Guide pratique du réseau sans-fil

Version française du *Wireless Howto*

Roberto Arcomano `<berto CHEZ bertolinux POINT com>`

Adaptation française: Clément Chaine

`<cchaine CHEZ dipsoft POINT com>`

Relecture de la version française: Vanessa Conchodon

`<nessie CHEZ little TIRET monster POINT org>`

Préparation de la publication de la v.f.: Jean-Philippe Guérard

`<jean TIRET philippe POINT guerard CHEZ laposte POINT net>`

Version : 1.7.fr.1.0

23 juillet 2003

Historique des versions

Version v1.7.fr.1.0	2003-07-23	CC,VC,JPG
Version v1.7	2002-08-23	RA

Le sans-fil est une nouvelle technologie de cartes réseau permettant des débits élevés (jusqu'à 11 Mbps) (NdT : aujourd'hui 22 Mbps, 54 Mbps et bientôt 100 Mbps). Ce document explique comment configurer un réseau sans-fil sous Linux, parle des problèmes de compatibilité, évoque les nécessités géographiques, et cætera.

Table des matières

1. Introduction

1.1. Introduction

Ce document donne quelques éléments sur les réseaux sans-fil, leur configuration, et leurs difficultés. Différent du réseau filaire, le sans-fil demande la connaissance de quelques astuces si l'on veut qu'il fonctionne bien. Vous devrez avoir quelques connaissances en matière d'antennes, savoir les positionner, avoir quelques connaissances en matière de configurations itinérantes, et cætera. Vous pourrez trouver une aide encore plus intéressante dans le *Wireless Howto* de Jean Tourrilhes [http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux] (ce document est en langue anglaise).

1.2. Commentaires et corrections

Merci de faire parvenir en anglais tous commentaires, suggestions, questions relatifs à ce document à l'adresse `<berto CHEZ bertolinux POINT com>`

N'hésitez pas à faire parvenir tous commentaires relatifs à la version française de ce document à `<commentaires CHEZ traduc POINT org>`

1.3. Nouvelles versions de ce document

Les nouvelles éditions de ce document sont disponibles sur http://www.bertolinux.com/ (en anglais).

Les nouvelles éditions françaises de ce document sont disponibles sur http://www.traduc.org/.

1.4. Copyright

1.5. Traductions

Vous êtes libre de traduire ce document, vous devez seulement :

1. Vérifier qu'aucune version de ce document n'existe déjà dans votre projet de documentation Linux (LDP) local.
2. Conserver la section « Introduction » dans son intégralité (y compris « Introduction », « Copyright », « Traductions », « Remerciements »)

Avertissement

Vous ne devez pas traduire la version texte ou la version HTML de ce fichier. Au contraire, votre traduction devrait partir de la version source au format Lyx, afin qu'il soit possible de la convertir dans tous les autres formats (texte, HTML, RIFF, et cætera) : pour ce faire, vous pouvez utiliser le logiciel « LyX », qu'il est possible de télécharger sur http://www.lyx.org.

Pour réaliser une traduction, nul besoin de demander mon autorisation ! Informez-moi simplement (si vous le souhaitez) de votre traduction.

Merci pour votre traduction !

1.6. Remerciements

Merci à Fatamorgana Computers [http://www.fatamorgana.com/] pour l'équipement matériel et l'expérience apportée.

Merci au Projet de documentation Linux [http://www.tldp.org/] pour la publication et la mise en ligne rapide de mon document.

2. Pré-requis

2.1. Qu'est ce que le sans-fil ?

Le sans-fil est une nouvelle technologie qui permet de connecter des ordinateurs à distance. Il fonctionne grâce à des cartes sans fil incluant un émetteur-récepteur à 2.4 GHz et présentant une interface logicielle semblable à Ethernet, avec une adresse matérielle unique pour chaque carte au monde. La puissance de transmission classique se situe entre 10-20 mW et 100 mW (voir les standards IEEE 802.11 et les autorisations de la FCC[1], de la CEPT[2] et de l'ART[3] pour la France).

2.2. Quelle est la distance maximum entre deux cartes radio ?

Le plus important dans les communications sans-fil est d'avoir une ligne à vue dégagée : vous *devez voir* (à l'œil nu ou avec des jumelles) la seconde antenne depuis la première, vous pouvez accepter (au pire) un petit arbre entre elles.

