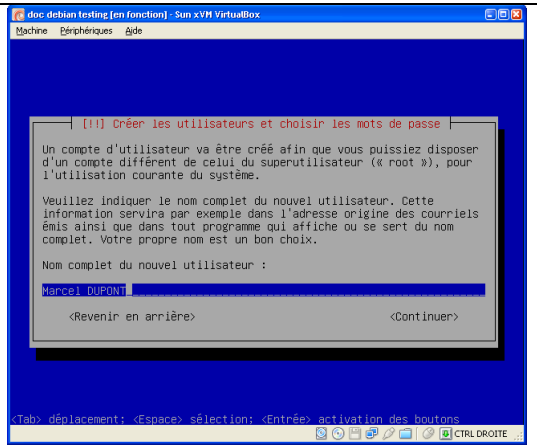
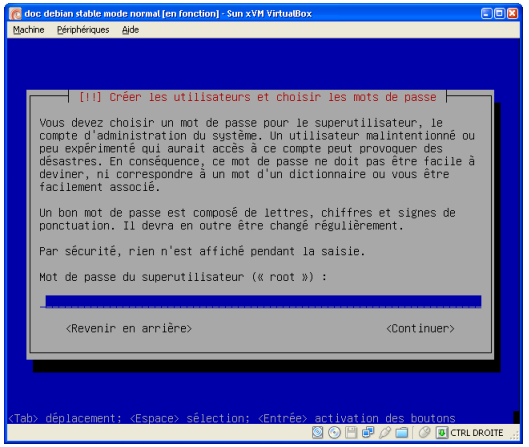
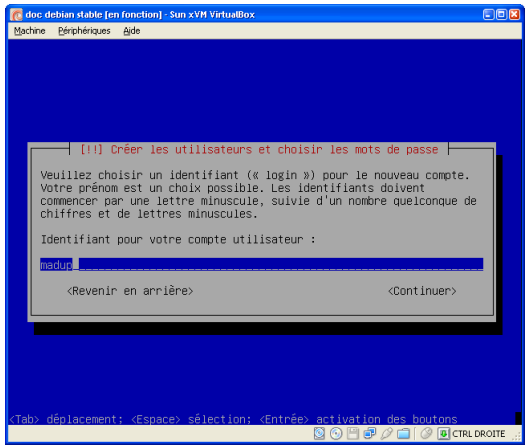
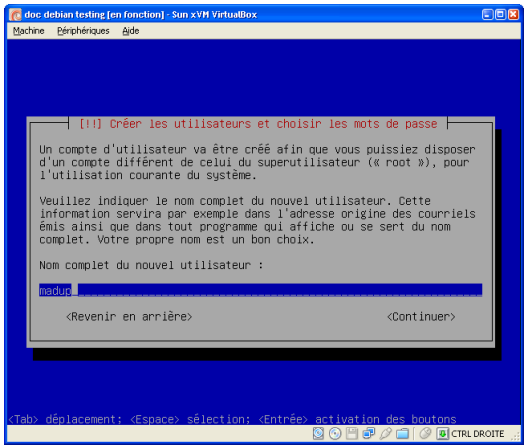
Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)
En Etch, c'est le noyau 2.6.18 et pas le 24.	- SMP = option pour les multi-processeurs (ou multi-core, ou multi-thread) - Le paquet linux-image-2.6-686 est un « méta-paquet », il permettra de maintenir à jour votre kernel 2.6.xx  Puis ça mouline encore un coup	

2.9 Création des comptes utilisateurs

Chapitre à inverser avec le précédent au besoin. Cf. remarque au chapitre précédent.

Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)
 Là on dit « oui », point :  L'écran suivant suit une nouvelle mode qui consiste à faire en sorte que personne ne se connecte en « root » (l'administrateur). Si vous dites « Oui », vous activez ici une option « à la	 Idem :  Même remarque, mêmes photos d'écrans…	Voir plus bas… en Etch « normal » on arrive directement plus loin dans le processus.

Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)
Ubuntu » où les commandes « root » sont accessibles via l'outil « sudo » aux utilisateurs assimilés administrateurs ; un monde où on ne passe plus root en cours de session. Je passe sur les bienfaits/méfaits de ce système. Pour une Debian, j'opte pour « Non », choix un peu historique :  Puis on crée un compte standard :	 	

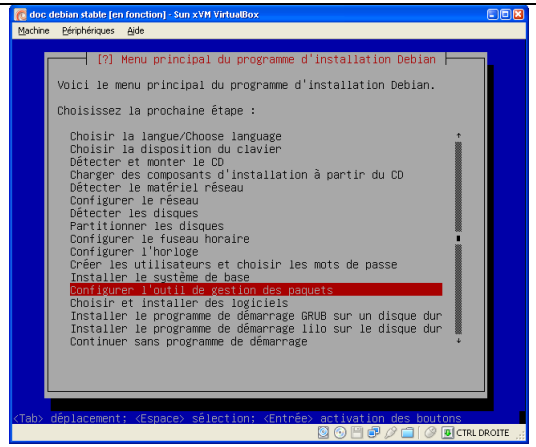
Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)
		
Son identifiant :	Idem :	
		
Puis vient le choix de son mot de passe, je passe, ce n'est pas très intéressant. Et enfin le mot de passe du « root », illustré à droite.	Puis vient le choix de son mot de passe, je passe, ce n'est pas très intéressant. Et enfin le mot de passe du « root », illustré à droite.	

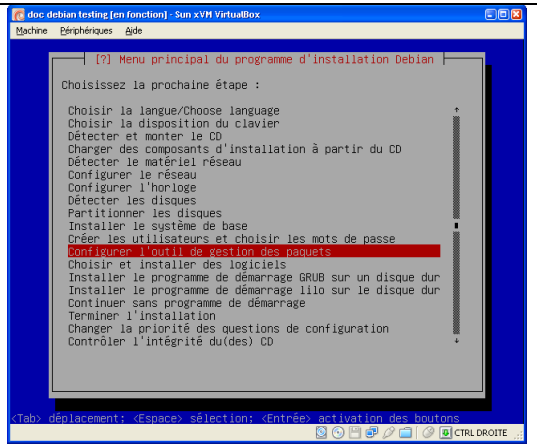
Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)

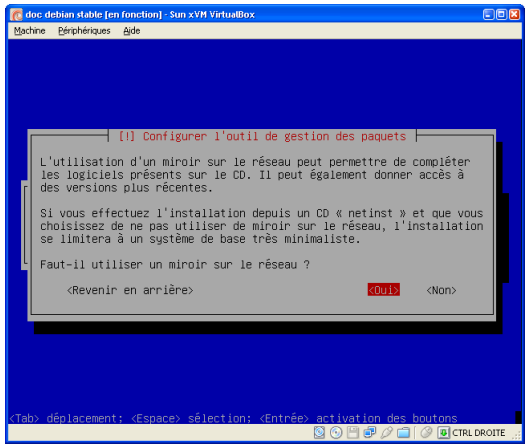
En Stable, vous enchaînez sur l'installation du système de base, cf chapitre précédent.

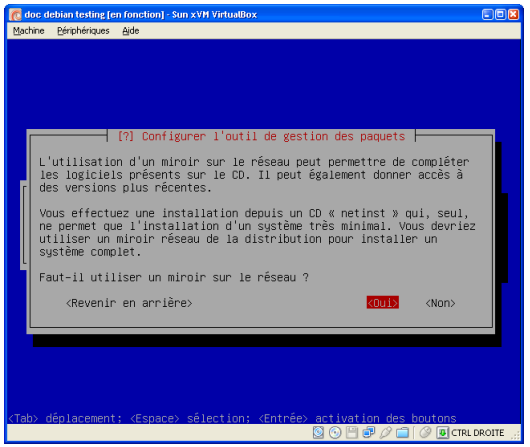
2.10 Configuration de gestion de l'outil de paquets

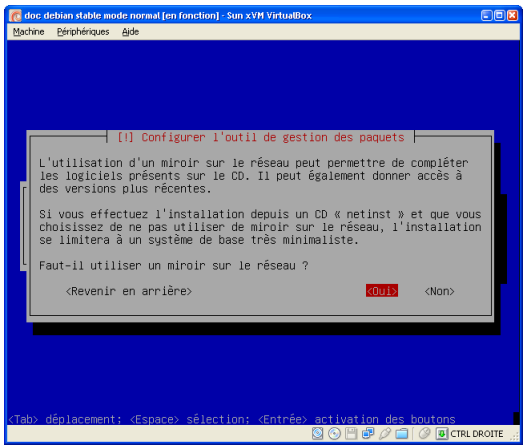
A priori, vous n'avez rien à changer dans cette partie, à moins de vouloir utiliser un dépôt de paquets bien spécifique.



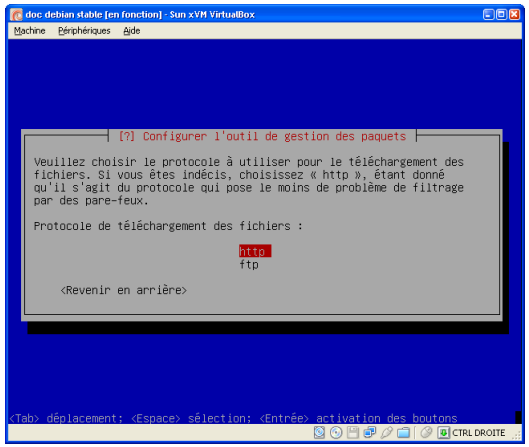


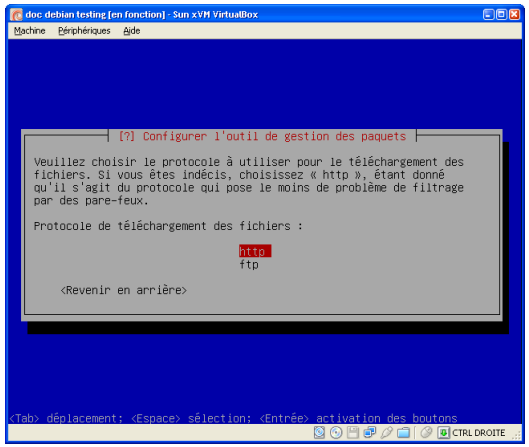


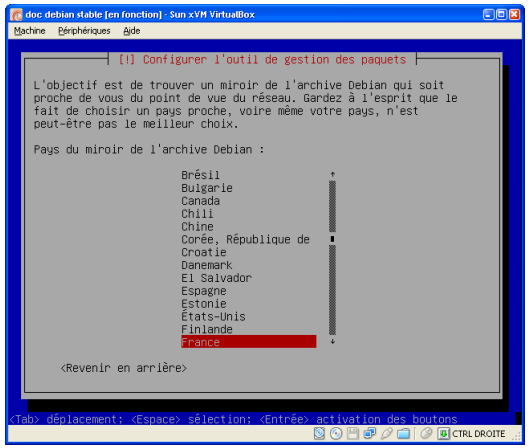


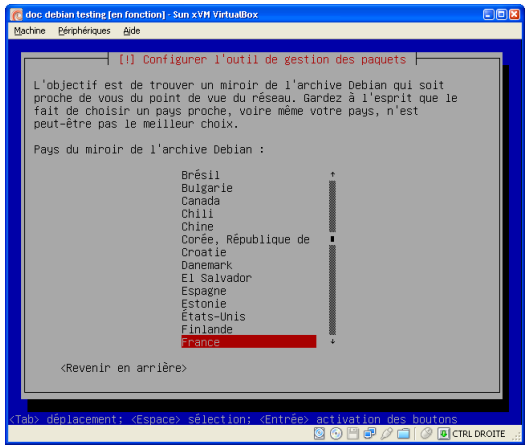


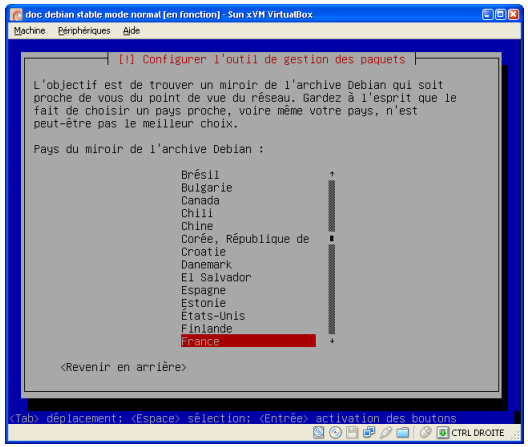
Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)
Choisissez (attention aux proxies en entreprise) :		

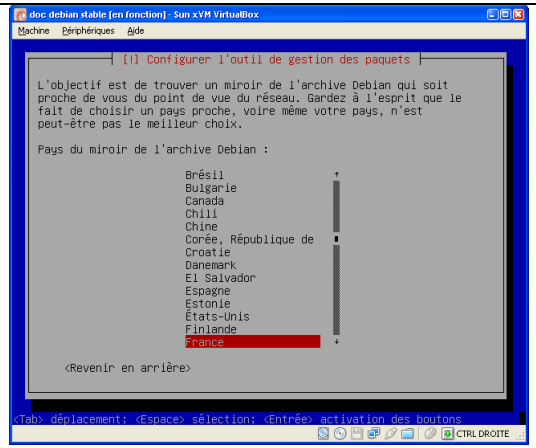
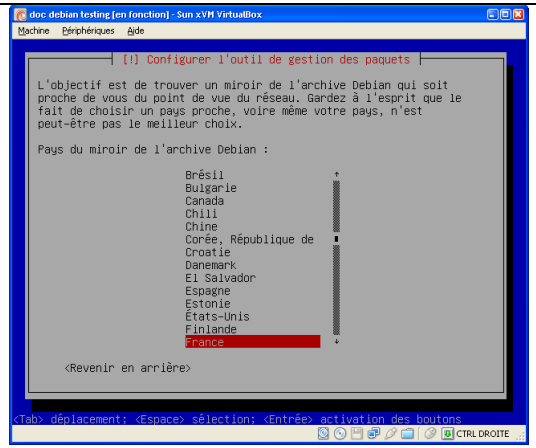
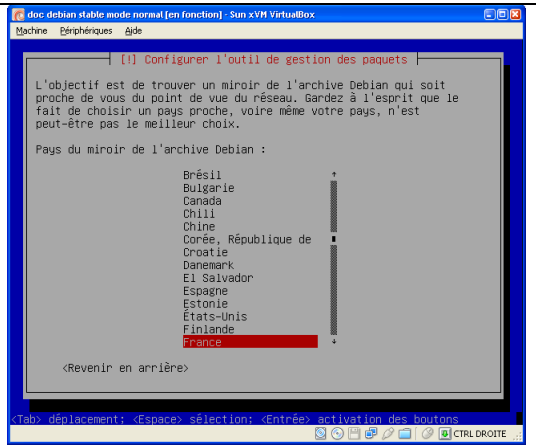
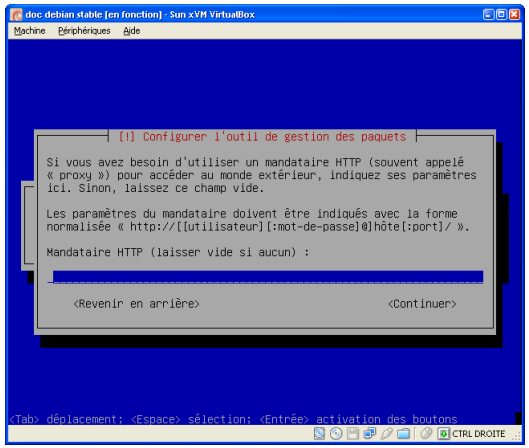
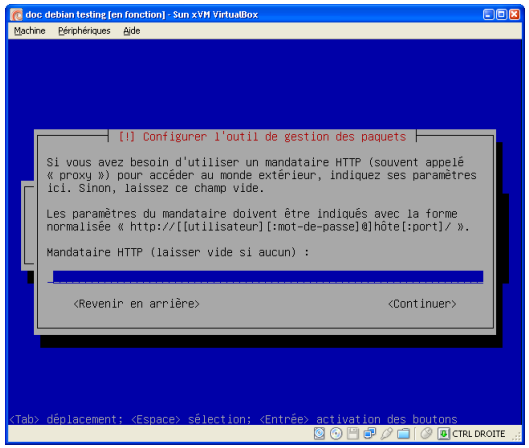
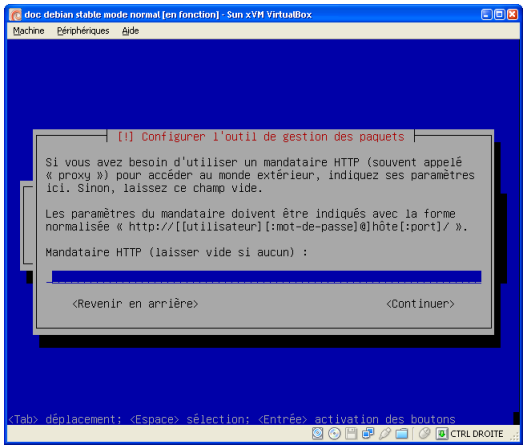


