

# XP et Ubuntu

## une cohabitation efficace grâce à Colinux

Depuis plusieurs années, je suivais du coin de l'oeil les évolutions de *Linux* et, si j'étais depuis longtemps convaincu de l'intérêt de la plate forme *Linux* pour des serveurs ou des machines dédiées, je n'avais toujours pas trouvé de distribution qui me satisfasse pleinement comme environnement de travail quotidien.

Et puis, un jour, je suis tombé sur *Ubuntu*<sup>1</sup> et j'ai tout de suite adhéré à la philosophie de cette distribution: simplicité, ergonomie, une base *Debian*<sup>2</sup> et tout cela sur un seul CD.

J'ai bien vite redimensionné mon disque dur et installé *Ubuntu* en multi-boot avec mon PC initialement équipé de *XP*<sup>3</sup>.

Plus je travaillais sous *Linux*, je redécouvrais le plaisir d'un environnement riche, sûr et stable et m'ouvrant les portes de toute la bibliothèque des applications OpenSource.

Mais bien vite, un problème est apparu: le besoin de relancer sans cesse mon PC pour passer d'un environnement à l'autre représentait une réelle perte de temps. Je me suis alors tourné vers un autre produit OpenSource: *Qemu*<sup>4</sup>.

*Qemu* est un logiciel qui, à l'instar de *Vmware*<sup>5</sup>, émule un PC virtuel. J'ai pu ainsi faire tourner *Ubuntu* dans un fenêtre tout en restant sous *XP*. Seulement, même avec l'adjonction du module accélérateur *KQemu*<sup>6</sup>, le système émulé était trop lent pour être utilisé de façon réellement productive.

C'est alors, que je me suis souvenu d'une autre projet nommé *Colinux*<sup>7</sup>. D'une certaine façon, *Colinux* est un aussi émulateur de PC à la différence qu'il ne cherche pas à simuler le fonctionnement d'un PC complet (CPU, RAM, BIOS, Carte vidéo et carte réseau) mais traduit plutôt les appels du programme *Linux* en commande *XP*. Ainsi, les accès au matériel se font en natif d'où un gain de performance significatif.

En faisant tourner *Ubuntu* sous *XP* avec *Colinux*, j'ai enfin utiliser les deux systèmes d'exploitation en même temps et surtout sans écart de vitesse significatif entre les deux environnement.

Cette article à pour but de décrire et d'expliquer chacune des étapes qui ont été nécessaires pour adapter *Ubuntu* à l'architecture *Colinux*.

Dans une première partie, nous allons aborder l'installation de *Colinux* et la mise en route d'une *Debian* équipée de l'environnement graphique *KDE*<sup>8</sup>.

---

1 <http://www.ubuntu.com/>

2 <http://www.debian.org/index.fr.html>

3 <http://www.microsoft.com/france/windows/xp/>

4 <http://fabrice.bellard.free.fr/qemu/>

5 <http://www.vmware.com/>

6 <http://fabrice.bellard.free.fr/qemu/qemu-accel.html>

7 <http://www.colinux.org/>

8 <http://www.kde.org/>

Dans une seconde partie, nous aborderons les réglages spécifiques à *Ubuntu*.

## 1.Installation de Colinux

### 1.1.Environment de test

Toute l'installation décrite ci-dessous a été réalisée sur une station de travail PC équipé de Microsoft Window XP SP2. Elle est connectée à un réseau local filaire. La connexion internet est assurée par un routeur faisant office de serveur DHCP c'est à dire que chaque PC du réseau récupère automatiquement à chaque démarrage les réglages qui lui permette de se connecter à internet.

### 1.2.Téléchargement et installation

Le programme Colinux version 0.6.4 se télécharge depuis l'adresse «<http://www.colinux.org>». Il se compose d'un unique fichier exécutable.



L'installation ne pose aucune difficulté particulière.

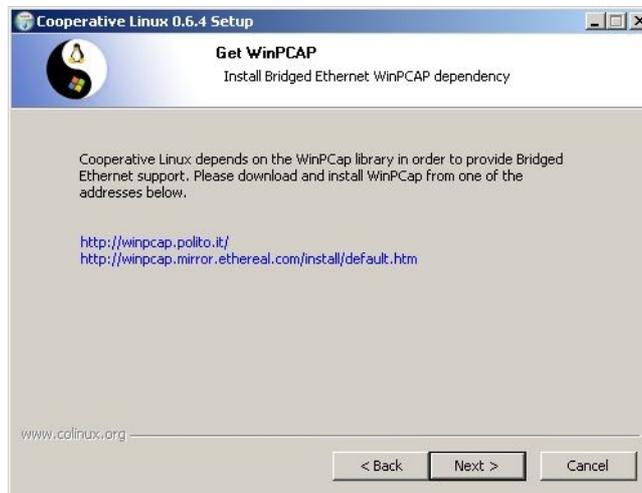
Après quelques écrans d'information, le programme propose les différentes options d'installation. On installe tous les composants même si *TAP-Win32* ne nous sera pas utile.



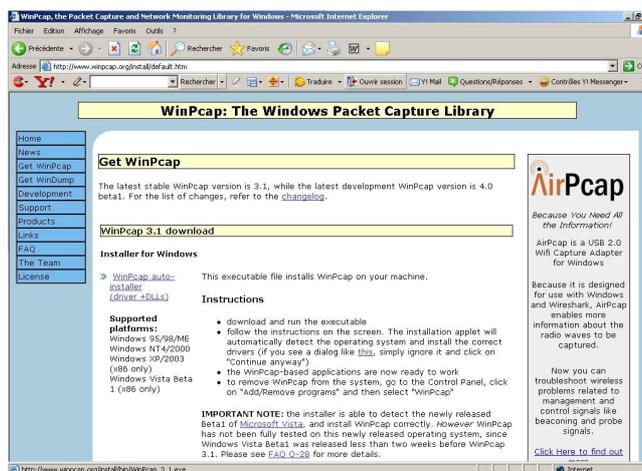
On change le répertoire d'installation pour <c:\colinux>.



L'écran suivant nous rappelle de télécharger et d'installer la dernière version de *WinPCAP*<sup>9</sup>.



*WinPCAP* installera les bibliothèques nécessaires au dialogue réseau entre *Colinux* et *XP* et le réseau LAN/WAN.



Enfin, le programme d'installation nous propose si on souhaite télécharger une image *Linux*. On choisit la distribution *Debian* ce qui nous permettra de valider notre installation

<sup>9</sup> <http://www.winpcap.org/>

avant d'abord *Ubuntu*.