La distance dépend de l'antenne (éventuellement de l'amplificateur) utilisée : 200 à 300 mètres avec

1 *Federal Communications Commissions*, l'autorité réglementaire des télécoms aux États-Unis
2 Conférence Européenne des Administrations des Postes et Télécommunications
3 Autorité de régulation des télécoms

une antenne omnidirectionnelle; 1 km avec une antenne directionnelle; 2 à 3 km avec une antenne omnidirectionnelle amplifiée (200 mW); plusieurs kilomètres avec une antenne parabolique; 50 à 60 km avec une antenne parabolique ou directionnelle amplifiée (plusieurs Watts).

Soyez averti qu'il n'est pas toujours légal d'amplifier les cartes sans-fil (NdT : 100 mW maximum en France), car vous pourriez violer les règles de l'ART (en France — ou de votre autorité de régulation nationale, de la CEPT ou de la FCC).

2.3. Quelle est la différence entre un réseau filaire et un réseau sans fil ?

Les réseaux filaires sont simples à configurer (au moins à bas niveau). Les réseau sans-fil sont très difficiles à configurer, à gérer, à déboguer… Les problèmes classiques des réseaux filaires comme l'installation matérielle, l'installation logicielle, la mise au point, et cætera, deviennent vraiment cruciaux avec le sans-fil :

1. Vous devez choisir la bonne carte sans-fil : il existe de nombreuses cartes de différents fabricants, avec des spécifications et des conditions d'installation très diverses. Si vous voulez créer un petit réseau local, vous devrez acheter des cartes sans-fil et un point d'accès conformes à la norme IEEE 802.11.
2. Beaucoup de cartes sont PCMCIA, vous devez donc d'abord installer la gestion des ports PCMCIA sous Linux.
3. Vous devez effectuer des tests avec deux systèmes simultanément, d'abord à courte distance, puis vous pourrez aller plus loin.
4. Vous devrez effectuer vos tests par tous les temps (en général avec de la pluie).
5. Finalement, vous vous réjouirez d'avoir réussi la configuration.

Si vous avez installé un répéteur (une machine sous Linux qui contient plusieurs cartes filaires et sans-fil) vous risquez d'avoir des difficultés à modifier sa configuration à distance !

2.4. Qu'ai-je besoin de savoir pour configurer un réseau sans-fil ?

Des connaissances diverses sont nécessaire à la configuration d'un réseau sans-fil :

Connaissances logicielles

1. Notions de base de réseau : adresses IP, masque de sous-réseau, passerelle… Ces notions sont abordées dans le « Guide pratique de mise en réseau sous Linux » (*Linux Networking HOWTO*)4.
2. Connaissances réseau spécifiques4 : mandataire ARP (*proxy arp*, ponts, système de fichier `/proc` abordés dans les petits guides *Proxy-ARP-Subnet Mini-Howto* et *Bridge Mini-Howto* ainsi que dans le fichier `Documentation/Networking/ip-sysctl.txt` livré avec les sources du noyau Linux (2.2.x ou 2.4.x).
3. Connaissances des réseaux sans-fil : de leurs modes d'accès (Ad-hoc, Infrastructure, et points d'accès), du concept de canaux, de la définition d'extérieur et d'intérieur, et ainsi de suite, renseignements que vous pourrez trouver dans n'importe quel document à propos du sans-fil : standards IEEE 802.11, CEPT, et cætera.

4Tous les guides pratiques cités dans ce document peuvent être récupérés en version française sur http://fr.tldp.org ou sur http://www.traduc.org et en version originale sur http://www.tldp.org.

Connaissances non-logicielles

1. Une expérience minimum des antennes, de leur installation physique et de la façon de les pointer.
2. L'installation matérielle d'un PC en prenant un soin particulier de ne pas produire d'interférences entre les différentes cartes sans-fil (si nécessaire).

Et surtout, pas mal de chance !

2.5. Pourquoi devrais-je installer un réseau sans-fil, et qu'en attendre ?

Pourquoi ? Par ce que vous n'êtes pas satisfait par les réseaux filaires.