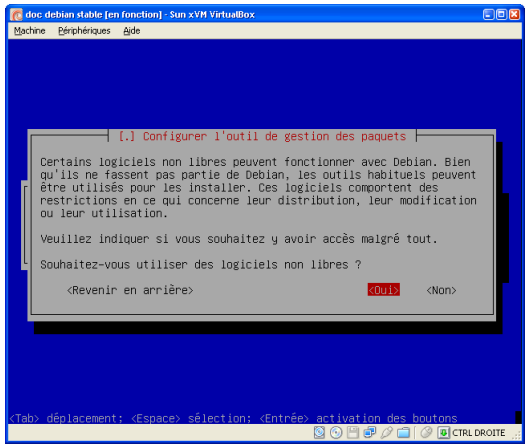
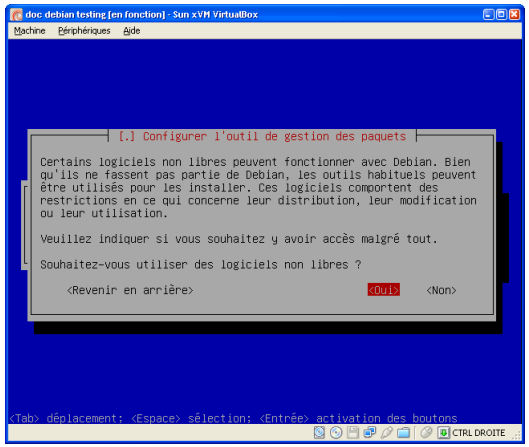


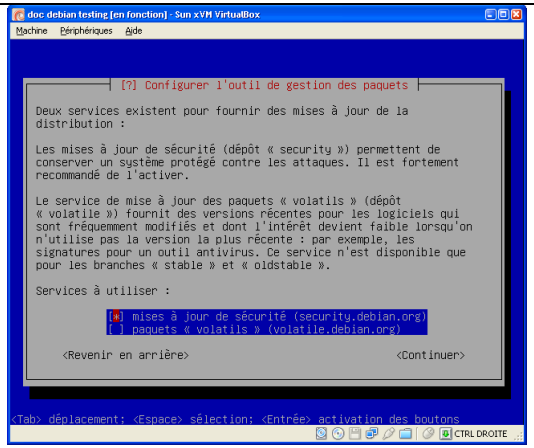






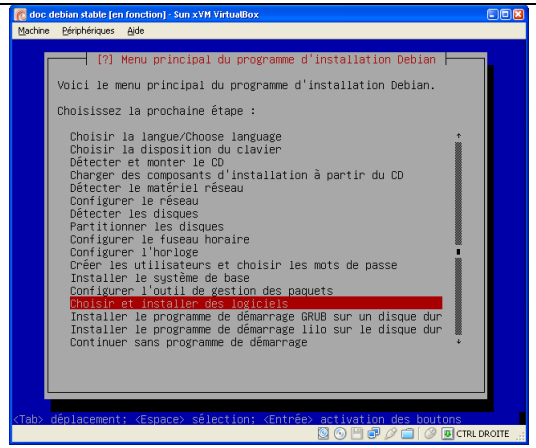
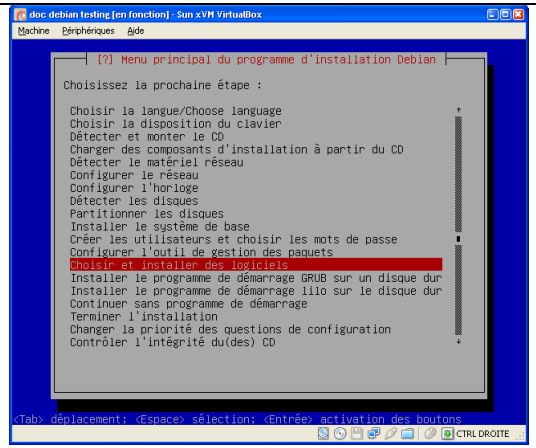
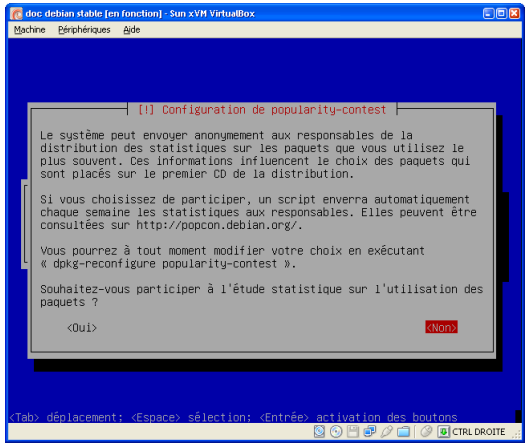
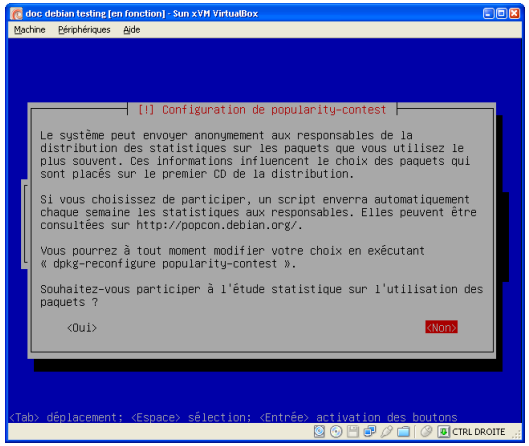
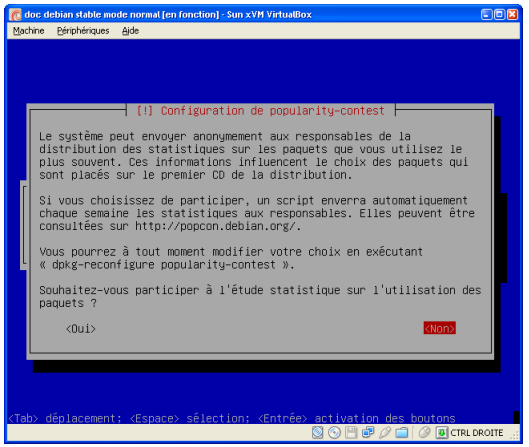
Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)
		
Si vous devez préciser un proxy pour sortir :	Si vous devez préciser un proxy pour sortir :	Si vous devez préciser un proxy pour sortir :
		
Vient ensuite une question cruciale (un troll) sous Debian : voulez-vous ou non inclure les dépôts de logiciels non libres (exemple : drivers NVidia). Ca ne coûte rien – à moins d'avoir une sérieuse		

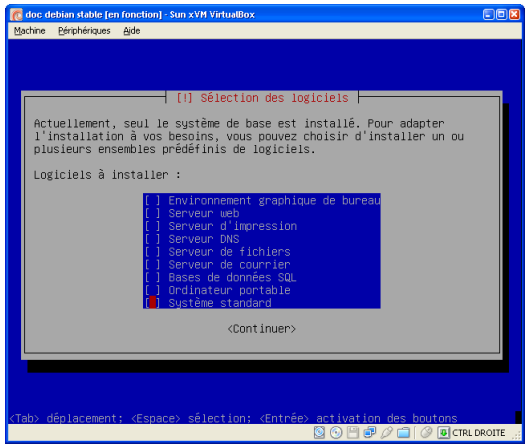
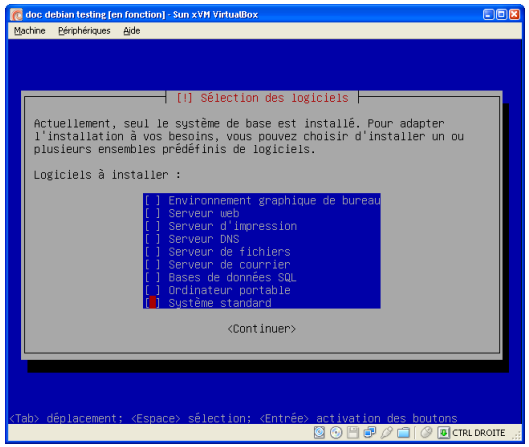
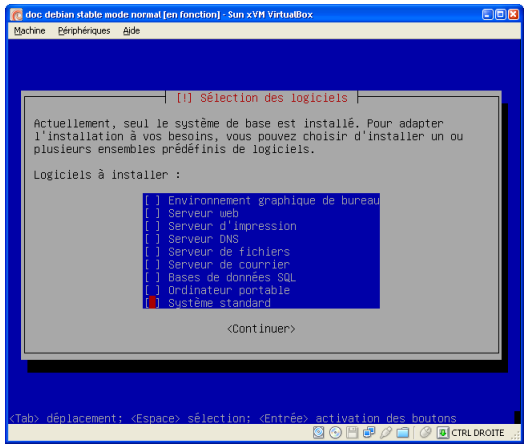
Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)
raison de ne pas le faire.  	 Etape supplémentaire intéressante pour « Lenny » : ce type de dépôt manquait en configuration de base. Le dépôt « securité » doit être ajouté – pas de discussion. Le dépôt « volatile » est utilisé par les applications basées sur des « listes » (au sens large) de données évoluant tous les jours : l'anti-virus s'il ne fallait en nommer qu'un :	Là, je crois que c'est automatique.

Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)
	 A vous de voir si c'est utile pour votre cas (au pire, ça ne dérange pas)	

2.11 Choix des paquets complémentaires à installer ; concours de popularité

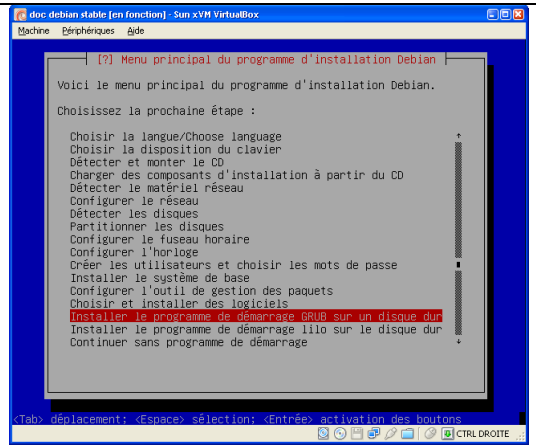
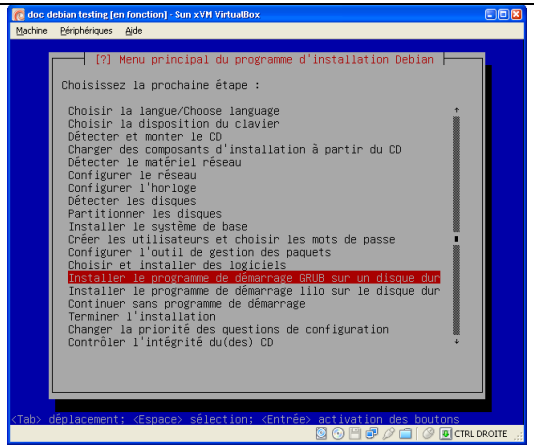
Je pars du principe qu'on installe un serveur pour une raison précise. Pas la peine de l'encombrer de fonctions inutiles (donc de failles liées à des outils pas configurés). Ainsi, je suggère de ne rien choisir à l'installation ; vous éviterez de vous demander si l'option « Base de données » va vous installer un MySQL ou un POSTGRESQL (ou les deux ?) etc. Je décoche tout par principe. Le chapitre « 3 Kit de survie » est là pour vous expliquer comment vous en sortir et choisir unitairement ce dont vous avez besoin.

Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)
		
Etape intermédiaire : voulez-vous remonter des statistiques sur les paquets que vous utilisez ? (oui ne mange pas de pain, sauf en contexte entreprise où toute communication devrait être évitée ?)	Idem :	Les installations se rejoignent ici :
		

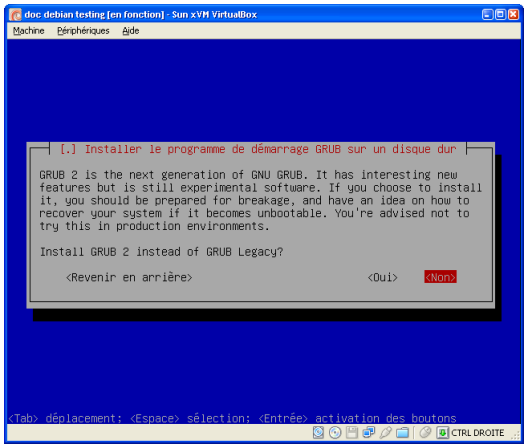
Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)
Nous voilà au cœur du sujet : 		

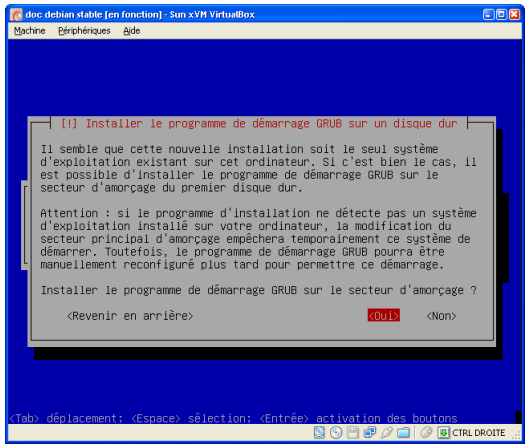
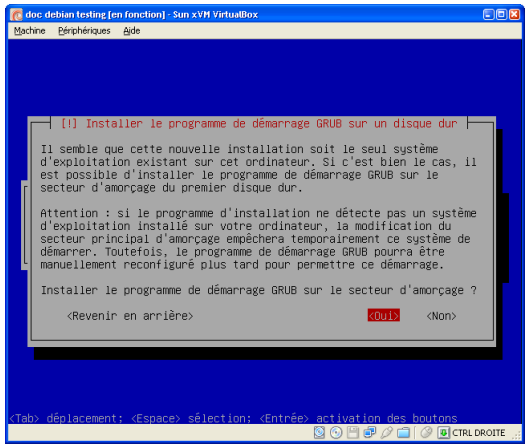
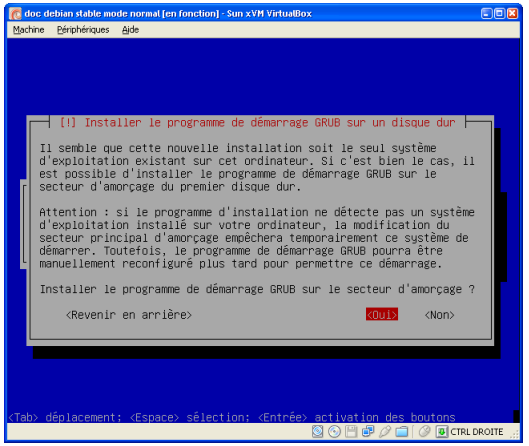
2.12 Programme de « boot »

Le débat est : LILO ou GRUB ? avant, j'étais à fond LILO car je ne connaissais pas GRUB. Maintenant, j'utilise GRUB. Je ne battrai pas avec vous, faites comme vous voulez (si vous laissez faire, ce sera GRUB et ce sera bien comme ça)

Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)
		

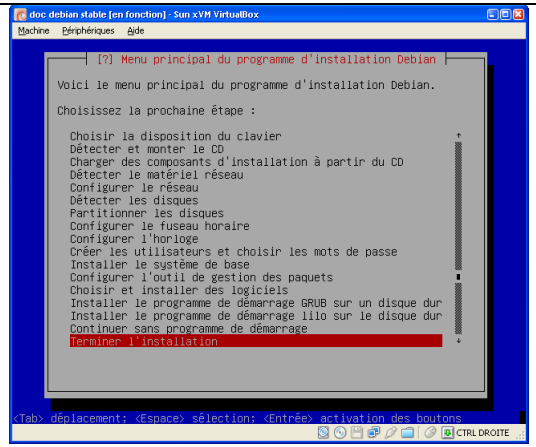
En « Lenny », on vous propose GRUB 2. Vu le message, je suggère d'attendre avant de beta-tester l'outil, dites donc « Non » :



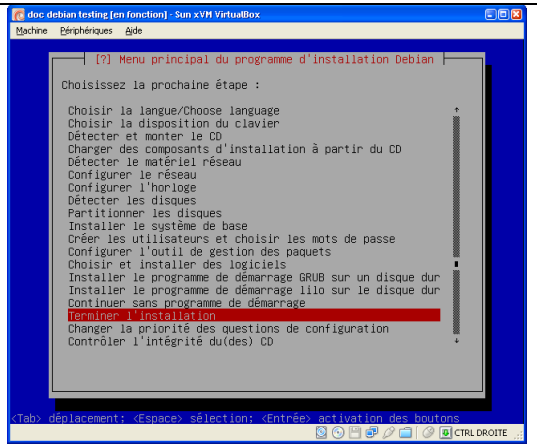
Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)
Installation sur le MBR ?  Dans l'écran suivant, vous pourrez choisir un mot de passe pour GRUB. C'est votre choix.	Idem :  Ici aussi.	En mode normal, c'est GRUB par défaut – il faut faire « Revenir en arrière » si vous voulez changer :  Mais pas en « stable normal »…

Stable « Etch » (mode expert)	Testing « Lenny » (mode expert)	Stable « Etch » (mode normal)

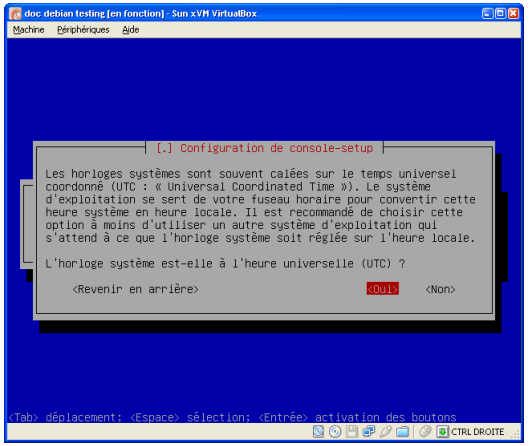
2.13 Enfin, le reboot final

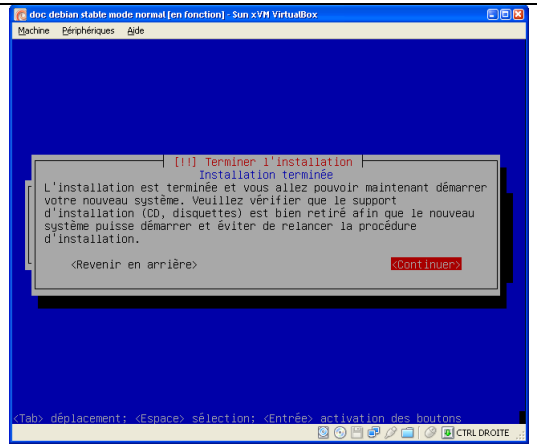


Je passe sur les écrans suivant. Y'a plus qu'à attendre un peu. Et ça rebootera sur votre nouvel OS.