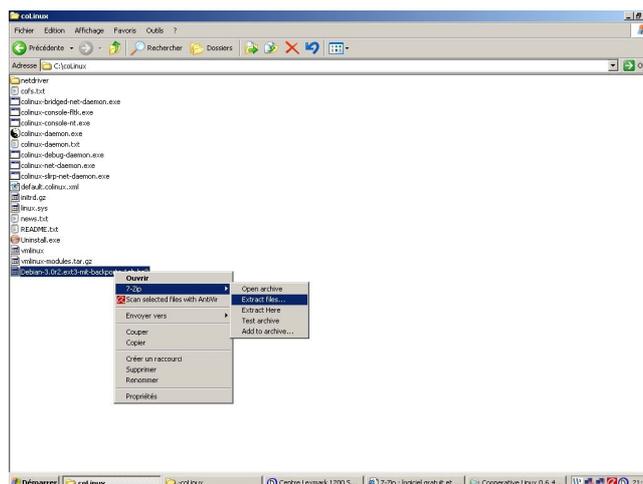
L'image *Debian* téléchargée pèse 28Mo mais se décompresse en un fichier de 1Go. *Colinux* ne gère pas les images disque compressées ce qui signifie qu'une système virtuel de 1Go correspondra à un fichier de 1Go sur une partition *Ms Windows*.

L'installation de *Colinux* est désormais terminée.

### 1.3.Paramétrage initiaux

#### a)Décompression de l'image Debian

Le programme d'installation de *Colinux* a rangé l'image *Debian* dans un fichier compressé nommé *Debian-3.0r2.ext3-mit-backports.1gb.bz2* stocké en *c:\colinux*. Nous allons la décompresser avec 7Zip<sup>10</sup>.

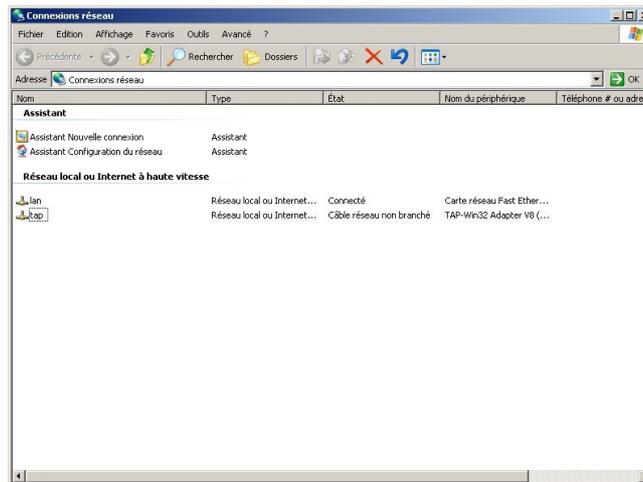


On renomme ensuite l'unique fichier extrait *debian.img* et on le recopie en <c:\colinux>.

<sup>10</sup> <http://www.7-zip.org/fr/>

## b) Configuration réseau

En renommant la connexion au réseau local de *XP* de *connexion au réseau local* en *lan* en évite un bug relatif aux noms de connexion contenant des caractères accentués<sup>11</sup>.



Le paramétrage de la connexion réseau de Colinux sur un PC équipé uniquement d'une carte WIFI est assez difficile essentiellement en raison des spécificités relatives à la sécurité des réseaux sans fils. Si nécessaire, il convient de se reporter au Wiki de <http://www.colinux.org/>.

## c) Création de fichier de configuration

Nous allons créer un fichier de configuration nommé *debian.xml* dont le contenu sera le suivant:

```
<?xml version="1.0" encoding="UTF-8">
<colinux>
  <block_device index="0" path="\DosDevices\c:\colinux\debian.img" enabled="true" />
  <bootparams>root=/dev/cobd0</bootparams>
  <image path="vmlinux" />
  <memory size="64" />
  <network index="0" type="bridged" name="lan"/>
</colinux>
```

Ce fichier est à placer en [c:\colinux](#).

On note que *debian.xml* défini:

- 64Mo de mémoire vive sont réservé pour *Colinux*  
<memory size="64" />
- le premier disque dur (*debian.img*) qui sera vu par *Colinux*.  
<block\_device index="0" path="\DosDevices\c:\colinux\debian.img" enabled="true" />
- le type de lien réseau entre *Colinux* et *XP*.  
<network index="0" type="bridged" name="lan"/>

<sup>11</sup> <http://wiki.colinux.org/wiki/RandomNotes>

Ainsi configurée, la carte réseau *XP* se verra attribuer deux adresses IP:

- une première appartenant au système *XP*
- une seconde attribuée au système Linux.

La carte réseau virtuelle et la carte réseau réelle seront vues comme deux matériels distincts et recevront chacune leurs paramètres automatiquement depuis le serveur DHCP. Linux pourra accéder directement au réseau local et à l'internet et les systèmes *Xp* et *Linux* pourront dialoguer entre eux.

### **d)Création de script de lancement**

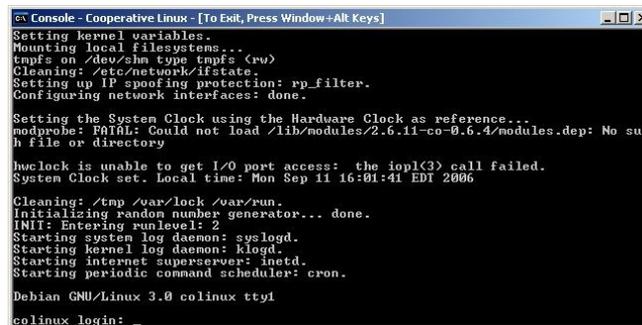
On crée un fichier nommé *debian.bat* dont le contenu listé ci-dessous.

```
colinux-daemon.exe -c debian.xml -t nt
```

*colinux-daemon.exe* désigne l'exécutable *Colinux*, *debian.xml* le fichier de configuration défini ci-dessus et le paramètre *-t nt* récite l'affichage de la console de contrôle de *Colinux*.