Les cartes sans-fil permettent de passer par-delà les jardins, les parcs, les maisons (mais vous *devez voir* l'autre côté)…

Les protocoles haut niveau utilisés par les cartes sans-fil sont les mêmes que ceux utilisés par les cartes Ethernet : TCP/IP sur du sans-fil est en gros semblable à TCP/IP sur de l'Ethernet. Cependant, faites attention aux partages réseau de Windows si vous utilisez Linux pour leur routage : en effet, vous devez savoir qu'un routeur IP ne laissera pas passer les messages de diffusion (voir le protocole NetBIOS) : dans ce cas, vous devrez utiliser un serveur WINS pour permettre l'exploration du réseau (voir la documentation de Samba).

Le sans-fil vous permet de créer un petit réseau local avec un point d'accès central (et peut-être un partage de connexion Internet) et d'en ouvrir l'accès à tout le monde par les ondes !

Imaginez un pays entièrement connecté par un réseau radio.

Imaginez un réseau qui permettrait de connecter tous les habitants d'un pays ensembles, de partager des fichiers, des applications audio, des applications vidéo nécessitant une large bande passante (comme le réseau câblé)…

Tout cela peut être réalisé (et ça l'est déjà dans certains pays) en utilisant des cartes sans-fil, des point d'accès sans-fil et des machines Linux sans-fil qui peuvent agir en tant que répéteur (à un niveau IP comme un routeur, ou si vous voulez, à un niveau liaison de données, avec un pilote pour pont.

2.6. Quelles sont les cartes sans-fil couvertes par ce guide pratique ?

Dans ce guide pratique, je commence avec une configuration générique — comme introduction au réseau sans-fil — puis je décris un exemple pour chacune des cartes que je connais directement, avec quelques astuces que vous pourrez utilisez pour améliorer leurs performances.

Liste des cartes sans-fil

1. Proxim Symphony — http://www.proxim.com
2. Webgear AviatorPRO 2.4 (nécessite un support PCMCIA) — http://www.webgear.com (NdT : n'existe plus)
3. Lucent Wavelan I, II, Orinoco — http://www.lucent.com et http://www.orinocowireless.com (activitée rachetée par Proxim)

4. Cabletron — http://www.enterasys.com/fr/ (NdT : Cabletron s'est recentré sur le marché des entreprises, et se nomme désormais Enterasys Networks)
5. YDI am930_isa — http://www.ydi.com
6. Modem Radio Siemens (Dect) — http://www.siemens.com
7. RadioLan (5 GHZ) — http://www.radiolan.com

Pour une liste plus exhaustive, voir le *Wireless Howto* de Jean Tourrilhes [http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux]

Le modem radio Siemens n'est pas vraiment une carte sans-fil 802.11, il s'agit en fait d'un modem que vous pouvez brancher sur le port série et qui fonctionne comme un modem (à 1800 MHz, technologie DECT). Son utilisation est décrite plus loin (cf. Annexe B, *Le Modem Radio Siemens DECT*).

Les cartes RadioLan fonctionnent à 5.4 GHz dans un environnement Windows 9x. Elle utilisent un point d'accès RadioLan qui sert de pont entre un réseau filaire et un réseau sans-fil. Il n'existe pas de pilotes Linux à ma connaissance.

2.7. Combien ça coûte ?

Les cartes sans-fil de la liste ci-dessus sont très chères : elles commencent à plusieurs centaines de dollars et dépassent le millier de dollars pour les points d'accès qui possèdent 2 cartes sans-fil (Lucent par exemple) et qui peuvent être utilisées comme pont (NdT : les prix ont beaucoup baissé depuis).

3. Informations techniques sur le sans-fil

Ici, je présentes quelques informations techniques pour mieux comprendre l'environnement sans-fil de base.

3.1. Couche physique

Au premier niveau du modèle OSI, nous pouvons avoir 3 types de spécifications :

1. FHSS : Étalement de spectre avec sauts de fréquence (*Frenquency Hopping Spread Spectrum*)
2. DSSS : Étalement de spectre en séquence directe (*Direct Sequence Spread Spectrum*)
3. Les connexions infrarouges, qui ne sont pas traitées dans ce guide.

3.2. Configurations

Les 2 types de configuration :

1. Le mode *Ad-hoc* (aussi appelé mode indépendant), où l'on a deux réseaux indépendants avec chacun son propre Ensemble de services de base (BSS — *Basic Service Set*). Chaque poste de travail a le même BSS.
2. Le mode *Infrastructure*, où plusieurs réseaux (avec chacun un BSS) peuvent communiquer entre eux par le biais d'un point d'accès (un pour chaque BSS) pour former un Ensemble de services étendus (ESS — *Extended Service Set*). Il y a aussi une fonction d'itinérance (*roaming*) qui permet à chaque station de se connecter au point d'accès le plus proche.