En Lenny, on doit encore régler un détail. A vous de dire si votre horloge de BIOS est réglée sur l'heure UTC ou non :



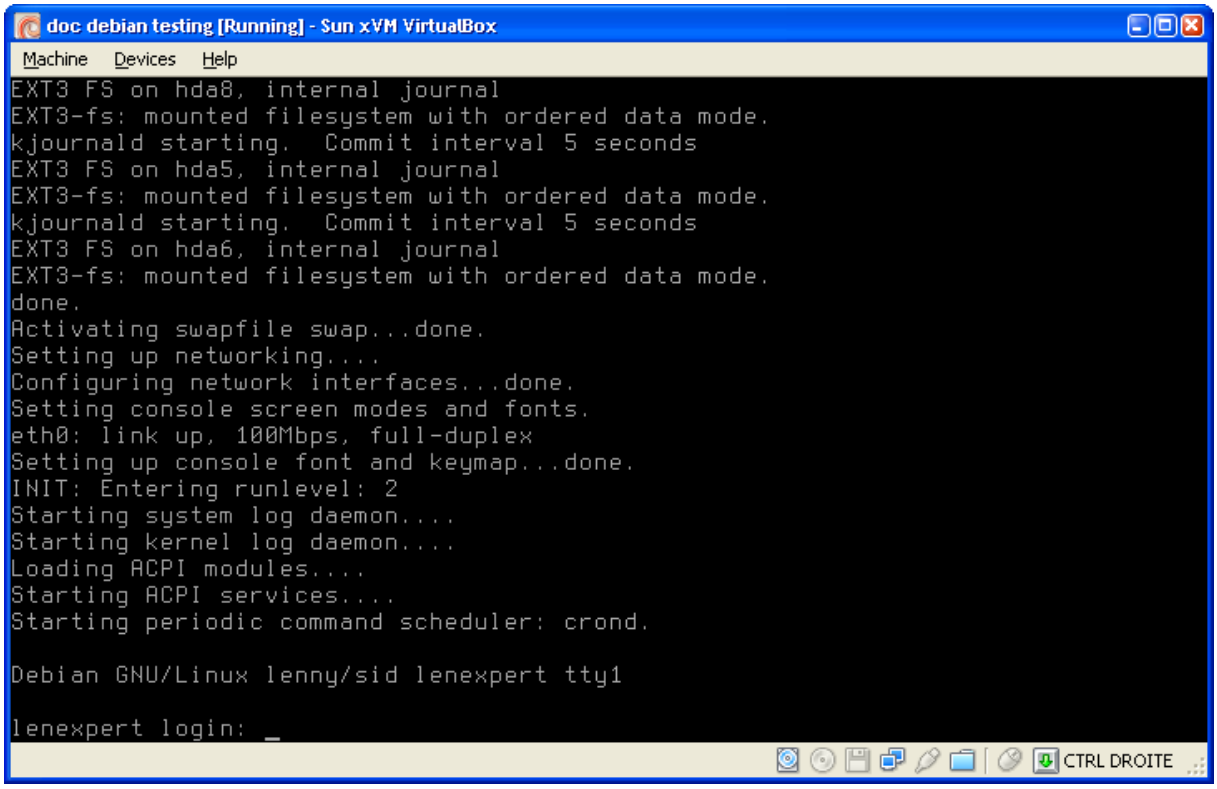


Je passe sur les écrans suivant. Y'a plus qu'à attendre un peu. Et ça rebootera sur votre nouvel OS.

3 Kit de survie : gérer les « paquets »

3.1 Intro

Ca y est, vous voilà devant votre système bien vide :



Et voilà la mire de login. Tant que j'y pense, en tant que l'utilisateur « root », sachez qu'il faut faire « halt » ou « reboot » pour arrêter ou rebooter votre PC. A la rigueur ctrl-alt-suppr. Ou encore « shutdown -h now » ou « shutdown -r now » pour arrêter ou rebooter respectivement.

Vous pouvez donc vous connecter en mode texte en tant que root (mauvaise idée en général, par sécurité, mais là, les commandes que je vais décrire nécessitent des droits root). Je passe sur le B.A-BA des commandes UNIX/LINUX, je vais plutôt me concentrer sur les spécificités de Debian. Je suppose dans la suite de ce texte que vous savez utiliser un éditeur de texte sous Linux. Dans les « Debian testing » récentes, l'outil « nano » est installé en remplacement du célèbre « vi » me semble-t-il. C'est très bien pour un néophyte, qui n'aura pas à se prendre la tête avec l'aspect un peu rebutant de « vi ». Cherchez un peu de doc sur « vi » et « nano » au besoin pour savoir comment les utiliser en mode console. Basiquement avec « nano », c'est « nano /mon_repertoire/mon_fichier ». Avec « vi » aussi, mais c'est juste un peu plus compliqué ensuite :)

Pour l'instant, je vais vous donner le minimum vital pour gérer vos packages, votre système et ainsi pouvoir passer de votre Debian mode console avec rien dedans à quelque chose de plus complet.

Une fois la philosophie acquise, vous pourrez faire n'importe quoi. N'oubliez pas d'aller voir sur http://michauko.org/blog/ pour des articles complémentaires.

3.2 Configuration simpliste de APT

APT (Advanced Package Tool) est **LE** outil de gestion des paquets. Il est extrêmement configurable, je vais juste donner les principaux éléments.

Le fichier de configuration où l'on indique les serveurs de distribution de paquets que l'on utilise est `/etc/apt/sources.list`. Il contient en théorie, suite à l'installation décrite ci-dessus, les lignes suivantes (en omettant celles en commentaires) :

```
lenexpert:~# cat /etc/apt/sources.list
deb ftp://ftp2.fr.debian.org/debian/ lenny main contrib non-free
deb-src ftp://ftp2.fr.debian.org/debian/ lenny main contrib non-free
deb http://security.debian.org/ lenny/updates main contrib non-free
```

La ligne « deb ftp » signifie en gros : « J'ai un serveur de packages deb (proposant des paquets de binaires, par opposition à paquets de sources deb-src), via ftp, à l'adresse ftp2.fr.debian.org, dans le répertoire debian, version « testing » des applis classées dans les groupes « main », « contrib…ution » et « non-free ». En gros.

La ligne « deb-src ftp » signifie la même chose pour les paquets de « sources de logiciels », vous n'en aurez pas besoin à priori. Vous pouvez commenter avec un « # » devant la ligne.

Dernière ligne : même esprit, mais sur LE serveur Debian dédié aux mises à jour de sécurité. Ne jamais oublier cette ligne quand vous êtes en « stable » ou « testing ». Pour info, cette ligne n'existe pas lorsque vous êtes en « unstable » car tout paquet mis à jour est d'abord dispo pour la « unstable », qu'il s'agisse de mise à jour de sécurité ou non.

Si vous deviez un jour ajouter une autre source au fichier, c'est ici que ça se passe. Ce genre de manipulation existe pour des applications développées à droite à gauche et non intégrée officiellement à Debian, mais pour lesquelles l'auteur propose des paquets (fichiers .deb) faits maison. Ou à la rigueur pour mixer les différentes version stable/testing/unstable sur votre système (genre « stabilité de la « stable » mais avec telle application beaucoup plus récente »). Attention ça peut devenir rock n' roll au niveau des dépendances entre versions de logiciels interdépendants.

Vous pouvez reconfigurer par la commande « apt-setup » aussi.

3.3 Mise à jour de la liste des paquets

Une fois vos sources définies, il faut régulièrement dire à Debian de récupérer la liste des paquets disponibles et leurs versions afin de voir ce qu'il y a de neuf à vous proposer. En root, tapez :

```
apt-get update
```

Si votre connexion web est ok, vous verrez une tentative de récupération de plusieurs fichiers auprès des serveurs. Pour automatiser ce traitement, on va installer le paquet « cron-apt » qui vous préviendra (par mail) de la disponibilité de nouveaux paquets parmi ceux que vous utilisez.

Voici un exemple de ce que ça fait :

```
lenexpert:~# apt-get update
Atteint http://security.debian.org lenny/updates Release.gpg
Ign http://security.debian.org lenny/updates/main Translation-fr
Réception de : 1 ftp://ftp2.fr.debian.org lenny Release.gpg [189B]
```

```
Ign http://security.debian.org lenny/updates/contrib Translation-fr
Ign http://security.debian.org lenny/updates/non-free Translation-fr
Atteint http://security.debian.org lenny/updates Release
Réception de : 2 ftp://ftp2.fr.debian.org lenny/main Translation-fr [573kB]
Ign http://security.debian.org lenny/updates/main Packages/DiffIndex
Ign http://security.debian.org lenny/updates/contrib Packages/DiffIndex
Ign http://security.debian.org lenny/updates/non-free Packages/DiffIndex
Réception de : 3 ftp://ftp2.fr.debian.org lenny/contrib Translation-fr
Ign ftp://ftp2.fr.debian.org lenny/contrib Translation-fr
Atteint http://security.debian.org lenny/updates/main Packages
Atteint http://security.debian.org lenny/updates/contrib Packages
Réception de : 4 ftp://ftp2.fr.debian.org lenny/non-free Translation-fr
Atteint http://security.debian.org lenny/updates/non-free Packages
Ign ftp://ftp2.fr.debian.org lenny/non-free Translation-fr
Réception de : 5 ftp://ftp2.fr.debian.org lenny Release [74,5kB]
Réception de : 6 ftp://ftp2.fr.debian.org lenny/main Packages/DiffIndex [2038B]
Atteint ftp://ftp2.fr.debian.org lenny/contrib Packages/DiffIndex
Atteint ftp://ftp2.fr.debian.org lenny/non-free Packages/DiffIndex
Réception de : 7 ftp://ftp2.fr.debian.org lenny/main 2008-11-28-0846.17.pdiff [4612B]
Réception de : 8 ftp://ftp2.fr.debian.org lenny/main 2008-11-28-0846.17.pdiff [4612B]
Réception de : 9 ftp://ftp2.fr.debian.org lenny/main 2008-11-28-0846.17.pdiff [4612B]
654ko réceptionnés en 7s (83,6ko/s)
Lecture des listes de paquets... Fait
lenexpert:~#
```

3.4 Mise à jour des paquets déjà installés

La suite logique de « **apt-get update »** est « **apt-get upgrade –s »**. Dans la plupart des commandes **apt-quelquechose**, le **-s** signifie « simulation ». Je recommande de TOUJOURS lancer les « upgrade » et les (dés)installations (voir plus bas) en mode simulation d'abord. Cela permet d'éviter des ennuis. Exemple : vous upgradez comme un bourrin et vous constatez tardivement que votre firewall a connu un changement majeur de version, vous auriez dû refaire - tout au moins relire - vos fichiers de configuration et vous tenir informé des évolutions entre les 2 versions. Si vous avez upgradé sans trop regarder et donc sans attacher d'importance aux paquets upgradés, votre firewall se met à bouder car les fichiers de configuration sont obsolètes (oui c'est du vécu).

Un autre exemple, vous voulez installer le logiciel « toto » et celui-ci demande 200 paquets en pré-requis, dont certains qui vont à l'encontre (en terme de version) de ceux que vous utilisez - ça peut arriver dans des cas tordus, mixage de versions Debian, intégration de backports etc - et donc il propose innocemment, et afin de satisfaire les dépendances, de tout dégager et d'installer les versions qui lui vont. Si vous ne faites pas attention, votre système risque d'être passablement modifié ou carrément inutilisable. Bref, « **-s** », *toujours*.

Toute la difficulté est de savoir si un paquet *tartempion* est critique ou non. Documentez-vous dessus, au besoin.

Résultat de la commande, si des paquets installés sur votre système ont été mis à jour, APT vous proposera de les downloader. Notez que les paquets sont censés bien se comporter et donc vous poser toutes les questions qu'il faut pendant l'installation de la nouvelle version. Comme d'habitude, il faut lire ce que vous raconte APT. Ne négligez jamais un message d'information et comprenez-le ou recopiez-le dans Google pour comprendre !

Un exemple d'upgrade :

```
lexepert:~# apt-get upgrade
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
Les paquets suivants ont été conservés :
  libsasl2-2 libsasl2-modules sasl2-bin wordpress
Les paquets suivants seront mis à jour :
  fetchmail libimlib2 libxi6 login passwd python2.5 python2.5-minimal spamassassin
8 mis à jour, 0 nouvellement installés, 0 à enlever et 4 non mis à jour.
Il est nécessaire de prendre 7838ko dans les archives.
Après cette opération, 238ko d'espace disque supplémentaires seront utilisés.
Souhaitez-vous continuer [O/n] ?
```

3.5 Pour aller plus loin avec la gestion des paquets

3.5.1 « update » régulier

Idéalement, il faut faire tourner un **apt-get upgrade** régulièrement, que vous installiez ou non des applis, pour des raisons de sécurité. Pensez à « Windows update » si vous êtes un windozien averti, c'est le même concept (en mieux ;)

Pour ne pas oublier l'update et ainsi être au courant qu'il y a des choses à faire, installez le paquet « cron-apt » (apt-get install cron-apt) qui ira voir les listes de mises à jour automatiquement – attention, de base, ça doit tourner la nuit.

3.5.2 aptitude

« aptitude » (installé par défaut) est une surcouche de « apt » qu'il est bon d'utiliser à la place de « apt-get ». Je vous ai décris le premier car il est indispensable de le connaître. « aptitude » permet d'être plus intelligent dans la reflexion : il saura par exemple quels paquets ont installés uniquement par dépendance avec une application que vous vouliez et, à la déinstallation, « aptitude » saura qu'il peut l'enlever. C'est automagique.

« aptitude » a aussi un mode « graphique » (ncurses), voyez :

Enfin, « aptitude » cherche à vous aider lorsque vous installez des paquets qui entrent un peu en contradiction avec d'autres. Il cherche des solutions à votre place. Vous verrez à l'usage.

Dans le même genre d'outils, il y a « synaptic » (à installer). Personnellement, je ne l'utilise pas. De plus, c'est réellement graphique, donc il faut exporter l'affichage ou avoir un serveur graphique sur la machine. Bref : joli, mais passablement inutile.

« aptitude » accepte les mêmes commandes (install, update, upgrade…) que « apt-get » (voir chapitres suivants).

3.5.3 Se faire prévenir des bugs connus

Il arrive qu'une application soit mise à disposition alors qu'il reste des bugs (ou qu'ils ont été découvert après coup), fatalement. Manque de chance, ça peut parfois vous mettre en vrac une application. Même si en général on peut réparer car le bug est connu et le trouve après coup sur Google – au pire on peut réinstaller l'ancienne version dans beaucoup de cas, c'est fâcheux, surtout quand on avait 1 minute à accorder à un upgrade qui va finalement se transformer en 3 heures de catastrophe.