### **1.4.Premier lancement**

On est prêt pour le premier lancement. Pour cela, il reste plus qu'à double-cliquer sur le fichier *debian.bat* et après une courte séquence d'initialisation, le système *Debian* demande un identification de connexion et un mot de passe. Le *login* par défaut est « *root* » et le mot de passe associé est « *root* ».



```
Console - Cooperative Linux - [To Exit, Press Window+Alt Keys]
Setting kernel variables.
Mounting local filesystems...
tmpfs on /dev/shm type tmpfs (rw)
Cleaning: /etc/network/ifstate.
Setting up IP spoofing protection: rp_filter.
Configuring network interfaces: done.
Setting the System Clock using the Hardware Clock as reference...
modprobe: FATAL: Could not load /lib/modules/2.6.11-co-0.6.4/modules.dep: No such
file or directory
hwclock is unable to get I/O port access: the iopl(3) call failed.
System Clock set. Local time: Mon Sep 11 16:01:41 EDT 2006
Cleaning: /tmp /var/lock /var/run.
Initializing random number generator... done.
INIT: Entering runlevel: 2
Starting system log daemon: syslogd.
Starting kernel log daemon: klogd.
Starting internet superserver: inetd.
Starting periodic command scheduler: cron.
Debian GNU/Linux 3.0 colinux tty1
colinux login: _
```

### **1.5.Réglages de la session Linux**

A ce stade, le système Linux fonctionne parfaitement sous *XP*. Toutefois, il y a deux adaptations à apporter à l'image *Debian* :

- configurer le clavier en AZERTY Français,
- configurer la carte réseau pour activer le mode client DHCP.

#### **a) Configuration clavier**

Il faut tout d'abord changer la configuration de clavier. Pour cela, on lance la commande suivante:

```
dpkg-reconfigure console-data
```

Le clavier configuré d'origine sous l'image *Debian* est un clavier QUERTY. Aussi, pour obtenir cette commande, il faudra saisir `dpkg-reconfigure console-data`.

En validant le premier message d'accueil en tapant sur <Entrée>, on accède à une première page de choix. On sélectionne ensuite avec les flèches <haut>/<bas> le menu « choisir un codage clavier différent ». On valide ce choix avec la touche <Entrée>.

De la même manière, on choisit ensuite un clavier AZERTY, puis le modèle « french » et finalement « with euro (latin 9) ».



Désormais, *Debian* est configuré pour un clavier AZERTY.

## b) Configuration du réseau

On change ensuite la configuration réseau de l'image *Debian*. En effet, celle-ci est réglée avec une adresse IP fixe et nous allons configurer la carte réseau *Linux eth0* pour qu'elle se configure automatiquement avec les paramètres fournis par le serveur DHCP du réseau local.

La première étape consiste à couper la carte réseau en tapant la commande suivante:

```
ifdown eth0
```

On modifie ensuite le fichier de configuration `/etc/network/interfaces` avec l'éditeur de texte *nano*. On tape donc:

```
nano /etc/network/interfaces
```

On modifie la ligne débutant par `iface eth0` pour qu'elle devienne `iface eth0 inet dhcp`. On sauvegarde les modifications en tapant <CTRL>+<O> et on quitte *nano* en tapant <CTRL>+<X>.

```
GNU nano 1.0.6 File: /etc/network/interfaces
# /etc/network/interfaces -- configuration file for ifup(8), ifdown(8)
# The loopback interface
auto lo
iface lo inet loopback

# The first network card - this entry was created during the Debian installation
# (network, broadcast and gateway are optional)
auto eth0
iface eth0 inet dhcp
address 192.168.0.40
netmask 255.255.255.0
gateway 192.168.0.1

[ Read 13 lines ]
^G Get Help ^O WriteOut ^\ Replace ^V Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^R Read File ^W Where Is ^U Next Page ^U UnCut Txt ^T To Spell
```

Enfin, on réactive la carte réseau *eth0* en tapant:

```
ifup eth0
```

On peut vérifier la connexion à internet en tapant `ping www.google.fr`. La commande s'interrompt en tapant `<CTRL>+<C>`.

```
colinux:~# ping www.google.fr
PING www.google.com (64.233.183.99): 56 data bytes
64 bytes from 64.233.183.99: icmp_seq=0 ttl=242 time=77.6 ms
64 bytes from 64.233.183.99: icmp_seq=0 ttl=242 time=1015.3 ms <DUP!>
64 bytes from 64.233.183.99: icmp_seq=1 ttl=242 time=78.2 ms

--- www.google.com ping statistics ---
2 packets transmitted, 2 packets received, +1 duplicates, 0% packet loss
round-trip min/avg/max = 77.6/390.3/1015.3 ms
colinux:~# _
```

Puisque l'on dispose désormais d'une connexion internet, on en profite pour même *Debian* à jour en actualisant d'abord la base de données de mise à jour puis en appliquant d'éventuels correctifs. Ces deux actions se résument dans la commande suivante:

```
apt-get update && apt-get upgrade
```

## 1.6. Configuration de l'interface graphique

L'étape suivante consiste à installer une interface graphique à notre session *Linux*.

La grosse difficulté est que *Colinux* ne gère pas l'affichage graphique. Il ne suffit donc pas d'installer un serveur affichage graphique *X*<sup>12</sup> pour transformer la fenêtre de console en un bureau graphique, Nous allons donc devoir déporter l'affichage de l'écran graphique vers un logiciel de prise à distance en l'occurrence *VNC*<sup>13</sup>.

### a) Installation du serveur graphique Xfree

On demande l'installation de serveur *X* sous *Debian* en tapant:

```
apt-get install xserver-xfree86 xfonts-base xfonts-75dpi
```

### b) Installation du gestionnaire de fenêtre KDE

Tout aussi simplement, on demande l'installation d'un version de base du gestionnaire de fenêtre *KDE* en tapant:

```
apt-get install kdebase
```

### c) Installation du serveur d'accès distant

Finalement, on installe le serveur *VNC* en tapant:

```
apt-get install vncserver
```

Il reste alors plus qu'à lancer l'interface graphique en tapant:

```
vncserver -geometry 1024x768 -depth 16
```

Le paramètre `-geometry-depth 16 1024x768` précise la résolution graphique souhaitée et `-depth 16` le nombre de couleurs gérées (ici 16bits soit 65535 couleurs).

Au premier lancement, le serveur *VNC* demande de définir un mot de passe limitant l'accès à distance à la session *Linux*. Choisissons « *debian* ». *VNC* demande une second saisie du mot de passe pour la confirmation.

En apparence, rien n'a changé. Et pourtant...