Ad-hoc est la méthode la plus simple (mais qui peut difficilement être utilisée à grande échelle) et permet à de nombreux hôtes de communiquer directement ensemble. La principale restriction de ce mode est que, pour pouvoir atteindre la totalité du réseau, il est nécessaire que chacun soit visible de tous les autres (au moins idéalement, car ce problème pourrait être résolu au niveau IP ! Pour en savoir plus, voir la Section 5.4, « Configuration IP »).

```
Mode Ad-hoc

                               A - - - - - - C
                                 \         /
                               |   \     /   |
                                     /\
                               |   /    \    |
                                 /        \
                               B - - - - - - D
```

Dans un environnement en mode Infrastructure, on utilise un point d'accès auquel *tous* les autres hôtes doivent se connecter pour partager le réseau.

```
Mode Infrastructure
                                     ESS

          A - - - | - Point d'accès - - Point d'accès - | - - - D

          B - - - |   BSS1                   BSS2       | - - - E

          C - - - |                                     | - - - F
```

B et C ne peuvent pas voir D, E et F, mais ils peuvent communiquer car ils utilisent tous le même ESS. Important : A, B et C pourraient aussi ne pas se voir.

De plus, on utilise les termes de intérieur (*indoor*) et extérieur (*outdoor*) pour distinguer une couverture de petite taille d'une grande couverture.

3.3. Compatibilité

Gardez bien en tête le fait qu'il existe beaucoup de cartes sans-fil dans le monde, mais que toutes ces cartes ne sont pas forcément capables de communiquer entre elles. Pour être capable de se parler, deux cartes doivent utiliser :

1. le même mode de configuration : tout en Ad-hoc ou tout en Infrastructure ;
2. la même couche physique : tout en DSSS ou tout en FHSS ;
3. le même protocole (par exemple, Proxim possède son propre protocole propriétaire, OpenAir, qui ne peut pas communiquer avec les autres cartes FHSS).

3.4. Dois-je utiliser Ad-hoc ou Infrastructure ?

Les points d'accès sont très pratiques et réduisent les problèmes, mais sont chers. Idéalement, pour un gros réseau, vous fonctionnerez en mode Infrastructure, alors qu'avec un petit nombre d'hôtes, vous choisirez plutôt le mode Ad-hoc : pourquoi dépenser beaucoup d'argent pour quelques hôtes ?

De toutes façons, sachez que si vous dépensez plus d'argent, tout fonctionnera probablement bien, alors qu'en dépensant moins, vous ferez probablement face à plus de problèmes.

3.5. Une machine Linux ne peut pas fonctionner

comme point d'accès ?

Bonne question !

Depuis peu, il existe une solution matérielle à ce problème : Prism2

Vous pourrez en apprendre plus sur http://hostap.epitest.fi/

4. La « caisse à outils »

4.1. Matériel nécessaire

Vous aurez besoin d'une machine Linux (un 486 ou mieux, au moins un Pentium 100, avec un minimum de 16 Mo de RAM), une cartes réseau sans-fil, une antenne (voir la Section 2.2, « Quelle est la distance maximum entre deux cartes radio ? »). Vous devrez avoir la même chose de l'autre côté (avec Windows 9x ou Windows NT si vous préférez) car vous devrez simuler une communication !

4.2. Logiciels nécessaires

Vous aurez besoin :

1. des sources d'un noyau récent et stable (2.2.x au moins) ;
2. de sources stables du pilote pcmcia (pcmcia-cs) si vous avez acheté une carte PCMCIA ;
3. de pilotes réseau sans-fil : si vous n'en avez pas vous, pourrez les télécharger depuis le site web du revendeur ou du constructeur de la carte. Si vous ne trouvez pas, vous pourrez consulter le *Wireless Howto* de Jean Tourrilhes [http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux].

Si vous ne les trouvez pas, même là, vous n'aurez plus qu'à attendre ou à convertir un pilote pour Windows en pilote pour Linux !! (Bonne chance !).