Pour éviter cela, installez le paquet « apt-listbugs », via la commande « aptitude install apt-listbugs ». Avant d'installer tel ou tel paquet (mis à jour ou première installation), le système ira farfouiller dans la base de bugs de Debian si la version du paquet contient des bugs ouverts. Vous connaîtrez l'intitulé du bug, sa gravité et sa référence pour le consulter sur le site de bugs de Debian (http://www.debian.org/Bugs/).

Indispensable.

3.6 Recherche et installation d'applications / de paquets

Pour installer une application dont vous connaissez le nom du package, la commande est

```
apt-get install le_nom_du_paquet -s
```

Même topo pour le **-s**, voir chapitre précédent. Ce mode simulation est très intéressant lors de l'installation car vous allez voir les dépendances, la volumétrie etc mais aussi – et surtout – les paquets recommandés ou suggérés. Lisez les noms des paquets proposés, ça peut servir. Exemple : vous installez une grosse application comme « OpenOffice », vous verrez passer comme paquets recommandés quelque chose concernant la langue française, les dictionnaires etc. Tout n'est pas forcément intégré au paquet principal et vous pouvez aussi prendre connaissance d'applications connexes intéressantes (exemple : un plug-in OpenOffice pour mozilla).

Un exemple si vous installez l'outil de galerie photos « Gallery2 » :

```
[root@toto: /]$ aptitude install gallery2 -s
Reading package lists... Done
Building dependency tree... Done
Reading extended state information
Initializing package states... Done
Reading task descriptions... Done
Building tag database... Done
The following NEW packages will be installed:
  gallery2 wwwconfig-common
The following packages are RECOMMENDED but will NOT be installed:
  dcraw ffmpeg jhead libjpeg-progs postgresql-7.4 postgresql-8.1
0 packages upgraded, 2 newly installed, 0 to remove and 30 not upgraded.
```

```
Need to get 9645kB of archives. After unpacking 60.5MB will be used.
Do you want to continue? [Y/n/?]
```

Vous voyez dans l'exemple précédent notamment « jhead », « dcraw » etc qui sont très intéressants pour ajouter des fonctionnalités à votre gallerie. Votre curiosité pourra vous pousser à vous renseigner sur lesdits paquets ; vous apprendrez forcément des choses, connaitrez des outils insoupçonnés.

Pour rechercher des paquets ou pour avoir une description rapide d'un paquet dont vous avez le nom exact, faites donc un :

```
apt-cache search un_mot_clef un_autre_mot_clef ou_carrement_le_nom_de_l_appli
```

Exemple :

```
apt-cache search open office
```

Vous aurez la liste des paquets concernant OOo et apprendrez du coup l'existence de quelques paquets intéressants : tous ceux concernants le dictionnaire, la traduction en français, quelques autres suites bureautiques, ou encore un plug-in OpenOffice.org pour Mozilla.

Autre exemple plus compliqué le jour où vous cherchez un bon outil de blog (sous-entendu « packagé en standard pour ma version de Debian » :

```
apt-cache search blog
```

Vous aurez une longue liste. Dans cette liste, vous trouverez de tout :

- des paquets pour développer en C++ tel ou tel machin truc de blog (on s'en fout, en général)
- des vrais outils de blog (drupal, wordpress…)
- mais aussi, c'est un exemple, le paquet « flickcurl-utils » qui pourrait intéresser les utilisateurs de FlickR.

3.7 Autres commandes APT & co

Je cite pour info, en vrac :

- **apt-get dist-upgrade** : c'est pour changer de version Debian (passer de « stable » à « testing » ou « unstable » ou encore de « testing » à « unstable »). Le retour arrière est impossible (ou très long et au fil de l'eau dans le cas « unstable » vers « testing ») et il faut bien évidemment avoir indiqué les bonnes adresses pour la unstable dans votre fichier sources.list. *NE PAS UTILISER CETTE COMMANDE SANS SAVOIR CE QUE VOUS FAITES.*
- **dpkg -l** : donne la liste des paquets installés (ou désinstallés) sur votre système. La liste est courte au début, elle ne vas pas le rester longtemps. Pour info, dpkg est en fait LE outil de base de gestion des paquets, APT n'est qu'une surcouche « conviviale ».
- **apt-get install un_paquet -s** : pour installer. Idem pour « aptitude ».

- **apt-get remove --purge un_paquet -s** : pour désinstaller (et purger les fichiers de conf, optionnel). Idem pour « aptitude ».

- **dpkg-reconfigure un_paquet** : pour relancer l'éventuelle configuration du paquet.

- Et évidemment : **man dpkg** et **man apt-get, man aptitude** pour avoir le **man**uel de ces commandes. Pour les débutants, **man une_commande** donne le manuel de la commande, en anglais sauf si vous installez le paquet **apt-get install manpages-fr**).

- **apt-cache search un_mot_clef un_autre** : pour chercher des paquets. Exemple : **apt-cache search multimedia player**.

- **apt-cache show un_nom_de_paquet** : vous obtiendrez des informations détaillées pour le paquet en question.

- **apt-get install -s -t testing_ou_unstable un_paquet** : ça va tenter d'installer le paquet en version « testing_ou_unstable » si vous utilisez une version plus ancienne (la stable par exemple). Très bien pour mixer les versions et bénéficier de certaines nouveautés sans pour autant changer de release, mais attention, les dépendances se compliquent et on peut arriver à un point de non retour. Remarque : il faut ajouter les sources des versions Debian souhaitées dans le `sources.list` et évidemment avoir fait un **apt-get update**.

3.8 Conclusion

Ce rapide aperçu de l'outil de gestion des paquets vous permettra de garder à jour votre système et de pouvoir y installer ce que vous souhaitez (reste à configurer les outils installés !). Il y énormément d'autres choses à dire là-dessus, mais une fois la base acquise, allez trainer sur des blogs pro-debian pour trouver plein d'informations complémentaires (figer une version d'un paquet ; définir des priorités entre « releases » etc).

4 Quelques erreurs « post-installation »

J'ai passé sous silence certaines opérations importantes en fin d'installation, les voici (si elles n'ont pas changées depuis le temps)

4.1 Problème de caractères accentués

Je ne sais plus trop si ce problème arrive encore. Toutefois, si votre système vous affiche des accents comme ça :

```
Les paquets supplÃ©mentaires suivants seront installÃ©sÂ :
```

Faites donc un :

```
dpkg-reconfigure locales
```

Et assurez-vous d'avoir sélectionner les « locales » « fr_FR ISO-8859-1 », « fr_FR.UTF-8 UTF-8 », « fr_FR@euro ISO-8859-15 » et fixez la dernière par défaut (ou la première, long sujet polémique ☺

A noter que sur un serveur sérieux, il vaut mieux tout configurer en anglais (donc les locales « en_USblabla ». Ainsi, vous aurez des messages d'erreur en anglais, plus facile à recracher dans Google pour chercher de l'aide.

4.2 C'est tout à ma connaissance

Les choses ont donc évolué.

5 Outils indispensables

5.1 Serveur SSH

SSH est « Secure SHell », une sorte de telnet réellement sécurisé. Ca sert à ouvrir une session texte à distance dans de « bonnes conditions ». C'est l'outil indispensable sur votre serveur Debian pour l'administrer à distance. Installez donc le paquet « ssh » sans hésiter. Si vous administrez votre serveur Debian depuis un poste Windows – cas classique – utilisez le client PuTTY, premier lien sur Google.

Je ne vous détaille pas les différences entre authentification par couple clefs publiques et clefs privées ou par mot de passe, vous trouverez tout ce qu'il faut sur le web à ce niveau là.

En fait, je tenais surtout à mentionner cet outil indispensable pour l'administration de serveur d'une part et en remplacement du serveur FTP (présenté ci-dessous) d'autre part. SSH intègre une fonction de transfert de fichier, en mode SFTP « SSH File Transfer Protocol » (ne pas confondre avec du FTP sur SSL). Ca ne coûte qu'un peu de temps CPU en plus pour le cryptage/décryptage, mais ça sécurisé vos échanges (les mots de passe FTP sont en clair sur un réseau).

Lorsque vous avez un serveur SSH opérationnel, vous pouvez l'utiliser pour du transfert de fichiers grâce à des outils comme WinSCP, Filezilla etc. Je parle de clients Windows – cas classique d'utilisation là aussi. Sous Linux, il y a des tonnes de clients texte ou graphiques. Le seul hic de cette méthode est que vos amis en général n'utilisent pas ce mode et il faut leur expliquer :) si ces amis doivent échanger des données avec votre serveur.

Si vous donnez accès à votre serveur à des gens pour qu'ils utilisent uniquement un compte mail IMAP (voir chapitre sur le sujet), vous pouvez estimer qu'ils ne feront jamais de SSH « réel ». Dans ce cas, vous pouvez leur interdire l'accès SSH ou au contraire lister les gens autorisés (ou forcer l'utilisation de clef plutôt que l'authentification par mot de passe). Ceci vous protègera contre des accès interdits à votre serveur. Sachez que la plupart des failles de tel ou tel « morceau de Linux » n'est souvent exploitable que depuis un compte local. Autant ne pas offrir sur un plateau le compte de belle-maman qui n'a jamais compris l'intérêt d'un mot de passe de plus de 3 lettres.

Pour ce faire, jouez avec ces paramètres particulièrement intéressants dans /etc/ssh/sshd_config :

```
mon_serveur:~# cat /etc/ssh/sshd_config
Protocol 2 # et uniquement le 2 !
UsePrivilegeSeparation yes
PermitRootLogin no # utilisez su depuis un compte non-admin
PermitEmptyPasswords no
AllowUsers toto tata autre_login # ou DenyUsers, au choix
```

Redémarrez le service via « /etc/init.d/ssh restart ».

5.2 Monter un serveur FTP

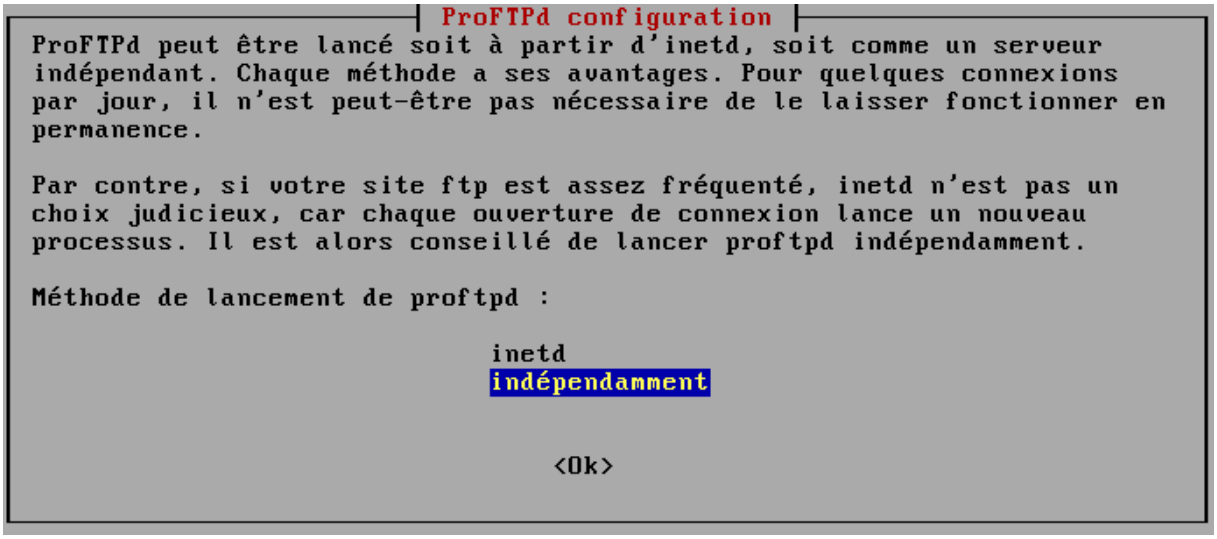
Alors premièrement, c'est quelque chose que je ne recommande pas. Ou alors pour des transferts de votre réseau local vers votre serveur/firewall Debian, par commodité – voir mon baratin au chapitre précédent.

A vous de bien configurer votre firewall, notamment un module particulier du noyau à ne pas oublier, voir sur mon blog cet article.

Pour installer un tel service (si vraiment vous avez besoin), installez par exemple l'application « ProFTPD », un serveur libre largement reconnu sur la planète. Je vous propose une configuration simpliste pour, basiquement, uploader/downloader des fichiers avec des comptes nommés (pour l'anonyme, ne comptez pas sur moi ;)

```
apt-get install proftpd
```

A vous de voir pour la question suivante, c'est assez clair (qu'est-ce qu'inetd ?) :



Ensuite, vous passerez au minimum une fois sur le fichier `/etc/proftpd.conf` afin de voir quel type d'informations on peut mettre. Vous verrez que par défaut, le compte « anonymous » n'est pas activé. Il faut donc faire des transferts FTP avec un compte nommé, c'est-à-dire un compte déclaré sur l'OS.

5.3 Régler l'heure

Je voulais juste suggérer d'installer le paquet « ntpdate » qui vous permettra de vous synchroniser à des serveurs de temps (avec des horloges atomiques tu penses, faut bien ça).

Votre heure machine sera mise à jour à peine la paquet installé (il me semble).

5.4 Optionnel : synthétiser vos logs : paquet logwatch

Installez le paquet « logwatch ».

C'est quasimment impensable de lire tous vos /var/log/ importants (sans compter les sous-répertoires apache2, etc). Installez le paquet logwatch, ne faites rien, vous aurez un mail synthétique (adressé au root) le lendemain matin :)

C'est très pratique, quoique non exhaustif et ça donne un bon aperçu de ce qu'il peut se passer sur votre serveur.

5.5 Bannir les vilains lourdingues du web : fail2ban

Avec « logwatch », vous verrez que certains lourdeaux tentent de percer des passwords ssh – par exemple – à longueur de temps, avec des robots je suppose.

Installez le paquet « fail2ban ». En gros, lorsqu'une IP va faire trop d'erreur en trop peu de temps, vous bannirez cette IP momentanément. Le paramétrage de l'application en standard est convenable.

5.6 Environnement bureautique ?

Bon, si c'est ça le but de votre serveur Debian (simple PC bureautique sous Debian ?), j'avais dit d'installer Ubuntu ! Donc, en quelques mots, voici comment installer rapidement un environnement qu'il faudra peaufiner longtemps ensuite.

Il vous faudra au minimum les paquets suivants :

```
aptitude install gnome xserver-xorg
```

Eventuellement le paquet « gdm ». Voilà, je ne détaille pas plus. Quand je monte un PC bureautique sous Linux, maintenant, c'est Ubuntu ☺

6 Firewalling sous Linux

Un des sujets les moins simples, mais ô combien crucial ! Une fois votre machine connectée au web, c'est la première chose à faire, clairement avant d'installer une base MySQL ou un serveur FTP... Sinon, vous avez un risque – suivant votre configuration, que tel ou tel service soit accessible sur le web, avec des mots de passe par défaut (donc connus), sans que vous ne vous en doutiez (puisque vous ne lisez pas les logs ;)

6.1 Principe du firewalling sous Linux

Sous Linux (noyau 2.4 et +), les fonctions de firewalling sont intégrées au noyau. Je passe sur les détails. L'outil de gestion des règles s'appelle « iptables » et c'est merdique à utiliser pour un non initié, avouons-le. Lignes de commandes tordues etc. Nuits blanches à la clef. Ceci dit, le manuel raconte presque tout (« man iptables »).

Ca tombe bien, un mec bien sympa a écrit « shorewall » pour rendre ça lisible. Ca reste en mode texte, mais c'est beaucoup plus lisible et vous vous concentrez uniquement sur les règles particulières à donner au firewall. Des scripts tout prêts sont disponibles sur le site web www.shorewall.net pour les cas classiques (PC passerelle, poste seul, DMZ etc).

Le principe de mise en place est, outre le **aptitude install shorewall** qui va bien, de copier les squelettes de fichiers depuis /usr/share/shorewall (je crois) ou depuis le site web (qui donne des exemples), de les paramétrer et **enfin – et à la fin seulement – d'indiquer que le firewall est configuré et peut être pris chargé** (fichier « /etc/default/shorewall » à modifier, valeur « startup=1 ») et c'est parti.

Je ne vais pas faire ici un cours sur les concepts de sécurité de vos machines, je vous décris un (le ?) outil sous Debian pour avoir une machine bien protégée avec shorewall.

6.2 Mise en place pour une utilisation simple type passerelle & serveur

Si vous souhaitez mettre en place un linux qui fait office de firewall/routeur/serveur mail/etc avec un réseau privé derrière, utilisez les exemples ci-dessous. J'insiste sur le paramètre IP_FORWARDING cité plus bas. Vous repenserez à lui lorsque vous perdrez 2 heures à chercher pourquoi votre réseau local ne passe plus la passerelle alors que votre Debian a accès au web.