## 1.7. Installation du client XP

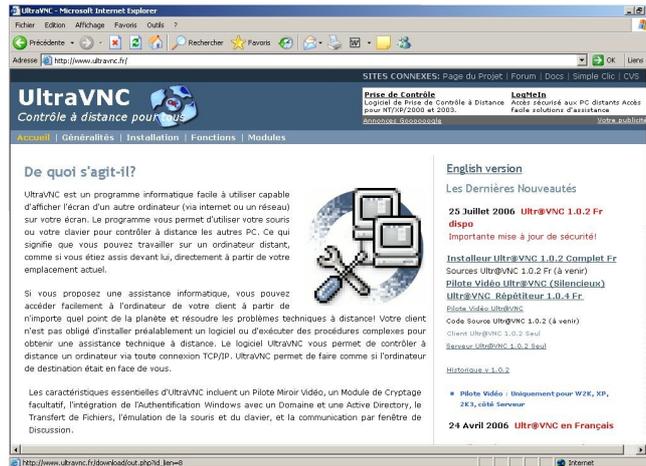
Maintenant, on télécharge un client *VNC* pour *Ms Windows* en l'occurrence *ultravnc*

---

12 <http://www.xfree86.org/>

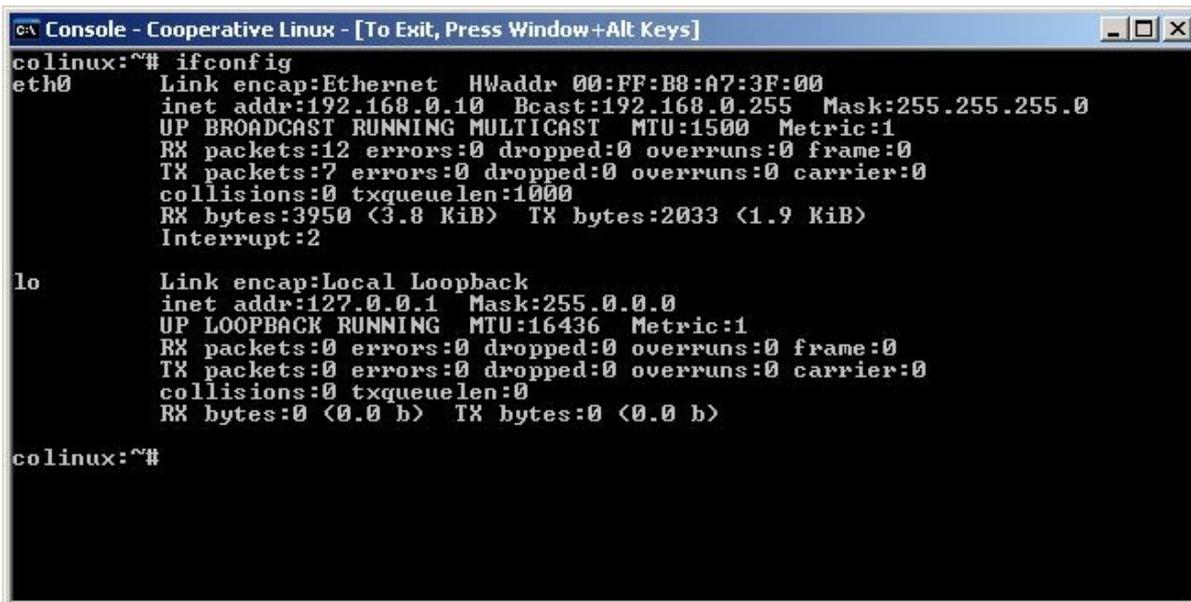
13 <http://en.wikipedia.org/wiki/VNC>

viewer<sup>14</sup>.



Le client VNC est un exécutable qui ne requière aucune installation. Il nécessite seulement qu'on lui indique l'adresse IP de la session *Linux* tournant sous *Colinux*.

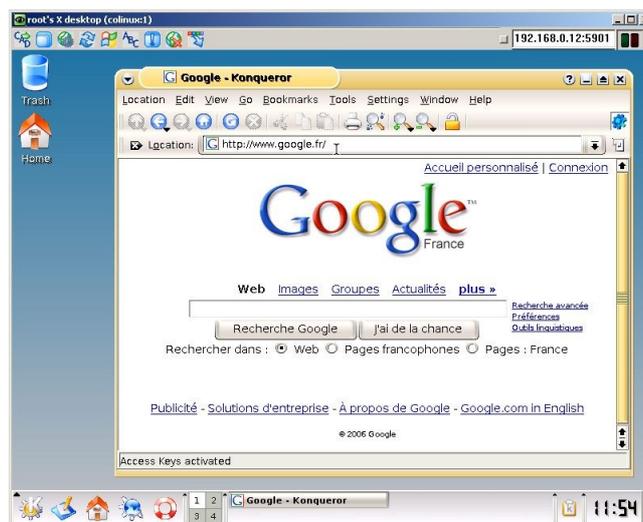
En tapant la commande `ifconfig eth0` dans la console *Debian* on récupère les informations de configuration de la carte réseau *eth0* et notamment son adresse IP. Dans l'illustration ci-dessous, l'adresse IP délivrée par le serveur DHCP est 192.168.0.10.



On reporte cette information dans le client VNC en ajoutant `:5901` à la fin.



Dès que l'on clique sur <Connect>, une nouvelle fenêtre apparaît. Elle nous montre le bureau KDE géré par la session *Debian* tournant sous *Colinux*.



## 1.8. Fermer la session Linux

On ferme facilement la session Linux et en termine ainsi le programme Colinux proprement en tapant la commande suivante:.

```
reboot
```

## 2.Migration vers Ubuntu

### 2.1. Agrandissement de l'image disque

Pour accueillir une distribution comme Ubuntu, il est clair qu'un disque virtuel de 1Go

comme celui que nous avons utilisé jusqu'à maintenant ne suffira pas.

Aussi, nous allons d'abord créer un disque virtuel de 3Go.

Windows Xp dispose d'une commande pour cela: *fsutil*.

```
fsutil file createnew c:\colinux\ubuntu_3g.img 3000000000
```

Un fichier nommé *ubuntu\_3g.img* d'une taille légèrement inférieure à 3 GigaOctets est ainsi créé dans le répertoire courant de Colinix.

Nous allons maintenant procéder à la copie complète du disque Debian sur le nouveau disque Ubuntu.

Tout d'abord, il convient de modifier le fichier de configuration de la machine virtuelle Colinix *debian.xml* afin de lui indiquer la présence de ce nouveau disque.

```
<?xml version="1.0" encoding="UTF-8">
<colinux>
  <block_device index="0" path="\DosDevices\c:\colinux\debian.img" enabled="true" />
  <block_device index="1" path="\DosDevices\c:\colinux\ubuntu_3g.img" enabled="true" />
  <bootparams>root=/dev/cobd0</bootparams>
  <image path="vmlinux" />
  <memory size="64" />
  <network index="0" type="bridged" name="lan"/>
</colinux>
```

On lance ensuite la machine virtuelle en double-cliquant sur *debian.bat*.