Après cela, vous devrez recompiler votre noyau, recompiler vos sources pcmcia (si vous en avez besoin pour votre carte sans-fil), et enfin recompiler votre pilote sans-fil. Ceci est la situation classique, mais pour certaines cartes, vous n'aurez peut-être a effectuer que l'étape 3 ou seulement les étapes 1 et 3, cela dépend des pilotes.

5. Vue d'ensemble de la configuration d'un réseau sans-fil

5.1. Étapes fondamentales

Une fois que vous avez le matériel nécessaire et que vous aurez compilé tout ce dont vous avez besoin, vous devrez effectuer les étapes fondamentales de configuration d'un réseau sans-fil :

1. Configuration bas-niveau du noyau de Linux. Cela lui permettra de voir votre carte sans-fil (à un bas-niveau, soit par exemple : ports E/S, interruptions, dma…) : vous devrez voir apparaître un message quelconque qui vous prévienne que la carte sans-fil a bien été trouvée et configurée.
2. Configuration au niveau liaison de données. Pour chaque carte sans fil, il existe un utilitaire qui permet de paramétrer les valeurs classiques du niveau liaison de données pour le sans-fil. Par exemple, pour la Proxim Symphony, l'utilitaire s'appelle **proxcfg** alors que les paramètres des cartes PCMCIA se trouve dans les fichiers de configuration PCMCIA. Vous devrez configurer toutes vos cartes sans-fil de manière cohérente si vous voulez qu'elle puissent fonctionner en-

semble.

3. Configuration IP. Arrivé à cette étape, vous devriez pouvoir utiliser les commandes **ifconfig** et **route** pour modifier les paramètres IP.

4. Utiliser quelques astuces améliorer les performances et éviter les conflits. Les éléments de base de votre réseau fonctionnent désormais correctement : en complément, vous devrez ajuster quelques paramètres spécifiques, tels que le mandataire ARM (*proxy arp*, la redirection de l'écho ICMP, le pontage, le changement de canal, et cætera, pour optimiser votre réseau et éviter des conflits bizarres et mangeurs de bande passante.

N.B. : les étapes 1, 2 et 3 correspondent au niveaux correspondant du modèle OSI, alors que l'étape 4 est un complément destiné à résoudre les situations générées par les masques de réseau 255.255.255.255. En fait un masque de sous-réseau dont les 32 bits sont à 1 viole le standard OSI car cela force le réseau à utiliser la même adresse pour les messages de diffusion et les messages destinés à l'IP de la machine. De plus, l'adresse du réseau n'existe pas.

On pourrait critiquer ce point de vue, mais l'utilisation du modèle OSI pour configurer un réseau sans-fil entraînerait une perte importante lors de la configuration des sous-réseaux ; en effet, chaque sous-réseau gaspille 2 adresses IP (adresse réseau et adresse de diffusion) et cette structure fait perdre en souplesse pour l'affectation des adresses IP (de manière géographique). Vous en trouverez plus à ce sujet en annexe (cf. Annexe A, *Le masque 255.255.255.255, le mandataire ARP et la fonction de pont*).

Vous noterez que l'étape 2 n'est pas nécessaire dans un réseau filaire, car il n'y a pas de paramètres particuliers à configurer à ce niveau.

5.2. Configuration bas-niveau du noyau

Il s'agit d'un problème récurrent dans l'administration d'un PC : faire que votre noyau (ou vos logiciels d'une manière générale) reconnaisse votre matériel.

Les cartes sans-fil sont encore plus complexes car beaucoup d'entre-elles se branchent sur un port PCMCIA. Donc, vous devrez d'abord permettre à votre noyau de voir l'adaptateur PCMCIA, ensuite vous pourrez essayer d'installer le pilote matériel spécifique à votre carte sans-fil.

Dans les configurations PCMCIA, vous devrez :

1. Installer les sources du noyau Linux, de http://www.kernel.org dans `/usr/src/linux` (cf. les utilitaires **tar** et **gzip**)

2. Installer les sources des pilotes pcmcia, se trouvant sur http://pcmcia-cs.sourceforge.net/ dans `/usr/src/pcmcia` (voir les utilitaires tar et gzip)

3. Configurer et recompiler votre noyau : lisez le fichier `README` inclus dans `/usr/src/linux`

4. Configurez et recompilez les sources du pilote pcmcia : dans `/usr/src/pcmcia` utilisez **configure** et **make**. Assurez vous que votre pilote est présent, sinon vous devrez l'installer en suivant les instruction du pilote (normalement, un **`tar xvzf`** ***`driver.tgz`*** dans le répertoire `pcmcia` devrait suffire). Ensuite, entrez **`make all`** pour compiler. Enfin, entrez **`make install`**.