Ce sont des exemples de fichiers de configuration d'une machine à moi en shorewall 3.0.3-1 – je ne mentionne que les fichiers pour lesquels j'ai apporté des modifications, les autres sont restés tels quels. Ma liste de fichiers dans /etc/shorewall/ est : *accounting action.Limit actions action.SSHKnock action.Whitelist blacklist interfaces Limit masq params policy routestopped rules shorewall.conf SSHKnock start stop tcstart tos tunnels Whitelist zones*.

Je ne recopie ici que les parties variables (hors lignes démarrant par #, donc). L'ordre de modifications des fichiers est relativement logique, afin de comprendre ce qu'on raconte au firewall.

6.2.1 Fichier « /etc/shorewall/interfaces »

Ici, on indique nos différentes interfaces réseaux. Exemple d'un cas avec une ligne ADSL sur une carte Ethernet, connectée sur eth1 et le réseau local sur eth0. Les options sont documentées dans les commentaires du fichier, elles permettent de se protéger contre quelques attaques classiques et précisent 2/3 choses pour le firewall. C'est assez simple : vous dites « le net est sur eth1 » et « le réseau local est sur eth0 » :

```
###############################################################################
#ZONE INTERFACE BROADCAST OPTIONS
net eth1 detect dhcp,tcpflags,norfc1918,routefilter,nosmurfs,logmartians,blacklist
loc eth0 detect dhcp,tcpflags,detectnets,nosmurfs
```

6.2.2 Fichier « /etc/shorewall/zones »

Ici, on indique simplement que notre réseau est constitué de 3 éléments : le firewall en lui-même, le réseau Internet et le réseau local. Ces noms de zones sont la base de tout, on écrira des règles de firewall grâce à ces noms, exemple « autoriser tout de loc vers net » pour permettre à votre réseau local de sortir sans restriction. En quelque sorte, on relie les « interfaces » à ces noms de zones.

```
###############################################################################
#ZONE TYPE OPTIONS IN OUT
# OPTIONS OPTIONS
fw firewall
net ipv4
loc ipv4
```

6.2.3 Fichier « /etc/shorewall/policy »

Ici on décrit macroscopiquement les règles par défaut pour le firewall. Si un cas n'est pas énuméré dans le fichier « rules » (voir ci-dessous), alors on applique les règles générales décrites ici. Les commentaires décrivent l'action :

```
###############################################################################
#SOURCE DEST POLICY LOG LEVEL LIMIT:BURST
loc net ACCEPT # signifie qu'on accepte que le réseau local sorte
# vers Internet – sous-entendu sans restriction
# On pourrait être plus filtrant dans le fichier rules
# If you want open access to the Internet from your Firewall
# remove the comment from the following line.
$FW net ACCEPT # Ici on tolère que le FireWall sorte vers Internet, sans
# restriction de ports de destination non plus
$FW loc ACCEPT # idem pour FW -> loc – c'est bourrin : en cas de
# compromission de votre passerelle, le LAN est exposé. Il
# faut déporter la sécurité au niveau des PC du LAN dans ce
# cas.
net all DROP # le reste n'est pas toléré et ignoré (DROP), notamment le
# net -> FW. La raison est qu'on va énumérer les règles au
# compte-goutte dans le fichier « rules ».
#
# THE FOLLOWING POLICY MUST BE LAST
#
all all REJECT INFO # là c'est pour se prémunir de cas loufoques :
# spoofing etc
```

6.2.4 Fichier « /etc/shorewall/rules »

Le gros du sujet : on décrit port par port et zones par zones les exceptions aux règles générales définies dans « policy ». Je ne donne que quelques exemples :

```
#############################################################################################################
###############
#ACTION SOURCE DEST PROTO DEST SOURCE ORIGINAL RATE USER/
# PORT PORT(S) DEST LIMIT GROUP
# PORT PORT(S) DEST LIMIT GROUP
# loc -> fw
Ping/ACCEPT loc fw # pour qu'on puisse faire des ping du réseau local vers le linux
FTP/ACCEPT loc fw # idem pour FTP : compléter pour SSH ou n'importe quoi d'autre
# ces 2 commandes disposent d'Alias prédéfinis simplifiant la syntaxe
# sinon, il faudrait taper :
ACCEPT loc fw tcp 3128 # pour tolérer loc -> fw sur le proxy Squid
ACCEPT loc fw tcp 80 # ........................... serveur Web
# d'autres exemples :
ACCEPT loc fw tcp 5901:5910 # plusieurs VNC
ACCEPT loc fw udp 8767 # TEAMSPEAK
ACCEPT loc fw tcp 14534 # TEAMSPEAK conf web
# net -> loc (NAT en entrée du firewall)
# pour rendre accessible des services hébergés sur votre réseau local par du mapping de ports
DNAT net loc: 192.168.0.7 tcp 4662 # emule hébergé sur le LAN
DNAT net loc: 192.168.0.7 udp 4672 # emule
DNAT net:212.27.38.253loc:192.168.0.7 tcp8080# freeplayer.freebox.fr
# net -> fw : accès aux services hébergés sur le linux
SMTP/ACCEPT net fw # pour le serveur de mails
IMAP/ACCEPT net fw
SSH/ACCEPT net fw
ACCEPT net fw tcp 6881:6885 # bittorrent
# etc...
# fw -> loc : je l'ai en POLICY ACCEPT, je garde ca pour la syntaxe
#Ping/ACCEPT fw loc
#VNC/ACCEPT fw loc:192.168.0.5
#ACCEPT fw loc:192.168.0.6 tcp 5900 # VNC windows
```

Vous avez pigé le principe ? ce n'est pas plus dur à maintenir qu'une interface graphique à la firewall de Windows XP. Ca va même plus vite et c'est plus précis comme ça. Plus facile à sauvegarder aussi...

6.2.5 Fichiers « /etc/shorewall/masq » et « shorewall.conf »

Le fichier « masq » décrit l'interface réseau et la plage d'IP autorisée à « traverser » le Linux – qui joue donc un rôle de passerelle.

```
#############################################################################
#INTERFACE SUBNET ADDRESS PROTO PORT(S) IPSEC
eth1 192.168.0.0/24
```

« eth1 » étant mon ADSL, on dit ici que le sous-réseau 192.168.0.x peut « aller vers » eth1, donc aller sur le web.

Parallèlement, l'unique paramètre que je modifie dans le fichier « shorewall.conf »est :

```
#
# ENABLE IP FORWARDING
#
# If you say "On" or "on" here, IPV4 Packet Forwarding is enabled. If you
```

```
# say "Off" or "off", packet forwarding will be disabled. You would only want
# to disable packet forwarding if you are installing Shorewall on a
# standalone system or if you want all traffic through the Shorewall system
# to be handled by proxies.
#
# If you set this variable to "Keep" or "keep", Shorewall will neither
# enable nor disable packet forwarding.
#
IP_FORWARDING=On # super important, sinon le LAN ne sortira pas !
```

6.2.6 Fichier « /etc/shorewall/routestopped »

Il faut indiquer quelles éventuelles interfaces continuent de fonctionner lorsqu'on arrête le firewall :

```
############################################################################
#INTERFACE HOST(S) OPTIONS
eth0 192.168.0.0/24
```

Donc, mon LAN sur eth0 et sur toute sa plage affectable pourra communiquer avec le linux lorsque je ferai un : « /etc/init.d/shorewall stop »

6.3 Relance du bazar

Voilà, avec ces quelques fichiers, votre firewall devrait ronronner. Les sules modifications que vous aurez à apporter au quotidien sont sur le fichier « rules » et plus rarement « policy ».

A la fin, vous faites un : « /etc/init.d/shorewall restart »

Et vous voyez si tout concorde (accès web, depuis votre LAN, depuis le firewall). Vous pouvez aussi regarder le fichier de log « /var/log/shorewall-init.log ».

6.4 Dernières remarques

Idéalement, vous contrôlerez la pertinence de vos accès entrants depuis l'extérieur en allant en SSH chez un copain Linuxien puis ferez un « nmap » vers votre machine pour voir ce qui est ouvert (ou bleu :(.

Attention : quand vous « upgradez » votre système, méfiez-vous toujours d'un upgrade majeur de Shorewall qui nécessite de revoir assez profondément les fichiers de configuration. Je me suis fait piéger maintes fois. Dans tous les cas, je vous recommande de toujours prendre les nouvelles versions de fichiers, « shorewall.conf » inclus, et donc de penser au paramètre « IP_FORWARDING=On ».

7 Serveur SMTP (mail), accès IMAP, anti-spam, webmail etc

Gros sujet. Ce chapitre présente l'installation et la configuration de :

- Un serveur de mails
- Un accès en IMAP (genre POP3 en mieux pour ceux qui ne connaissent pas).
- Un accès webmail.
- Le tri automatique et efficace du SPAM, à plusieurs niveaux.
- La sécurisation de l'accès SMTP par SASL, (authentification sur les comptes Linux).

Un des pré-requis pour que cette fonctionnalité de serveur ressemble à quelque chose, c'est que vous ayez un nom de domaine. Au pire un nom du genre « chezmoi.dyndns.org », au mieux un vrai nom comme « mondomaine.org ». Je prends l'exemple de « mondomaine.org » ci-dessous.

J'ai retenu les outils suivants :

- « postfix » pour le serveur de mails, la bibliothèque « libauthen-sasl » pour le SASL.
- « procmail » comme « processeur » (outil de traitement des mails).
- « courier-imap » pour le serveur IMAP
- « squirrelmail » pour le webmail – voyez sur mon blog cet article pour mettre IlohaMail (moins moche)
- « postgrey » pour le GreyListing
- « spamassassin » et « pyzor » pour (enfin, contre ;) le SPAM
- Vous ajouterez aussi Rules Emporium et tout un tas d'autres trucs pour affiner le tri du spam. C'est décrit sur mon blog ici.
- Plein de paquets « *sasl* » pour la mise en place de l'authentification en envoi.

Quelques compléments d'informations sur mon blog :

- Attention à un bug SASL – ça date, mais je n'ai pas rejoué avec SASL depuis.
- Partage de boîtes mails IMAP avec courier-imap
- Créer des répondeurs automatiques, ce genre de trucs
- Faire mumuse avec un proxy POP/IMAP

7.1 Mise en place de postfix

Installez le paquet postfix et procmail :

```
apt-get install postfix procmail courier-imap
```

Il faut installer courier-imap dès à présent afin d'avoir la commande « maildirmake » (voir plus bas).

Configurez le fichier prinpical de postfix « /etc/postfix/main.cf ». Le fichier est à ce stade incomplet, il se limite à décrire le serveur, il n'y a rien sur l'envoi sécurisé. Je mets **en gras** les passages importants :

```
monlinux:# cat /etc/postfix/main.cf
# See /usr/share/postfix/main.cf.dist for a commented, more complete version
smtpd_banner = $myhostname ESMTP $mail_name (Debian/GNU)
biff = no
# appending .domain is the MUA's job.
```

```
append_dot_mydomain = no
# Uncomment the next line to generate "delayed mail" warnings
#delay_warning_time = 4h
myhostname = mail.mondomaine.org
mydomain = maison.loc
alias_maps = hash:/etc/aliases
alias_database = hash:/etc/aliases
myorigin = /etc/mailname # ce fichier contient le nom du serveur copié lors d'un envoi
mydestination = $myhostname, $mydomain, localhost.localdomain, localhost, mondomaine.org,
mail.mondomaine.org
append_dot_mydomain = no
#relayhost = smtp.free.fr
fallback_relay = smtp.free.fr # si le serveur en face rejette votre serveur, on s'appuie sur
celui
# de votre fournisseur, ici FREE. Ce n'est pas obligatoire.
mynetworks = 127.0.0.0/8, 192.168.0.0/24 # ceci pour autoriser l'envoi depuis mon LAN sans
# authentification (voir smtpd_recipient/sender
mailbox_command = /usr/bin/procmail # on utilise PROCMAIL pour traiter les mails entrants
mailbox_size_limit = 0
recipient_delimiter = +
inet_interfaces = all # sauf si vous voulez écouter sur une interface réseau seulement
smtpd_recipient_restrictions = permit_mynetworks, reject_unauth_destination
smtpd_sender_restrictions = permit_mynetworks
reject_unauth_pipelining
reject_unauth_destination
reject_unknown_sender_domain
```

Il y a peu à dire à ce stade sur le fichier, notez tout de même que postfix enverra les mails au programme « procmail » pour traitement. Pourquoi ? Car on va utiliser un format de boîte mail particulier et le mail doit être traité par procmail pour être « converti » au niveau de la boîte mail de l'utilisateur.

Il faut donc maintenant paramétrer **chaque compte utilisateur déjà existant** habitilité à recevoir des mails pour qu'on les traite correctement.

Exemple en configurant votre « home directory » pour qu'il reçoive vos mails. Créez donc un fichier « .procmailrc » dans votre répertoire personnel :

```
toto@monlinux:~# cat ~/.procmailrc
PATH=$HOME/bin:/usr/bin:/usr/ucb:/bin:/usr/local/bin:.
MAILDIR=$HOME/Maildir/ # You'd better make sure it exists. Ne pas oublier le "/" de fin !!!
# Sinon on n'est pas en format Maildir !!!
DEFAULT=$MAILDIR
LOGFILE=$MAILDIR/log
LOCKFILE=$HOME/.lockmail
SHELL=/bin/sh
```

Le fichier ci-dessus ne fait aucun traitement sinon indiquer qu'il faut stocker le mail reçu dans le répertoire « ~/Maildir ». Nous verrons plus tard comment lui faire trier le spam.

J'ai choisi de stocker les mails au format « maildir », c'est une arborescence de mails, contrairement au format mbox (tous les mails en un fichier). Il nous faut créer le répertoire « Maildir » via la commande « maildirmake » (installée avec l'outil courier-imap) :

```
toto@monlinux:~$ maildirmake Maildir
toto@monlinux:~$ ls -lR Maildir/
Maildir/:
```

```
total 12
drwx------ 2 test users 4096 2006-07-31 11:48 cur
drwx------ 2 test users 4096 2006-07-31 11:48 new
drwx------ 2 test users 4096 2006-07-31 11:48 tmp
Maildir/cur:
total 0
Maildir/new:
total 0
Maildir/tmp:
total 0
```

Voilà, votre arborescence vierge est créée. N'y touchez pas à la main sans raison.

Enfin, relancez le service « postfix » pour prendre en compte les modifications : « /etc/init.d/postfix restart ».

Si votre firewall le permet, vous devriez être capable de recevoir les mails. N'hésitez pas à aller lire les fichiers « mail.* » dans « /var/log/ » pour débugger au besoin.

Pour l'envoi, le paramètre « **mynetworks** » indique pour l'heure que votre LAN (s'il est bien en 192.168.0.x) peut envoyer des mails sans authentification.

Il reste à :

- Accéder avec un client mail, par le protocole IMAP
- Accéder en webmail
- Trier le spam et enfin ouvrir votre serveur en envoi depuis n'importe où, après authentification !

Si vous ne comptez envoyer des mails que depuis votre Debian ou depuis votre réseau local, vous pourrez zapper la partie SASL. Mais si vous comptez par exemple donner des comptes Linux (et donc mails) à des amis, il serait agréable qu'ils puissent utiliser votre SMTP pour envoyer, plutôt que leur fournisseur d'accès. Ce chapitre est décrit en dernier. Pour l'instant, il faut pouvoir aller lire vos mails :

7.2 Mise en place de l'IMAP

A ce stade, on sait recevoir des mails et les stocker chez les utilisateurs. Reste la consultation. Si vous utilisez Thunderbird par exemple pour lire vos mails, on va se créer un accès en IMAP et vous configurerez votre compte pour utiliser le serveur « imap.mondomaine.org » pourvu qu'il existe dans votre DNS.

Par rapport à du POP3 (plus connu en général), en IMAP, il y a toujours synchronisation entre tous vos clients mails (thunderbird au boulot, à la maison, sur le portable, chez maman etc) et votre base de mails. Les mails restent sur le serveur. C'est du coup aussi synchronisé avec ce que vous verrez en « webmail » (voir chapitre suivant). Les répertoires où vous pourrez trier vos mails sont aussi synchronisés du coup. Bref, que du bonheur. J'installe aussi l'IMAP car le client webmail que j'utilise demande un accès IMAP, pas POP.

On a installé précédemment le paquet « courier-imap » car il fourni l'outil maildirmake.

Vérifiez bien que le paramètre suivant colle avec votre fichier « ~/.procmailrc » des chapitres précédents :

```
monlinux:/etc/default# cat /etc/default/courier
# This file is automatically generated by maintainer scripts.
# You may modifiy this file, but additional values and format
# modifications will not be preserved.
MAILDIR="Maildir"
```

Configurez votre client mail et ça devrait rouler. Il suffit simplement de le faire pointer sur votre serveur IMAP, port 143, avec votre login Linux et mot de passe associé.