Une fois le système Linux démarré, nous allons partitionner le nouveau disque avec la commande *cfdisk*.

```
cfdisk /dev/cobd1
```

On notera que Colinix ne respecte pas les appellations traditionnelles pour les disques (hda,hdb... pour IDE, sda,sdb... pour SCSI).

Ici, les disques se nomment cobd1, cobd2...

*cfdisk* nous alerte que le disque ne contient pas de partition.

```
Console - Cooperative Linux - [To Exit, Press Window+Alt Keys]

No partition table or unknown signature on partition table
Do you wish to start with a zero table [y/N] ?
```

On continue en tapant 'y' puis on crée une nouvelle partition primaire en choisissant à l'aide de la flèche du clavier le menu « new » puis « primary » et en validant la taille par défaut.

Ensuite, on active l'option « bootable ».

Enfin, on enregistre la table de partition, en choisissant l'option « write ».

```
Console - Cooperative Linux - [To Exit, Press Window+Alt Keys]

cfdisk 2.11n

Disk Drive: /dev/cobd1
Size: 300000000 bytes
Heads: 255 Sectors per Track: 63 Cylinders: 364

Name      Flags      Part Type  FS Type   [Label]    Size (MB)
-----
cobd1p1  Boot      Primary   Linux

[Bootable] [ Delete ] [ Help ] [Maximize] [ Print ]
[ Quit ]   [ Type ]  [ Units ] [ Write ]

Write partition table to disk <this might destroy data>
```

Finalement, on quitte l'éditeur de partition en validant « quit ».

Nous allons maintenant pouvoir formater cette nouvelle partition en utilisant le système de fichier journalisé « ext3 ».

```
mke2fs -j /dev/cobd1
```

On crée ensuite un répertoire qui servira de point de montage temporaire à cette partition. Cette étape va nous permettre de recopier l'intégralité du disque principal *cobd0* vers *cobd1*.

```
mkdir /mnt/nvdisk
```

On procède au montage manuel de la partition avec la commande:

```
mount -t ext3 /dev/cobd1 /mnt/nvdisk
```

Il ne reste plus qu'à transférer le système du disque original vers le nouveau disque à l'aide de la commande suivante:

```
cp -ax / /mnt/nvdisk
```

Une fois, cette opération réalisée, on peut éteindre le système.

```
shutdown -h now
```

On efface toute trace du disque de 1 Go dans le fichier de configuration *debian.xml* et on indique *ubuntu\_3g.img* est le disque sur lequel on souhaite démarrer.

```
<?xml version="1.0" encoding="UTF-8">
<colinux>
  <block_device index="0" path="\DosDevices\c:\colinux\ubuntu_3g.img" enabled="true" />
  <bootparams>root=/dev/cobd0</bootparams>
  <image path="vmlinux" />
  <memory size="64" />
  <network index="0" type="bridged" name="lan"/>
</colinux>
```

## 2.2. Ajout d'une partition Swap

Nous allons maintenant créer un disque virtuel de 512M pour accueillir le fichier d'échange.

La commande ci-dessous crée un fichier nommé 'swap\_512.img' d'une taille légèrement inférieure à 512 MegaOctets.

```
fsutil file createnew swap_512.img 512000000
```

Nous allons de nouveau modifier le fichier de configuration *debian.xml* pour intégrer ce nouveau disque.

```
<?xml version="1.0" encoding="UTF-8">
<colinux>
  <block_device index="0" path="\DosDevices\c:\colinux\ubuntu_3g.img" enabled="true" />
  <block_device index="1" path="\DosDevices\c:\colinux\swap_512.img" enabled="true" />
  <bootparams>root=/dev/cobd0</bootparams>
  <image path="vmlinux" />
  <memory size="64" />
  <network index="0" type="bridged" name="lan"/>
</colinux>
```

On relance la machine virtuelle.

Une fois, le système Linux démarré, nous allons partitionner le nouveau disque avec la commande:

```
fdisk /dev/cobd1
```

On crée une nouvelle partition primaire en choisissant « new » puis « primary » et en validant la

taille par défaut.

On saisie la valeur '82' dans l'option «type» afin de préciser qu'il s'agit d'une partition de type 'swap'.

Enfin, on enregistre la table de partition, en choisissant l'option « write ».

Nous allons maintenant formater cette nouvelle partition en tapant la commande suivante:

```
mkswap /dev/cobd1
```

On édite le fichier des gestion des points de montage *fstab* afin d'y intégrer la partition de swap.

```
nano /etc/fstab
```

Le fichier doit ressembler à ceci:



```
GNU nano 1.0.6 File: /etc/fstab Modified
# /etc/fstab: static file system information.
#
# <file system> <mount point> <type> <options> <dump> <pass>
/dev/cobd0 / ext3 errors=remount-ro 0 1
/dev/cobd1 none swap sw 0 0
proc /proc proc defaults 0 0
tmpfs /dev/shm tmpfs defaults 0 0
```

On appuie sur <CTRL>+<O> pour enregistrer puis <CTRL>+<X> pour sortir de l'éditeur de texte *nano*.

Finalement, on active le swap en tapant:

```
swapon -a
```

On peut vérifier que le swap est bien activé avec la commande:

```
swapon -s
```



```
colinux:~# swapon -s
Filename      Type      Size      Used      Priority
/dev/cobd1   partition 499992    0         -1
colinux:~#
```

### **2.3. Changement de la source de paquet Debian**

Sous Debian, les serveurs internet d'où le système peut télécharger les fichiers et les programmes dont il a besoin sont listés dans le fichier `/etc/apt/sources.list`.

Nous allons remplacer les sources de paquets Debian par celles d'Ubuntu avec l'éditeur de texte nano.

```
nano /etc/apt/sources.list
```

On efface le contenu et le remplace avec les sources françaises d'une Ubuntu .

```
deb http://fr.archive.ubuntu.com/ubuntu/ breezy main restricted
deb-src http://fr.archive.ubuntu.com/ubuntu/ breezy main restricted
deb http://fr.archive.ubuntu.com/ubuntu/ breezy-updates main restricted
deb-src http://fr.archive.ubuntu.com/ubuntu/ breezy-updates main restricted
```

### **2.4. Migration vers Ubuntu**

La procédure de migration est très simple. On met d'abord à jour l'annuaire des paquets disponibles avec la commande.

```
apt-get update
```

On procède à la mise à jour à proprement parlé de la distribution avec la commande suivante:

```
apt-get dist-upgrade
```

Théoriquement, la migration devrait se passer sans problèmes mais il est fort probable que la commande génère des erreurs. La commande suivante devrait résoudre bien des soucis:

```
apt-get -o "APT::Force-loopbreak=true" dist-upgrade
```

Éventuellement, il faudra peut-être finir la mise à jour avec la commande suivante:

```
apt-get -o "APT::Force-loopbreak=true" -f install
```

## **3. Installation de Bureau graphique Ubuntu**

Désormais, notre Linux tournant sous Colinux est une distribution Ubuntu. Mais, comme la Debian d'origine, il ne contient d'une session texte et bien peu d'applications.