5. Après l'installation, vous trouverez des fichiers de configuration utiles dans `/etc/pcmcia`.

Dans les autres cas :

1. Si votre pilote est présent dans les sources du noyau Linux (ce qui est souvent le cas des cartes récentes), vous devrez l'installer dans un répertoire puis le compiler.

Une fois que vous connaîtrez le nom du module, vous devrez le charger : dans une configuration PCMCIA vous aurez seulement besoin de lancer le démon pcmcia (**/etc/rc.d/init.d/pcmcia start** sous RedHat). Pour les autres configurations, il suffira d'utiliser **modprobe *nom_du_module options*** . Dans les *options*, vous donnerez les ports d'E/S, ainsi que les paramètres d'IRQ et de la liaison données (voir la Section 5.3, « Configuration de la liaison données ») au pilote sans-fil.

Dans tous les cas, voici les outils pouvant être utilisés pour vérifier si votre matériel est correctement reconnu par le pilote :

1. **tail /var/log/messages** qui affiche les dernières entrées du journal système (`syslog`) ;
2. **dmesg** qui affiche les messages du noyau ;
3. le répertoire `/proc` : les fichiers `ioports` (informations sur les ports d'E/S), `devices` (périphériques reconnus), `interrupts` (IRQ) et les sous-répertoires de `driver`, contenant des informations spécifiques à certains pilotes.

5.3. Configuration de la liaison données

Qu'est-ce que c'est ?

Les réseaux filaires ont seulement besoin d'être connectés et l'on peut immédiatement définir les paramètres TCP/IP.

En revanche, pour les réseaux sans-fil il est nécessaire de paramétrer la liaison données, afin de définir par exemple :

1. De quelle type de réseau sans-fil est-ce que je dépends (Ad-hoc ou Infrastructure) ?
2. Quel canal dois-je utiliser ?
3. À quel sous-réseau (BSSID) est-ce que j'appartiens, quel est mon identifiant (ESSID) ?
4. Ma communication est-elle protégée par un algorithme de chiffrement ? Quelle est la longueur de la clé ?

Comme vous le voyez, il y a beaucoup de paramètres à définir. Ceci tient à l'architecture des réseaux sans-fil : une personne se trouvant dans un périmètre proche n'aurait qu'à diriger une antenne dans la bonne direction et à entrer les bons paramètres TCP/IP pour intercepter vos paquets et utiliser vos services.

De plus, la présence d'un grand nombre de sous-réseaux sans-fil pourrait générer des interférences entre ceux-ci.

Il existe donc :

1. des options à définir lors du chargement du module : **modprobe ray_cs essid='*LINUX*'** par exemple, ou
2. des utilitaires permettant de paramétrer le pilote en cours de fonctionnement : **proxcfg eth1 master** par exemple.

5.4. Configuration IP

Il s'agit du troisième problème auquel vous allez faire face. À cette étape, la situation ne devient problématique que lorsque votre réseau évolue pour devenir plus grand.

Souvenez-vous : le réseau IP sans-fil ne vous stressera pas si vous ne le stressez pas !

Une configuration simple

```
                    Chaque hôte voit tous les autres

                             A - - - - - C
                               \       /
                             |   \   /   |
                                   /\
                             |   /    \  |
                               /        \
                             B - - - - - D
```

Une configuration comme celle-ci ne requiert rien de particulier (à un niveau IP) : vous devez seulement attribuer une adresse IP à chaque hôte et un masque de sous-réseau global cohérent.

Une configuration plus complexe

```
                     A ne voit pas B directement

                              A <- - - -
                          Pas \        |
                           de \        C
                communication \        |
                              B <- - - -
```

Dans ce cas, A et B ne peuvent communiquer qu'en passant par C

Si le réseau est en mode Infrastructure et que C est le point d'accès, il n'y a pas de problème. En mode Ad-hoc, vous pouvez aussi donner à un hôte le rôle de *maître* (Je sais que ce terme n'est pas si cérémonieux !). Celui-ci pourra créer un BSS auquel n'importe quel autre hôte pourra se joindre.