7.3 Mise en place du webmail

Je considère que vous avez déjà un serveur Apache opérationnel (un chapitre décrit ceci dans ma doc, avec Gallery2). Allez, là aussi c'est simple, on installe ceci :

```
apt-get install squirrelmail squirrelmail-locales
```

Ensuite, un outil est fourni pour configurer Squirrelmail. Par défaut, tout est nickel, **mais il faut passer au moins une fois dans ce menu et sauvegarder la configuration afin « d'activer le site ».**

```
monlinux:~# /etc/squirrelmail/conf.pl
SquirrelMail Configuration : Read: config.php (1.4.0)
---------------------------------------------------------
Main Menu --
1. Organization Preferences
2. Server Settings # ici vous indiquerez où est votre SMTP, sur quel port écoute l'IMAP etc
3. Folder Defaults
4. General Options
5. Themes
6. Address Books
7. Message of the Day (MOTD)
8. Plugins
9. Database
10. Languages
D. Set pre-defined settings for specific IMAP servers
C Turn color on
S Save data
Q Quit
Command >> q
You have not saved your data.
Save? [Y/n]: n # Vous mettrez Y évidemment
Exiting conf.pl.
You might want to test your configuration by browsing to
http://your-squirrelmail-location/src/configtest.php # vous pouvez tester ça pour être sûr
Happy SquirrelMailing!
```

Ensuite, il faut indiquer à votre serveur web un « alias » pour l'URL du webmail. Exemple avec Apache2 :

```
monlinux:~# cat /etc/apache2/conf.d/webmail
Alias /webmail /usr/share/squirrelmail # vous choisissez le nom
<Directory /usr/share/squirrelmail>
Options ...
...
```

Redémarrez votre serveur web et accéder à votre compte via « http://mondomaine.org/webmail/ ».

Ce n'est pas magnifique, mais très fonctionnel :

7.4 Tri du spam

On peut envisager la chose sous différents aspects :

- En premier lieu, votre serveur peut chercher à identifier le spam dès la tentativede réception d'un mail sur votre serveur. Ce sont des critères sur le cheminement du mail (typiquement son origine à priori) qui sont utilisés.

- Puis, pour les mails qui passent (qui à priori ont une raison de passer), on peut encore faire des tests plutôt liés au contenu du mail pour identifier le spam. Ce sera fait au niveau de l'utilisateur, de sa boîte mail lorsqu'un mail va y être écrit.

On va mettre tout ça en place. Vous verrez, c'est marrant, ça donne de meilleurs résultats que les filtres anti-spam que j'ai eu l'occasion de subir chez différents fournisseurs de mails gratuit. C'est pour ça que j'utilise mon propre serveur :) pas uniquement pour l'aspect geek de la chose :)

7.4.1 Principes

7.4.1.1 Niveau serveur : Greylisting, RBL et Cie

Si je ne dis pas de bêtise :

Le « greylisting » se situe entre le blacklisting et le whitelisting – non je ne rigole pas. Le blacklisting, c'est le fait de bannir d'office telle ou telle IP, plage d'IP, hostname etc. Le whitelisting, c'est au contraire de dire amen à telle IP (d'un ami digne deconfiance par exemple). Le greylisting, c'est de forcer le serveur distant à « prouver » qu'il est de bonne foi. En gros, les spammeurs ont un objectif : spammer le plus possible. Ca veut dire : consonmmer (payer) du réseau et du CPU pour générer les mails. Déjà qu'on les embête à analyser leur image-spam et donc à les faire dépenser du temps CPU pour générer des images de plus en plus complexes à analyser, ils ne vont quand même pas nous envoyer un spam 2 fois !!!! C'est là le truc : avec le greylisting, lorsqu'un serveur inconnu s'adresse à votre serveur, on refuse le mail et on lui demande de le renvoyer (ça se passe au niveau serveur, ça n'embête pas l'utilisateur). Les spammeurs ne le font pas, voilà. Les autres renvoient le mail. Au bout de 5 fois qu'un serveur s'est fait embêté par nos soins, on le « whiteliste » et zou, fini de l'embêter.

A mon avis ça ne durera pas comme méthode, mais pour l'instant, c'est assez efficace.

Les listes RBL et autres sont, pour faire simple, des listes d'IP connues comme étant des spammeurs ou spammeurs potentiels (trop d'openrelays connus, comme chez proxad, et oui !). Donc on va dire à notre serveur de mails d'utiliser ces listes pour rejeter purement et simplement certains vilains. A nous de choisir des listes intelligentes, mises à jour etc.

7.4.1.2 Niveau utilisateur : spamassassin, pyzor, rulesemporium etc

« Spamassassin » est un outil de détection par règles et apprentissage, et pyzor, une base de données online recensant des clefs de hachage (il me semble) sur des mails identifiés comme SPAM. Pyzor est utilisé directement et silencieusement par spamassasin, il n'y aura donc rien à faire que d'installer le paquet. « Spamassassin » se base sur une analyse de contenu pour détecter les spams

(mots-clefs etc). Les techniques de spam évoluant chaque jour, il faut faire évoluer spamassassin aussi car ses méthodes d'apprentissage sont comprises et détournées par le spammeurs. Actuellement (décembre 2006), installer spamassassin et pyzor uniquement ne servira presque à rien.

C'est pour cette raison que je vous parle de www.rulesemporium.com qui est un site donnant des règles mises à jour régulièrement. Ce sont des règles souvent très précises qui identifient tel ou tel type de spam qui était/est en vogue.

Lisez le contenu des filtres ici et notez cette URL qui n'est pas dans le menu de leur site (pourquoi ?). Le premier lien contient tous leurs filtres, l'autre un filtre spécial qui fait de l'analyse des images : sans aller jusqu'à de l'OCR, ça émet des hypothèses sur le ratio des couleurs de pixels par exemple (ce qui explique sans doute la complexification récente des image-spam avec des grands pâtés de couleurs et des caractères mal alignés).

7.4.2 Mise en place de tout ça

7.4.2.1 Greylisting, RBL & co

Installer le paquet « postgrey » : « apt-get install postgrey ». Vous pourrez consulter les fichiers dans « /etc/postgrey » pour info.

Ensuite, il faut dire à « postfix » de se référer à greylist lors de la réception d'un mail. On ajoute alors la ligne suivante dans le fichier « /etc/postfix/main.cf » :

```
smtpd_sender_restrictions = ... ajoutez : check_policy_service inet:127.0.0.1:60000
```

Cette adresse réseau est la porte d'entrée du processus postgrey (un port réseau sur votre machine).

Pour le RBL et tout ça, vous n'aurez qu'à trouver des listes publiques qui vous conviennent et de modifier à nouveau le champ « smtpd_sernder_recipients » pour y faire figurer par exemple :

```
smtpd_sender_restrictions = ... ajoutez ce qui suit :
reject_rbl_client sbl-xbl.spamhaus.org
reject_rbl_client list.dsbl.org
reject_rbl_client smtp.dnsbl.sorbs.net
reject_rbl_client web.dnsbl.sorbs.net
```

Attention à l'ordre des valeurs dans une instruction de « main.cf ». Je donne à la fin un exemple de fichier « main.cf » complet et qui-fonctionne (tm).

Remarque sur les RBL : vous n'êtes pas à l'abri qu'un jour une liste commence à blacklister n'importe quoi, pour une raison ou une autre. Donc méfiance. D'autant qu'avec du greylisting et un spam-assassin + Rules Emporium, vous êtes relativement peinard. Au besoin, ne mettez pas ces listes dans un premier temps. Perso, je n'en n'utilise plus.

7.4.2.2 Spamassassin et autres outils connexes

7.4.2.2.1 La base Spamassassin et Pyzor

Pour spamassassin en revanche, il faut le faire appeler dans « procmail ». Il faudra penser aussi à lui faire parcourir la liste des spams de tous vos utilisateurs afin d'affiner sensiblement ses critères de reconnaissance.

Je pars du principe que votre installation est simpliste : un petit serveur sans prétention avec quelques comptes :) Inutile d'installer un processus permanent de spamassassin (spamd/spamc). Lorsque postfix recevra un mail et l'enverra à procmail, procmail appellera alors spamassassin pour contrôler que ce n'est pas du spam. Le processus est gourmand, mais si vous ne recevez pas 100 mails à la seconde, ce sera très correct.

Installez a les 2 paquets spamassassin et pyzor, tout est maintenant en place, il ne reste maintenant qu'à indiquer à votre « ~/.procmailrc » (et à tous ceux de vos utilisateurs) que vous utilisez « spamassassin » et que si spamassassin s'affole, on place directement le message dans un répertoire « spam » de votre arborescence de mails. Pour ceci, modifiez vos « ~/.procmailrc » comme suit (chaque caractère est important, je ne détaille pas la syntaxe) :

```
monlinux:~# cat .procmailrc
# RAS sur les lignes suivantes :
PATH=$HOME/bin:/usr/bin:/usr/ucb:/bin:/usr/local/bin:.
MAILDIR=$HOME/Maildir/ # You'd better make sure it exists
DEFAULT=$MAILDIR
LOGFILE=$MAILDIR/log
LOCKFILE=$HOME/.lockmail
SHELL=/bin/sh
# Nouveautés :
:0fw: spamassassin.lock # 1ère directive
* < 256000 # si la taille du fichier est < 256 ko (les spams ne sont pas plus gros actuellement)
| spamassassin # alors passer le message à spamassassin pour analyse
:0: # deuxième directive : lue si spamassassin a gueulé
* ^X-Spam-Flag: YES # ajouter un en-tête et
.spam/ # placer le mail dans le sous-répertoire « spam » de mon Maildir
# attention : le "/" indique le format Maildir et le "." car c'est comme ça, cf. plus bas
#TEST
#:0:
#* ^From.blablabla*
#.mailinglist/
```

Il faut vous créer le répertoire « spam » en question. Passez par exemple par l'interface webmail ou votre client IMAP car il ne suffit pas de créer un répertoire « .spam » dans « Maildir ». Si vous le faites depuis votre client mail, ça créera l'arborescence dans les règles de l'art Notez qu'un répertoire « spam » est stocké « .spam » dans ~/Maildir. C'est comme ça.

Si vous constatez que spamassassin n'est pas au top, pensez à lui faire reconnaître le spam régulièrement (et le « ham » comme on dit aussi). Par exemple comme ceci :

```
sa-learn --ham ~/Maildir/cur/
sa-learn --spam ~/Maildir/.spam/cur/
```

7.4.2.2.2 spamassassin sur un gros serveur

Sur un serveur traitant beaucoup de mails, l'appel à spamassassin par chaque « .procmailrc » n'est pas très performant. Il faut alors utiliser le « daemon » « spamd », en le faisant appeler par postfix.

Il faut l'activer dans le fichier « /etc/default/spamassassin ». Je tâcherai de détailler cela plus tard.

7.4.2.2.3 Rulesemporium

Les choses ont évolué récemment sur la mise à jour de Rules Emporium. Donc lisez leur site pour comprendre comment ça fonctionne puis choisissez les règles qui vous conviennent.

Enfin, reportez-vous à mon article http://michauko.org/blog/2008/03/20/rulesdujour-vs-spamassassin/ pour faire mettre à jour les règles Rules Emporium par SpamAssassin directement. Un exemple est donné en commentaire.

Ainsi que cette adresse pour le plug-in d'analyse de spam caché dans des images-de-texte : http://michauko.org/blog/2007/12/18/spamassassin-32-et-imageinfo/

7.5 Sécurisation SASL

La bibliothèque SASL permet de demander une authentification par mot de passe sur un serveur SMTP. Il y a plusieurs méthodes d'authentification, on peut demander à un LDAP, créer des comptes de mails spécifiques etc. On peut aussi simplement demander à contrôler que l'identifiant/mot de passe soit un compte utilisateur Linux valide sur la machine. C'est ce qu'on va faire. C'est simple et efficace.

Récemment, les paquets SASL et postfix ont évolué et la syntaxe et le fonctionnement ont changé. Je coupe ce chapitre en deux, l'ancienne et la nouvelle version. Plus tard, je supprimerai l'ancienne.

Dans les deux cas, il faut installer tout ça en complément de postfix :

```
apt-get install libsasl2 libauthen-sasl-cyrus-perl sasl2-bin postfix-tls libsasl2-modules
```

Pour postfix-tls (cryptage des échanges mails), ce n'est pas obligatoire et pas encore décrit dans cette doc.

Identifiez la version de postfix et sasl que vous utilisez avec la commande :

```
dpkg -l *sasl* postfix* | egrep "^ii"
```

Soit vous êtes dans la nouvelle version (SASL 2.1.22 et postfix 2.3.4) :

```
ii libauthen-sasl-cyrus-perl 0.13-server-1 Perl extension for Cyrus SASL library
ii libauthen-sasl-perl 2.10-1 Authen::SASL - SASL Authentication framework
ii libsasl2 2.1.22.dfsg1-5 Authentication abstraction library
ii libsasl2-2 2.1.22.dfsg1-5 Authentication abstraction library
ii libsasl2-modules 2.1.22.dfsg1-5 Pluggable Authentication Modules for SASL
ii postfix 2.3.4-2 A high-performance mail transport agent
ii sasl2-bin 2.1.22.dfsg1-5 Administration programs for SASL users datab
```

Soit dans l'ancienne (SASL 2.1.19 et postfix 2.2.je.sais.plus).

7.5.1 Ancienne version

J'ai trouvé une documentation très claire sur ce site. Je remercie son auteur. Je reprends à ma sauce ses informations, avec sa permission. Vous trouverez sur son site les informations pour configurer votre client mail (Thunderbird par exemple) pour utiliser votre SMTP en envoi.

On doit indiquer le mécanisme d'authentification souhaité, via le fichier « /etc/default/saslauthd » :

```
monlinux:~# cat /etc/default/saslauthd
# This needs to be uncommented before saslauthd will be run automatically
START=yes # à faire en théorie une fois que tout fonctionne
# You must specify the authentication mechanisms you wish to use.
# This defaults to "pam" for PAM support, but may also include
# "shadow" or "sasldb", like this:
# MECHANISMS="pam shadow"
MECHANISMS="shadow" # on utilise le fichier shadow (les utilisateurs du système) comme base d'authentification
PARAMS="-r -m /var/spool/postfix/var/run/saslauthd"
```

Il faut créer le répertoire mentionné ci-dessus, avec les bonnes permissions :

```
mkdir -p /var/spool/postfix/var/run/saslauthd
chown postfix:sasl /var/spool/postfix/var/run/saslauthd
```

Et enfin, indiquer à postfix qu'on utilise SASL, et qu'il faut donc s'authentifier pour envoyer des mails.

Créez le fichier « /etc/postfix/sasl/smtpd.conf » :

```
pwcheck_method: saslauthd
#mech_list: digest-md5 cram-md5 plain login
mech_list: plain login
```

Modifiez donc les paramètres « smtpd_[recipient|sender]_restrictions » du fichier « /etc/postfix/main.cf » et ajoutez quelques autres directives :

```
smtpd_sasl_auth_enable = yes
smtpd_sasl_local_domain =
broken_sasl_auth_clients = yes
smtpd_recipient_restrictions = permit_mynetworks, permit_sasl_authenticated,
reject_unauth_destination
smtpd_sender_restrictions = ..... ajoutez permit_sasl_authenticated aussi
```

Le « permit_mynetworks » permettra toujours d'envoyer depuis votre réseau local sans authentification.

Enfin, (re)démarrer le service d'authentification SASL :

```
/etc/init.d/saslauthd start
```

Et voilou, tout devrait fonctionner. Pour tester, n'oubliez pas d'utiliser votre SMTP depuis l'extérieur afin de passer grâce à « permit_sasl_authenticated » et pas grâce à « permit_mynetworks ». Vous suivez ?

7.5.2 Nouvelle version

C'est très similaire à l'ancienne version. Mais si vous vous cantonnez à ça, vous aurez des problèmes d'authentification bien que tout-soit-bien-configuré (c'est du vécu, je n'avais pas vu le changement de version lors d'une migration de serveur). En fait, il faut faire comme pour l'ancienne version, en intégrant les différences suivantes :

```
monlinux:~# cat /etc/default/saslauthd
START=yes
MECHANISMS="pam"
MECH_OPTIONS=""
THREADS=5
OPTIONS="-r -c -m /var/spool/postfix/var/run/saslauthd"
```

La création du répertoire « /var/spool/postfix/var/run/saslauthd » reste à faire, les permissions changent il me semble mais c'est corrigé par le lancement du service :)

Le reste est à faire aussi et il faut effectuer les opérations suivantes :

```
adduser postfix sasl # pour ajouter l'utilisateur postfix dans le group sasl
# question de droits pour communiquer
ln -s /etc/postfix/sasl/smtpd.conf /usr/lib/sasl2/smtpd.conf # pour régler un pb dans postfix
# semble-t-il
```

7.6 Sujets connexes divers

7.6.1 Reverse DNS

Certains serveurs de mails vont vous rejeter si vous n'avez pas un reverse-DNS correct. Ceci évite quelques spammers en herbe qui monte une serveur un peu « cheap » comme nous venons de le faire. « AOL » contrôle votre reverse-DNS il me semble.