Nous allons demander le téléchargement et l'installation de l'ensemble des programmes du CD Ubuntu en une seule commande:

```
apt-get install ubuntu-desktop
```

Évidemment, cette opération prends un certain temps (environ 1heure 30 sur mon PC).

### 3.1. Déactivation de X au démarrage

Une distribution Ubuntu démarre automatiquement une session graphique GNOME. Comme Colinux ne sait pas gérer de console graphique, il faut immédiatement désactiver cette fonctionnalité.

C'est chose faite avec la commande suivante:

```
echo "null" > /etc/X11/default-display-manager
```

On désormais éteindre la session Ubuntu.

## 4. Configurer un accès en lecture/écriture entre à Xp et Ubuntu

On rajoute une ligne dans le fichier de configuration *debian.xml*. Celle-ci crée un nouveau disque virtuel *codb2* dans le contenu est le répertoire XP [c:\colinux](#).

```
<?xml version="1.0" encoding="UTF-8">
<colinux>
  <block_device index="0" path="\DosDevices\c:\colinux\ubuntu_3g.img" enabled="true" />
  <block_device index="1" path="\DosDevices\c:\colinux\swap_512.img" enabled="true" />
    <cofs_device index="0" path="\DosDevices\c:\coLinux\" type="flat" enabled="true" />
  <bootparams>root=/dev/cobd0</bootparams>
  <image path="vmlinux" />
  <memory size="64" />
  <network index="0" type="bridged" name="lan"/>
</colinux>
```

On relance le système Ubuntu en double-cliquant sur *debian.bat*.

Sous l'invite de commande, on crée un point de montage pour ce disque d'un genre un peu particulier:

```
mkdir /windows
```

Ce disque sera monté avec le commande suivante:

```
mount -t cofs 0 /windows
```

Désormais, tout ce que l'utilisateur Ubuntu écrira dans le dossier */windows* sera lisible par l'utilisateur XP et tout ce que l'utilisateur XP écrira dans [c:\colinux](#) sera lisible par l'utilisateur Ubuntu.

## 5. Accéder Ubuntu depuis XP

Nous allons utiliser le répertoire d'échange entre Ubuntu et Xp pour y télécharger les fichiers d'installation du serveur d'accès distant de NoMachine.

## 5.1. Installer et configurer NX server sous ubuntu

La société NoMachine<sup>15</sup> a développé et intégré des techniques de compression du protocole X. Ainsi, toutes les communications entre un PC local et un PC distant sont compressées pour plus de performance. On obtient un taux de compression variant de 1:10 à 1:100.

Il devient alors possible de travailler à distance sur un bureau Linux distant même en pleine écran et même avec une faible bande passante.

Le confort d'utilisation est sans commune mesure avec VNC.

Les interfaces graphiques sont distribuées sous une licence propriétaire, mais toutes les bibliothèques (compression, communication et les bibliothèques X modifiées) sont disponibles sous licence GPL.

Il existe également une version gratuite du serveur d'accès distant – limitée à une seule session. C'est ce module que nous allons utiliser pour établir un pont graphique entre Xp et Ubuntu sous Colinux.



On télécharge dans l'onglet « Download » du site de NoMachine les paquets pour Debian (DEB) des trois logiciels suivants:

- NX Free edition for Linux
- NX node
- NX client

Tous ces fichiers sont à sauvegarder dans [c:\colinux](#).

On procède ensuite à l'installation du serveur en appliquant successivement les quatre commandes ci-dessous:

```
dpkg -i /windows/nxclient_2.1.0-6_i386.deb
dpkg -i /windows/nxnode_2.1.0-7_i386.deb
dpkg -i /windows/nxserver_2.1.0-9_i386.deb
apt-get install ssh openssh-server
```

On crée un utilisateur *ubuntu* autorisé à prendre le contrôle à distance de Ubuntu avec la commande suivante:

---

<sup>15</sup> <http://www.nomachine.com/>

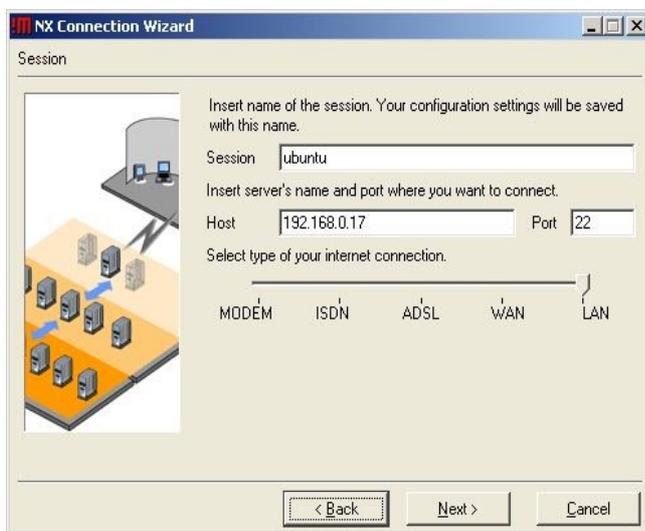
```
d/usr/NX/bin/nxserver --useradd ubuntu --system
```

Il faut choisir puis confirmer le mot de passe de cet utilisateur.