Au niveau IP, nous sommes ici totalement connectés : A et B communiquent avec C en utilisant la même interface de C. En conséquence, si vous essayez de faire un **ping** de A vers B, vous recevrez de nombreux paquets `ICMP REDIRECT` (redirection ICMP) envoyés par C, celui-ci informant A que sa destination se trouve déjà dans le réseau d'où provient sa requête.

Solution : sur la machine C, entrez **`echo 0 > /proc/sys/net/ipv4/conf/`*`ethx`*`/send_redirects`** (où *`ethx`* est l'interface de C vers A et B) pour annuler l'envoi de ces messages.

Un autre problème : quel masque de sous-réseau attribuer à A et B ? Si vous attribuez à A un masque qui inclut A et B, rien ne fonctionnera, car A n'utilisera pas la passerelle (C) mais fera directement des requêtes ARP afin d'obtenir l'adresse MAC de sa destination, qui hélas est injoignable directement.

Vous pourriez envisager d'utiliser un mandataire ARP (*proxy arp*), mais cela ne marchera pas non plus. En effet, un mandataire ARP ne répondra à la source que si la destination et la source se trouvent sur deux interfaces différentes, ce qui n'est pas le cas !

Vous devrez donc attribuer un masque de sous-réseau très petit (Win9x autorise `255.255.255.254`, alors que sous WinNT il faudra au moins `255.255.255.248`), et vous devez vous assurer que les hôtes A et B n'ont pas une adresse réseau identique.

Exemples :

1. `IP(A) = x.y.z.2/31` et `IP(B) = x.y.z.3/31`. Cela ne fonctionnera pas car A pen-

sera que B est dans son sous-réseau, et cherchera à réaliser une requête ARP pour le joindre. B ne recevra pas la requête (il n'est pas joignable par A). C ne répondra pas, même s'il dispose d'un mandataire ARP car, pour lui, A et B se trouvent sur la même interface.

2. `IP(A) = x.y.z.1/31` et `IP(B) = x.y.z.2/31`. Cela fonctionnera, car A contactera C pour entrer en relation avec B (envoi des paquets destinés à B vers l'adresse MAC de C).

En général, avec un masque de sous-réseau de `255.255.255.254`, le système fonctionne avec 2 adresses IP en changeant simplement le dernier chiffre, et en s'assurant que si le dernier chiffre de l'un des adresses est paire, le dernier chiffre de l'autre adresse n'est pas le chiffre impair immédiatement supérieur.

Ce sont des contraintes de TCP/IP, mais c'est la seule méthode permettant d'obtenir un haut niveau de flexibilité.

Note

Si vous utilisez un point d'accès, c'est-à-dire un réseau en mode Infrastructure, vous n'aurez pas le problème de redirection, car tout sera résolu au niveau liaison de données (presque tous les points d'accès se comportent comme des ponts). Mais les points d'accès sont chers (un millier de dollars, voire plus)5 et il est plus économique d'utiliser un Pentium 133 avec 32 Mo de mémoire vive comme routeur, même avec 2 cartes ou plus.

Accès à Internet

```
A - - - - - C - - Internet
  \       /
|   \   /   |
      /\
|   /    \  |
  /       \
B - - - - - D
```

Il existe plusieurs situations :

1. Seule la machine C dispose d'une adresse IP publique (sur son interface côté Internet). Vous devrez simplement lui attribuer une adresse IP privée (`192.168.x.y` par exemple) pour le réseau sans-fil, ce qui permettra l'utilisation du routage et du masquage. C sera la passerelle par défaut de A, B et D.

2. Vous disposez d'un sous-réseau public visible depuis Internet et C est votre passerelle par défaut (entre votre réseau et Internet). Vous devrez seulement activer le routage sur C, et paramétrer définir C comme passerelle par défaut sur A, B et D.

3. Vous disposez d'un sous-réseau public visible depuis Internet et C n'est pas votre passerelle par défaut vers Internet. Deux solutions s'offrent à vous : modifier votre passerelle par défaut pour qu'elle pointe sur C pour les adresses appartenant à votre réseau. Par symétrie, C doit pointer sur votre passerelle par défaut pour aller vers Internet. L'autre solution serait d'activer un mandataire ARP (*proxy arp*) sur C (**`echo 1 > /proc/sys/net/ipv4/conf/`*`ethx`*`/proxy_arp`** où *`ethx`* est l'interface vers la passerelle par défaut), puis paramétrer C pour que sa passerelle par défaut pointe vers la passerelle par défaut. Utiliser un mandataire ARP est un détournement de TCP/IP mais fonctionne bien.