Le reverse-DNS, c'est la requête « inverse » de résolution de nom, celle qui dit que telle IP correspond à tel nom. En bref, ça met un peu plus de légalité et de traçabilité dans le propriétaire du nom de domaine et dans la confiance que l'on peut accorder au SMTP qui envoi quelque chose.

Si votre ip est X.Y.Z.T et que votre nom de domaine est mondomaine.org, il faut donc que les requêtes suivantes fonctionnent :

```
monlinux:/home/djk# host mondomaine.org
mondomaine.org A X.Y.Z.T
monlinux:/home/djk# host X.Y.Z.T
Name: mondomaine.org # et pas un truc genre vbo.....fbx.proxad.fr
# ou LTombouctou...abo.mamadoo.fr
Address: X.Y.Z.T
```

7.6.2 Ouvrez votre firewall

Ne pas oublier d'ouvrir les ports 25 (smtp), 143 (imap) et 80 (web pour le webmail).

7.6.3 Alias(es)

Cas d'école : vous avez votre site web http://mondomaine.org et vous voulez créer une adresse mail du genre infos@mondomaine.org. Plutôt que de créer un compte Linux « infos », on va ajouter un alias vers votre propre compte. Pour cela, éditez le fichier /etc/aliases en ajoutant une ligne du type :

```
infos: tartempion, super.dupont@qqpart.fr # tartempion étant un compte de votre Debian
```

Puis lancez la commande qui met à jour les alias :

```
postalias /etc/aliases
```

C'est tout.

7.6.4 MX record sur le DNS

Idéalement, il vous faudrait un enregistrement de type « MX » sur votre nom de domaine, exemple :

```
monlinux:~# host -t MX mondomaine.org
mondomaine.org MX 1 mail.mondomaine.org
```

```
mondomaine.org MX 5 backup.chezunpote.de.mondomaine.org
```

Je passe sur l'intérêt de la manipulation, documentez-vous un peu et régler votre DNS (ou celui de votre fournisseur de « mondomaine.org »).

7.6.5 Création d'un squelette pour les utilisateurs

Si vous ouvrez des comptes pour vos amis sur votre Debian, vous voudrez sûrement leur offrir directement un environnement paramétré permettant l'utilisation de votre fabuleux serveur-de-mails-sans-spam-qui-déchire(tm). Pour ne pas avoir à créer chez chacun un Maildir, un .procmailrc, un Maildir/.spam etc qui vont bien, vous pouvez utiliser les « squelettes ».

Créez donc une fois pour toutes un modèle de « home directory » type pour vos utilisateurs dans « /etc/skel » et lorsque vous créerez un compte, une copie de « skel » sera présente dans le « homedir » de l'utilisateur. Exemple ci-dessous d'un environnement mail reprenant tout ce qu'on vient de faire. Recréez à l'identique de ce que vous voyez ci-dessous et ça roulera :

```
monlinux:~# ls -al /etc/skel
total 40
drwxr-xr-x 5 root root 4096 2006-05-28 13:51 .
drwxr-xr-x 90 root root 4096 2006-07-31 11:51 ..
-rw-r--r-- 1 root root 35 2005-12-05 16:37 .bash_aliases
-rw-r--r-- 1 root root 220 2006-03-23 01:23 .bash_logout
-rw-r--r-- 1 root root 413 2005-07-06 09:42 .bash_profile
-rw-r--r-- 1 root root 2218 2006-04-18 12:03 .bashrc
drwx------ 6 root root 4096 2006-05-28 13:51 Maildir
-rw-r--r-- 1 root root 486 2006-05-28 13:56 .procmailrc
drwxr-xr-x 2 root root 4096 2005-10-19 16:53 public_html
```

Le fichier « .procmailrc » est exactement celui décrit dans les chapitres précédents.

« /etc/skel/Maildir » contient pour sa part :

```
monlinux:~# ls -al /etc/skel/Maildir/
total 28
drwx------ 6 root root 4096 2006-05-28 13:51 .
drwxr-xr-x 5 root root 4096 2006-05-28 13:51 ..
-rw-r--r-- 1 root root 47 2006-05-27 15:17 courierimapsubscribed
drwxr-xr-x 2 root root 4096 2006-05-28 13:51 cur
drwxr-xr-x 2 root root 4096 2006-05-28 13:51 new
drwxr-xr-x 5 root root 4096 2006-05-27 15:16 .spam
drwxr-xr-x 2 root root 4096 2006-05-28 13:51 tmp
monlinux:~# cat /etc/skel/Maildir/courierimapsubscribed
INBOX.Sent
INBOX.Trash
INBOX.Drafts
INBOX.spam # ce fichier indique quels sont les répertoires enregistrés dans votre base IMAP
# vu qu'on construit le contenu du squelette à la main, il faut créer ceci à la main
monlinux:~# ls -al /etc/skel/Maildir/.spam/
total 24
drwxr-xr-x 5 root root 4096 2006-05-27 15:16 .
drwx------ 6 root root 4096 2006-05-28 13:51 ..
-rw-r--r-- 1 root root 17 2006-05-27 15:16 courierimapacl
drwx------ 2 root root 4096 2006-05-27 15:16 cur
-rw------- 1 root root 0 2006-05-27 15:16 maildirfolder # fichier vide
drwx------ 2 root root 4096 2006-05-27 15:16 new
drwx------ 2 root root 4096 2006-05-27 15:16 tmp
```

```
monlinux:~# cat /etc/skel/Maildir/.spam/courierimapacl
owner aceilrstwx # me demandez pas :) ou lisez le man maildiracl
```

Les différents répertoires cur, new et tmp sont tous vides.

Respectez les permissions des fichiers généralement limitées au propriétaire (« root:root ») dans « /etc/skel ». Lorsque vous créerez un compte « toto », les permissions seront les mêmes, mais le propriétaire sera « toto:users ». Pigé ?

7.6.6 Les faux-positifs ?

J'ai choisi des règles « sans risque de faux-positifs », mais sait-on jamais. Si cela intervient au niveau de postgrey ou le RBL, le mail va passer à la trappe directement et vous n'aurez pas de soupçon à moins de lire vos logs de mails (voir chapitre sur « logwatch »).

Si l'erreur intervient au niveau de spamassassin, le faux-positif sera consultable dans le sous-répertoire de spam de votre boite aux lettres – pour peu que vous la parcouriez un peu de temps en temps.

7.7 Fichier « /etc/postfix/main.cf » finalisé (ou pas loin)

Avant que je n'oublie, pensez à l'outil « testsaslauthd » pour tester votre processus SASL. Ca vous évitera de devoir encoder des mots de passes en base64 et de faire un telnet mon_smtp 25.

Histoire d'être à peu près clair sur ce gros chapitre, je termine en vous donnant un fichier « etc/postfix/main.cf » assez complet, intégrant le RBL, procmail, le greylisting et le SASL :

```
myorigin = /etc/mailname # n'oubliez pas de contrôler le contenu !
smtpd_banner = $myhostname ESMTP $mail_name (Debian/GNU)
biff = no
append_dot_mydomain = no
myhostname = mail.mondomaine.org
mydomain = maison.loc
alias_maps = hash:/etc/aliases
alias_database = hash:/etc/aliases
mydestination = localhost, $myhostname, $mydomain, mondomaine.org, mail.mondomaine.org
#relayhost = # si besoin, genre smtp.free.fr
mynetworks = 127.0.0.0/8, 192.168.0.0.24 # si vous voulez accepter votre LAN sans
authentification
mailbox_size_limit = 0
recipient_delimiter = +
inet_interfaces = all
mailbox_command = /usr/bin/procmail
smtpd_sasl_auth_enable = yes
smtpd_sasl_local_domain =
broken_sasl_auth_clients = yes
smtpd_recipient_restrictions = permit_mynetworks, permit_sasl_authenticated,
reject_unauth_destination
smtpd_sender_restrictions = permit_sasl_authenticated
reject_unauth_pipelining
permit_mynetworks
reject_unauth_destination
reject_unknown_sender_domain
reject_rbl_client sbl-xbl.spamhaus.org
reject_rbl_client list.dsbl.org
reject_rbl_client smtp.dnsbl.sorbs.net
```

```
reject_rbl_client web.dnsbl.sorbs.net
check_policy_service inet:127.0.0.1:60000
```

8 Héberger et partager vos photos : Gallery2

Si vous avez un serveur Linux « tout le temps allumé », une bande passante honnête (en upload) et une furieuse envie d'utiliser une application digne de ce nom pour partager vos photos avec vos amis et belle-maman, installez Gallery2. C'est du bonheur en barre. Avec ça, ciao les sites ultra-minimalistes offerts par votre hébergeur ou vos albums statiques générés sous ACDSee.

On va installer Gallery2. C'est terrible. Ca demande une base de données, par exemple MySQL, un serveur web, par exemple Apache, quelques morceaux de PHP et c'est tout. On procèdera en plusieurs temps, l'installation MySQL d'abord, puis Apache et ses liens avec MySQL et le PHP, et enfin Gallery2 et quelques outils connexes.

Les sous-chapitres sur MySQL et Apache ont migré sur le blog. Celui de Gallery suivra, quand j'aurai refait des photos d'écrans à jour.

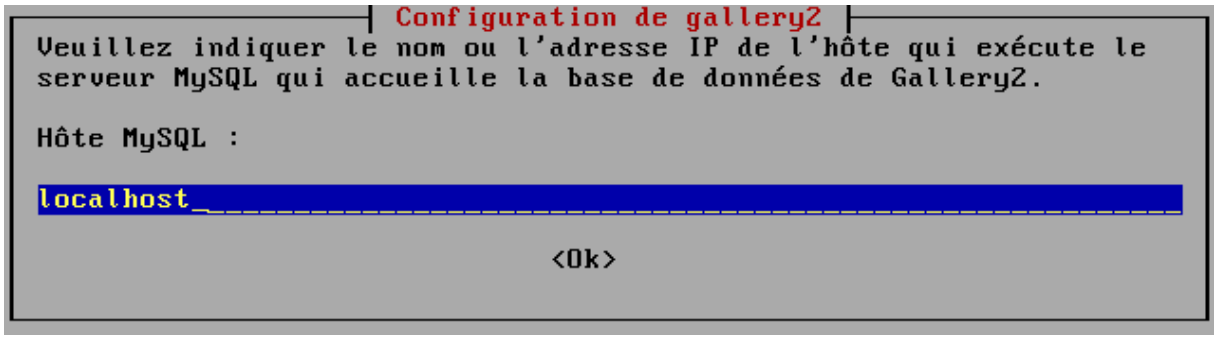
8.1 Installer Gallery2

Simplement : « apt-get install gallery2 jhead unzip dcraw ffmpeg »

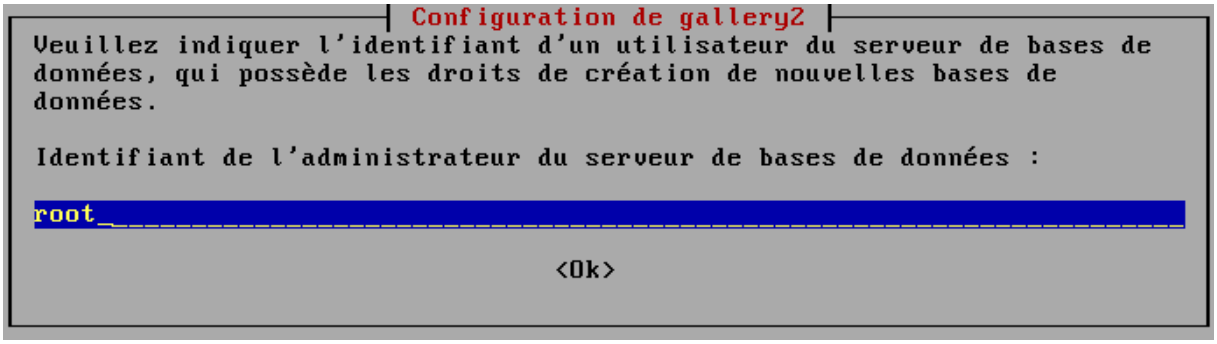
Quelques détails sur les paquets optionnels :

- jhead => Permet de manipuler les informations EXIF de vos photos numériques. Très pratique.
- unzip => Si des utilisateurs Gallery2 souhaitent déposer des photos par envois de zip, il faut pouvoir décompresser.
- Dcraw, ffmpeg => pour manipuler simplement les vidéos que vous pouvez poser dans Gallery2.

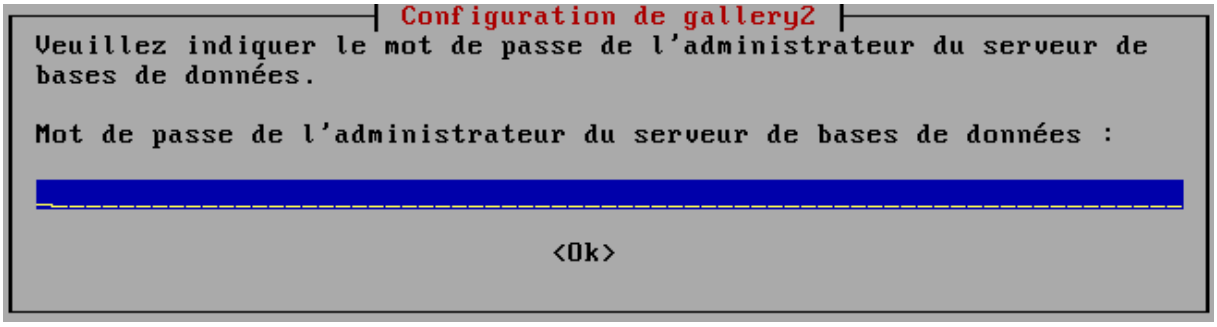
Laissez vous guider, ça devrait bien aller. Indiquez où est votre base de données, à priori sur la même machine, donc « localhost » :



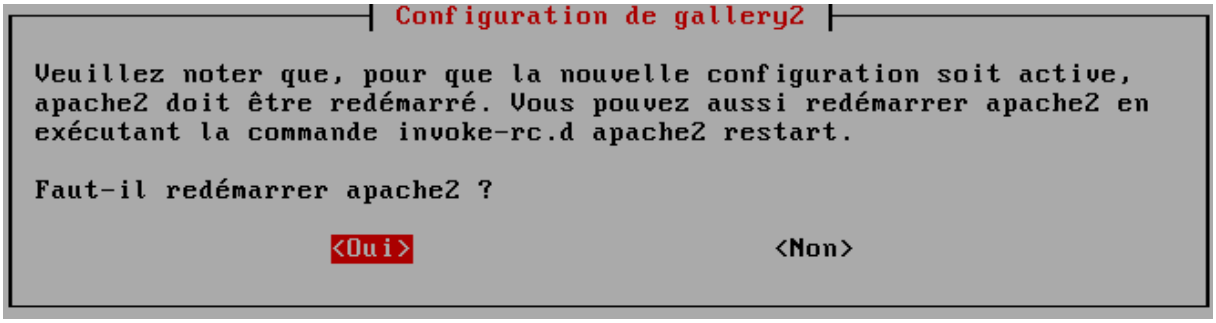
L'administrateur par défaut d'une base MySQL est « root » (rien à voir avec le « root » de l'OS, c'est le compte « root » propre à la base de données) :



Indiquez son mot de passe (si vous l'avez changé, je l'espère !) :



Enfin, il faut redémarrer Apache (ou Apache2) :



8.2 *Déclarer gallery2 dans apache*

La configuration est dans « /etc/apache2/conf.d ». Un fichier gallery2 (peu importe le nom) doit y être créé si ce n'est pas déjà fait avec le contenu suivant :

```
monlinux:/etc/apache2/conf.d# cat gallery2
Alias /gallery2 /usr/share/gallery2 # permet d'aller sur http://votre_serveur/gallery2/
<Directory /usr/share/gallery2>
Options FollowSymLinks
AllowOverride Limit Options FileInfo
</Directory>
```