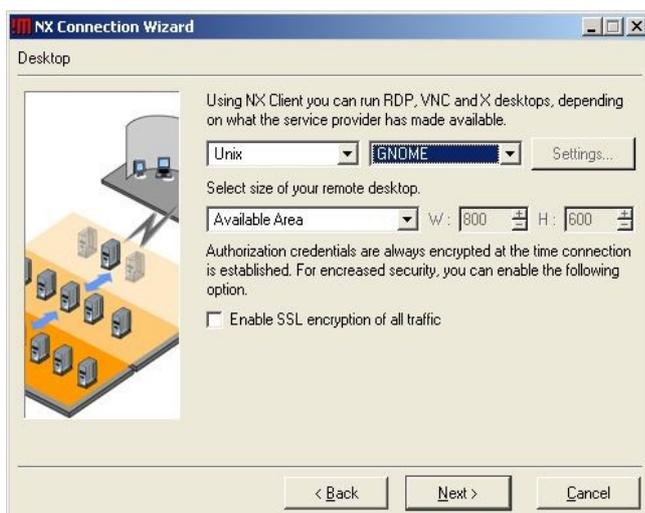
Il existe un projet dont l'objectif est de réaliser une implémentation totalement libre du serveur NX - FreeNX<sup>16</sup>

## 6. Installer NX client sous XP

Depuis le site de NoMachine, on télécharge maintenant le client pour Windows. L'installation ne présente aucune difficulté particulière et un assistant facilite le paramétrage du logiciel.



*ubuntu* est le nom que l'on souhaite donner à l'icône de raccourci et 192.168.0.17 est l'adresse IP du client *Colinux*.



On choisi de lancer GNOME, le bureau par défaut de Ubuntu.

<sup>16</sup> <http://doc.ubuntu-fr.org/serveur/freenx>

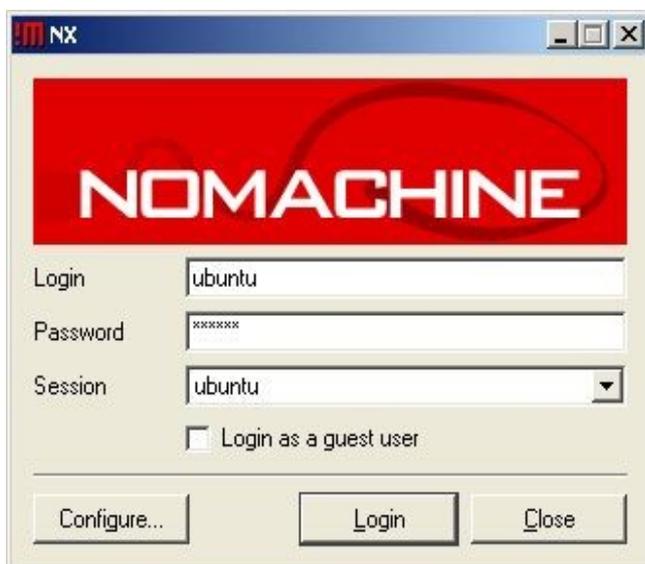
Il est à noter que l'on peut très bien utiliser le client NX pour lancer un autre gestionnaire de fenêtre que Gnome ou ne lancer qu'une seule application. Il faut alors choisir l'option « custom ».



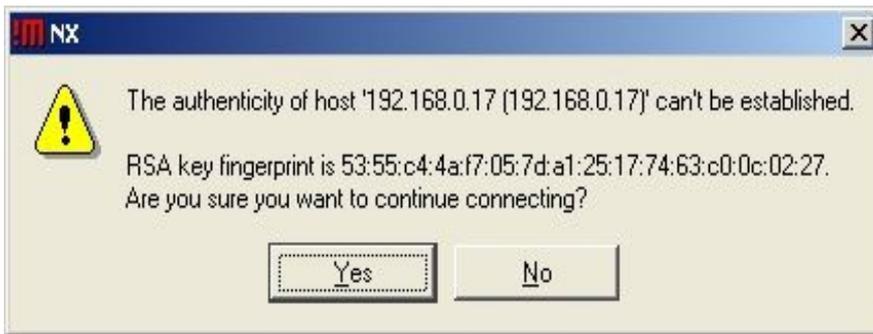
Le réglage du client Nx est terminé.

### **6.1. Connexion de Xp vers Ubuntu**

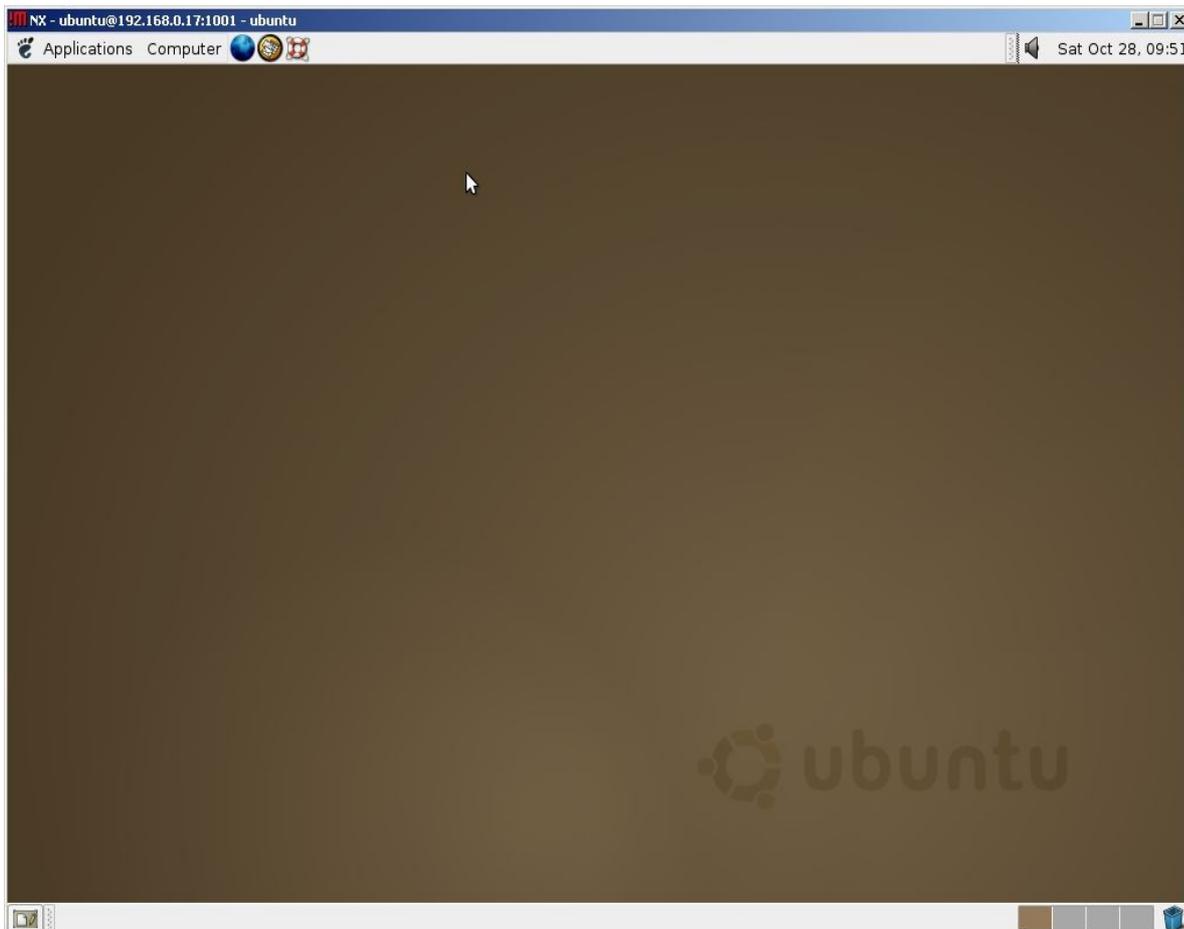
L'assistant d'installation du client NX a créé un raccourci sur le bureau de Xp. Lorsque l'on double-clique dessus, il faut tout d'abord indiquer le nom et le mot de passe de l'utilisateur ayant été autorisé à se connecter à distance.



Lors de la toute première connexion, le protocole SSH vous alerte sur l'absence d'une clé sécurisée sur le poste XP. On clique sur « yes » pour poursuivre la connexion.



Après quelques secondes, le bureau de Ubuntu apparaît dans une fenêtre.



## 7. Le son

Si votre carte son est reconnue par Colinux et si votre gestionnaire de fenêtre est Gnome, alors vous pourrez alors écouter le son des applications tournant sur Linux dans votre session XP. Il faut penser à cocher la case « enable multimédia support » dans l'onglet « services » du client NX.



## 8. Choisir une autre distribution que Ubuntu

Peut-être n'êtes vous pas aussi enthousiaste que moi au sujet d'Ubuntu ou il se peut que la procédure que je viens de vous décrire vous rebute. Sachez alors qu'il existe un projet nommé Colonist<sup>17</sup> dans le but est d'automatiser le processus d'installation dans une machine virtuelle Colinux de LiveCD des distributions suivantes:

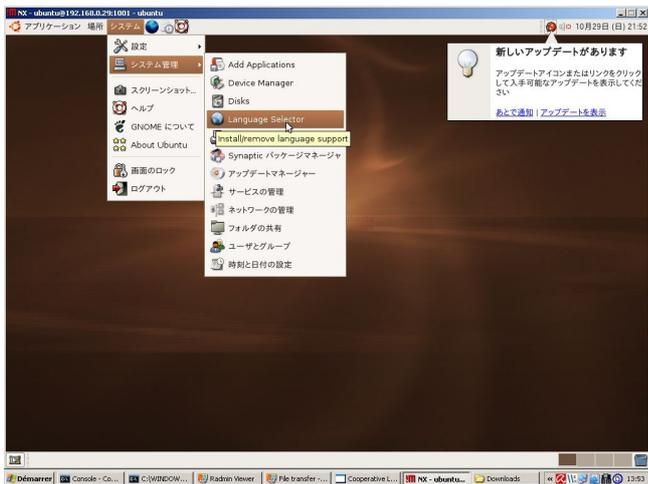
- Knoppix (Live CD) Debian 4.0.2
- Kubuntu Live Debian 5.10
- PCLinuxOS Mandriva Preview .92
- Slax Slackware 5.0.6
- Ubuntu Live Debian 5.10

On trouve même sur un site japonais<sup>18</sup> une distribution Ubuntu complète avec fichiers image et fichiers de configuration<sup>19</sup>. Le système démarre évidemment en japonais mais Linux permet le miracle de traduire tout un système d'une langue vers une autre.

<sup>17</sup> <http://colonist.sourceforge.net/>

<sup>18</sup> <http://www.ubuntulinux.jp/download/HowToUseCoLinuxImage>

<sup>19</sup> <http://archive.ubuntulinux.jp/releases/5.10/ubuntu-ja-5.10-colinux-i386-20051101.zip>



## 9. Conclusion

Nous voici à la conclusion de cet article avec désormais une session Ubuntu connectée au réseau Lan/Wan tournant en parallèle avec XP. Elle est équipée de l'interface graphique Gnome et l'on peut y accéder depuis n'importe quel PC du réseau local avec un client NX.

A titre d'information, la session texte *Ubuntu* ne met que 16 secondes à démarrer sur mon PC et une fois lancée, je peux utiliser *XP* ou *Linux* en même temps sans provoquer de ralentissements significatifs dans l'un ou l'autre des deux environnements.



### Olivier OLEJNICZAK

Partage son temps entre les responsabilités de :

- Directeur technique pour la SSII *Loginfo14.fr*
- Chef de projet informatique pour le PMI JLN *TaratataBijoux.com*

Installé à Caen sur un grand coup de coeur pour la ville et sa région

Diplômé d'informatique et électromécanique de l'U.T.C

Peut-être contacté sur *viaduc.com*

# Table des matières

1. Installation de Colinux.....	2
1.1. Environnement de test.....	2
1.2. Téléchargement et installation.....	2
1.3. Paramétrage initiaux.....	4
a) Décompression de l'image Debian.....	4
b) Configuration réseau.....	5
c) Création de fichier de configuration.....	5
d) Création de script de lancement.....	6
1.4. Premier lancement.....	6
1.5. Réglages de la session Linux.....	6
a) Configuration clavier.....	6
b) Configuration du réseau.....	7
1.6. Configuration de l'interface graphique.....	9
a) Installation du serveur graphique Xfree.....	9
b) Installation du gestionnaire de fenêtre KDE.....	9
c) Installation du serveur d'accès distant.....	9
1.7. Installation du client XP.....	9
1.8. Fermer la session Linux.....	11
2. Migration vers Ubuntu.....	11
2.1. Agrandissement de l'image disque.....	11
2.2. Ajout d'une partition Swap.....	14
2.3. Changement de la source de paquet Debian.....	16
2.4. Migration vers Ubuntu.....	16
3. Installation de Bureau graphique Ubuntu.....	16
3.1. Désactivation de X au démarrage.....	17
4. Configurer un accès en lecture/écriture entre à Xp et Ubuntu.....	17
5. Accéder Ubuntu depuis XP.....	17
5.1. Installer et configurer NX server sous ubuntu.....	18
6. Installer NX client sous XP.....	19
6.1. Connexion de Xp vers Ubuntu.....	20
7. Le son.....	21
8. Choisir une autre distribution que Ubuntu.....	22
9. Conclusion.....	23