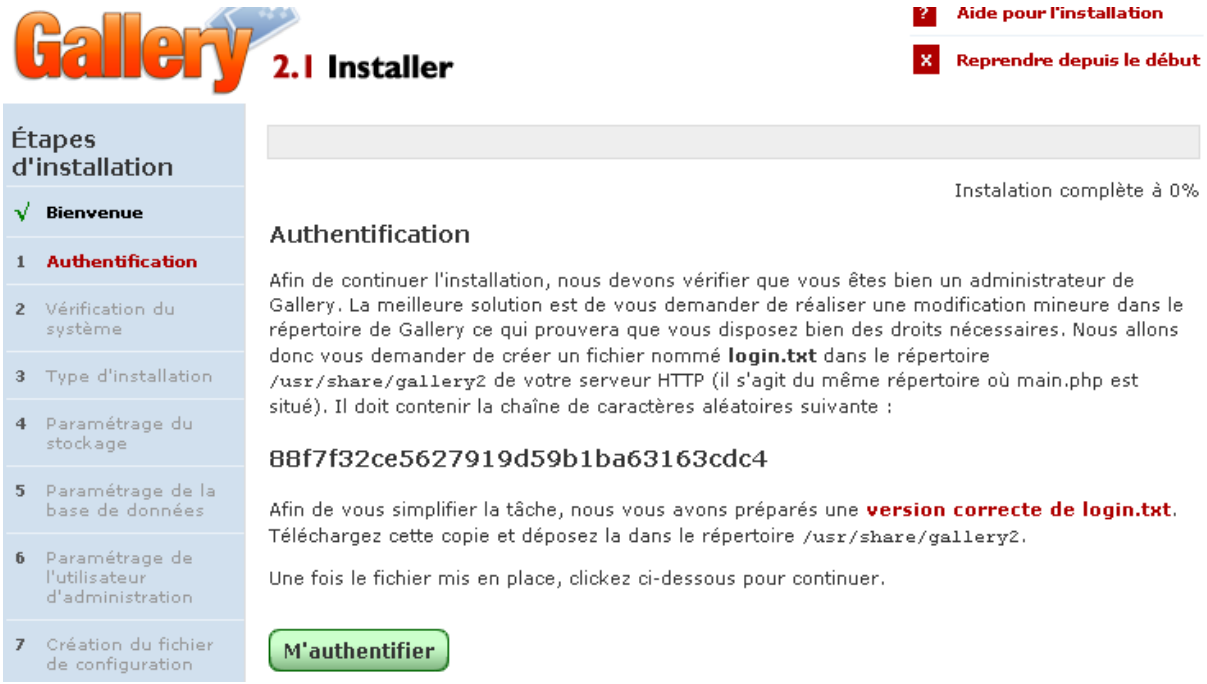
Relancez apache2.

```
/etc/init.d/apache2 restart
```

8.3 Configurer gallery2 par le web

Ensuite, il faut lancer la procédure d'installation Web de Gallery2 en vous connectant sur http://localhost/gallery2/install/ et en suivant les instructions :

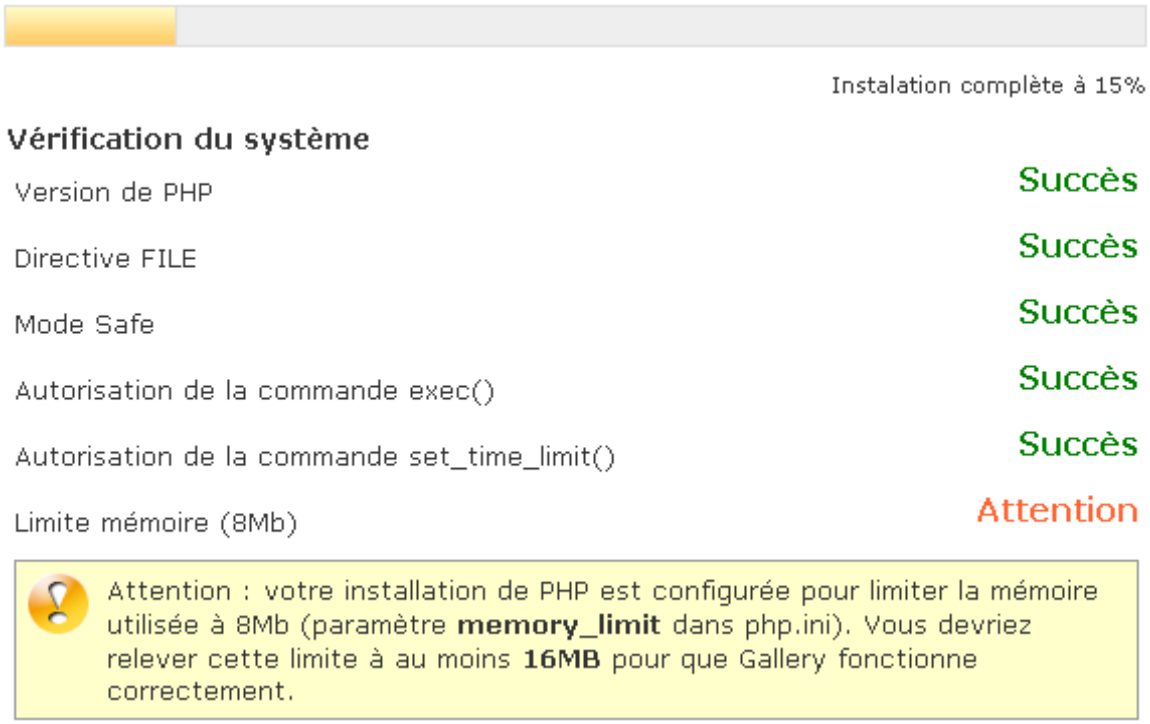
La suite est facile. Je décris l'étape suivante vaguement délicate :



Si votre serveur web était déjà ouvert sur Internet, n'importe qui pourrait tomber par hasard sur

votre site et donc lancer la configuration. Pour s'assurer que c'est bien vous, on vous demande d'injecter au niveau de l'OS (il faut être root) un fichier contenant une clef. Seul vous devriez être capable de le faire, ceci prouvant votre droit à configurer le bazar. Pigé ?

Donc, créez un fichier « login.txt » dans le répertoire « /usr/share/gallery2 » contenant la clef indiquée. Lors des étapes suivantes, les points notables sont :



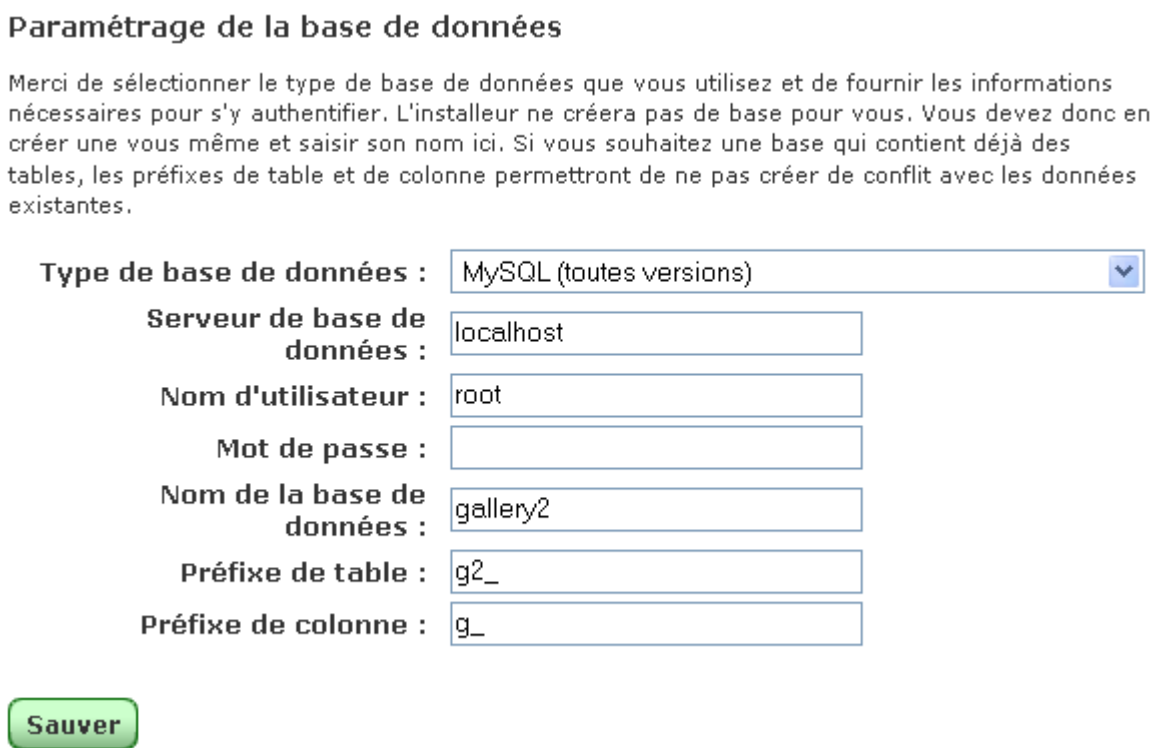
Pour augmenter la « limite mémoire », modifiez le fichier « /etc/php4/apache/php.ini » et changez le 8MB en 16MB pour la paramètre « memory_limit ».

Si vous utilisez apache2/php5, même problème, mais le fichier est dans « /etc/php5/apache2/ »

Ensuite :

Vous choisirez sûrement l'installation « standard », l'autre étant assez particulière.

Puis vient la connexion à la base de données :vous choisirez le nom du schéma de la base de données dans MySQL pour gallery2 :



Pas la peine de changer les valeurs à priori. N'oubliez pas le password du root MySQL.

Vers la fin, vous verrez tout un tas de plugin que vous pouvez activer dès maintenant ou plus tard. Faites votre choix, vous pourrez modifier par la suite.

Il vous reste maintenant à vous promener dans Gallery2, charger vos photos etc. Ca se paramètre dans tous les sens. Vous aurez peut-être besoin d'un serveur de mail opérationnel pour permettre aux utilisateurs de s'enregistrer en ligne et par échanges de mails. Je tâche d'ajouter ce chapitre dès que j'ai un moment (configuration postfix).

Pour plus tard, lorsque vous mettrez à jour votre système Debian avec « apt-get », vous aurez à vous souvenir du compte administrateur de la gallerie, ainsi que du compte root de la base MySQL. A chaque mise à jour de gallery2, vous devrez relancer une phase similaire à celle-ci, sauf que c'est une mise à jour et non une installation.

9 Sauvegarder votre machine Debian

Que vous ayez monté une Debian pour en faire un serveur Internet ou un environnement bureautique, vous aurez peut-être envie de sauvegarder de manière simple votre machine – au cas où... Quand je dis sauvegarder, c'est pouvoir réinstaller et re-paramétrer sans (trop) se casser la tête. Je ne parle pas de la sauvegarde de données personnelles, là vous saurez faire tout seul.

Il doit exister une tonne d'outils pour sauvegarder un système complètement. Je les trouve un peu obscurs ou compliqués, je préfère faire moi-même un script simple, quitte à perdre un peu de temps à la réinstallation, le jour où mon serveur va cramer ou que le disque aura un souci. Récemment j'ai migré d'un serveur perso à un serveur dédié grâce à ça.

J'ai créé un script root comme suit. Il sauvegarde tout dans un sous-répertoire « /root/backup ». Je le donne pour exemple :

```
monlinux:~# cat lance_backup.sh
#!/bin/sh
#utilisateurs dont je veux sauver les Maildir
UTILISATEURS="marcel dupont lajoie autre_login autre autre2 quidautre"
ERRORLOG="backup/error.log"
echo Efface l\'ancien backup
\rm -rf backup/*
echo Sauvegarde la liste des paquets
dpkg --get-selections > backup/liste_pkg
echo \/root sans le sous rep backup
tar -cjf backup/root.tar.bz2 --exclude 'backup' /root 2>>$ERRORLOG
echo \/etc
tar cjf backup/etc.tar.bz2 /etc 2>>$ERRORLOG
echo \/var\/log
tar -cjf backup/var.log.tar.bz2 /var/log 2>>$ERRORLOG
echo \/var\/www\*
tar -cjf backup/var.www.tar.bz2 /var/www* 2>>$ERRORLOG
echo \/var\/spool\/\* sans squid ni postfix
tar -cjf backup/var.spool.tar.bz2 --exclude 'squid' --exclude 'postfix' /var/spool
2>>$ERRORLOG
echo Compte tartempion sans certains répertoires gourmands
tar -cf backup/home.tartempion.tar --exclude 'gros_rep_volumineux' --exclude 'un_autre'
/home/djk 2>>$ERRORLOG
echo Sauvegarde des Maildir
for i in $UTILISATEURS
do
echo -e "\t$i"
tar -cjf backup/maildir."$i".tar.bz2 /home/"$i"/Maildir 2>>$ERRORLOG
done
echo Sauvegarde du \/home\/www \(racine de certains sites web\)
tar -cjf backup/home.www.tar.bz2 /home/www 2>>$ERRORLOG
echo Sauvegarde MySQL pour GALLERY
mysqldump --all-databases --lock-tables -u root -pmon_pass > backup/full_mysql.sql
2>>$ERRORLOG
bzip2 -9 backup/full_mysql.sql
echo Sauvegarde des fichiers de conf des HOME
tar -cjf backup/config_utilisateurs.tar.bz2 `find /home -name ".*" -type f | egrep -vi
"Maildir|authority|fake_windows"` 2>>$ERRORLOG
echo -n Taille du backup :
du -h backup
```

```
echo Enfin affichage du log d\'erreur \(sans certains messages connus\)
cat $ERRORLOG | grep -v "tar: Retrait de"
echo \<EOT\>
```

A la fin, dans le sous-répertoire « backup », vous aurez les principaux répertoires importants.

Le jour où il faudra réinstaller et re-paramétrer, je procéderai comme suit :

- Installation d'une Debian minimaliste (même version)
- Injection des paquets nécessaires :
- dpkg --set-selections < maliste
- apt-get update
- apt-get dselect-upgrade
- Recopie au compte-goutte des fichiers issus du backup de /etc (plutôt que de redescendre complètement /etc qui paraît complètement bourrin et voué à l'échec).
- Reconstruction des comptes utilisateurs, de leurs Maildir etc etc.

C'est pas très automatisé, mais ça fonctionne :) Enfin, il ne faut pas se louper lors de l'arrêt/relance de services (notamment postfix : ne pas récupérer des mails sans le voir puis écraser les boîtes aux lettres). Bref, ce n'est pas gagné quand même.

10 Outils à mentionner, références, autres distributions

10.1 Références

Il y en a beaucoup. J'aime particulièrement planet-libre.org qui agrège plusieurs blogs (dont le mien) et permet donc de connaître rapidement des sites de passionnés : d'actualité du libre, de configuration d'outils tordus (mais excellents) etc.

Ensuite, avec Google, vous irez partout où il faut.

10.2 Autres distributions

« Il n'y en a pas d'autres », comme diraient certains autour de moi. J'ai mentionné plusieurs fois Ubuntu. Je tiens à en mentionner deux autres, si votre but est de monter une passerelle Internet sécurisée en 3 clics (de clavier) :

- IPcop (http://www.ipcop.org/)
- Smoothwall (http://www.smoothwall.org/)

Ces deux distributions n'ont pour vocation que de proposer un système minimaliste – mais complet et sécurisé – pour rapidement mettre en place une passerelle Internet / firewall / DMZ / wifi / proxy / DHCP / j'en passe.

Ca s'installe en 10 minutes (normalement) et se configure entièrement graphiquement (http). Si vous n'avez pas besoin de plus, ne vous fatiguez pas avec Debian à galérer avec iptables, shorewall etc.

10.3 Connaître des paquets (Debian, Ubuntu)

En vrac, plein de paquets qu'on met parfois du temps à trouver (à soupçonner l'existence). N'hésitez pas à compléter ma liste.

A CONTINUER

11 Annexes

11.1 Installation en mode graphique

On le voit bien sur le premier écran d'installation d'une « Testing », le mode graphique est de plus en plus présent. Il est néanmoins assez inutile. Je fais l'impasse.

11.2 GNU Free Documentation License

Version 1.2, November 2002

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document »free » in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of »copyleft », which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The »Document », below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as »you ». You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A »Modified Version » of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A »Secondary Section » is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The »Invariant Sections » are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The »Cover Texts » are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A »Transparent » copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not »Transparent » is called »Opaque ».

Examples of suitable formats for Transparent copies include plain

ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The »Title Page » means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, »Title Page » means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section »Entitled XYZ » means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as »Acknowledgements », »Dedications », »Endorsements », or »History ».) To »Preserve the Title » of such a section when you modify the Document means that it remains a section »Entitled XYZ » according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified

Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
D. Preserve all the copyright notices of the Document.
E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
H. Include an unaltered copy of this License.
I. Preserve the section Entitled »History », Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled »History » in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the »History » section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
K. For any section Entitled »Acknowledgements » or »Dedications », Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
M. Delete any section Entitled »Endorsements ». Such a section may not be included in the Modified Version.
N. Do not retitle any existing section to be Entitled »Endorsements » or to conflict in title with any Invariant Section.
O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled »Endorsements », provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but

different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled »History » in the various original documents, forming one section Entitled »History »; likewise combine any sections Entitled »Acknowledgements », and any sections Entitled »Dedications ». You must delete all sections Entitled »Endorsements ».

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an »aggregate » if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled »Acknowledgements », »Dedications », or »History », the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License »or any later version » applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the

Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.