Réseaux mixtes : filaires et sans-fil

```
Internet
```

5NdT : les prix ont maintenant beaucoup baissé en ce qui concerne le 802.11b. L'achat d'un point d'accès peut donc se révéler une solution plus économique que l'utilisation d'une machine Linux.

```
                                       \
                                        \        E
                                         \     /
                                          \  /       sans-fil
                               A - - - - - C - - F
                                 \       /   \
                               |   \   /   |   \
                      filaire        /\          G
                               |   /    \  |
                                 /       \
                               B - - - - - D
```

Maintenant, C relie deux réseaux : sans-fil à droite et filaire à gauche.

De plus, il y a un accès à Internet, vous avez donc au total 3 cartes réseau dans C.

Quels adresses IP attribuer aux hôtes ? Vous avez deux solutions :

1. Séparer le réseau en deux sous-réseaux : par exemple `192.168.1.0/24` et `192.168.2.0/24`. Cette solution est rapide mais ne sera pas extensible si vous utilisez des adresses Internet publiques, car cela obligerait à gaspiller trop d'adresses IP.

2. Activer un mandataire ARP (*proxy arp*) sur C pour les 2 interfaces. Les paramètres réseau (adresse réseau et masque) sont les mêmes pour le sans-fil et le filaire, mais grâce au mandataire ARP, on peut choisir quelles adresses IP sont sur le réseau filaire et quelles adresses sont sur le réseau sans-fil.

Examinons cette deuxième solution plus en détails

Par exemple, considérez que vous avez un sous-réseau public en `x.y.z.0/24`

Les interfaces sont :

```
ifconfig eth0 x.y.z.C netmask 255.255.255.255 # filaire
ifconfig eth1 x.y.z.C netmask 255.255.255.255 # sans-fil
ifconfig eth2 x.y.z.C netmask 255.255.255.255 # vers Internet
```

Les routes statiques pour `eth2` sont :

```
route add adresse_passerelle dev eth2
route add default gw adresse_passerelle
```

Ces routes servent à diriger vers votre passerelle par défaut toutes les requêtes vers Internet : comme vous pouvez le remarquer, vous devez d'abord indiquer à Linux où se trouve le routeur, puis de diriger par défaut les requêtes vers celui-ci.

Routes statiques sur `eth0` :

```
route add x.y.z.A dev eth0
route add x.y.z.B dev eth0
route add x.y.z.D dev eth0
```

(Les hôtes A, B et D se trouvent sur le réseau filaire.)

Routes statiques sur `eth1` :

```
route add x.y.z.E dev eth1
route add x.y.z.F dev eth1
route add x.y.z.G dev eth1
```

(Les hôtes E, F et G se trouvent sur le réseau sans-fil.)

Cela vous donnera une très grande souplesse, mais vous devrez définir à la main la configuration relative à chaque hôte.

6. Configuration

Avertissement

Cette partie du texte original a été partiellement réécrite pour la version française. En effet, une grande partie des informations fournies par la version originale étaient trop obsolètes pour être utiles. Ces modifications ont été transmises à l'auteur pour inclusion dans les prochaines versions.

Je présente ici quelques exemples (que j'espère utiles) de la façon de configurer certaines cartes sans-fil économiques courantes.

6.1. Informations générales sur la configuration

Les cartes sans-fil ont une interface similaire à n'importe quelle autre carte Ethernet, vous devrez donc ajouter dans `/etc/conf.modules` :

```
alias ethx module
```

- *`ethx`* est l'interface que vous voulez attribuer à votre carte sans-fil ;
- *`module`* est le nom du module du noyau.

```
options module io=0xAAA irq=I …
```

- *`0xAAA`* est l'adresse de base d'E/S correspondant à la carte ;
- *`I`* est l'IRQ utilisée par votre carte ;
- et ainsi de suite s'il y a d'autre paramètres.

Après cela, vous pourrez utiliser les commandes **ifconfig** et **route** pour configurer votre cartes au niveau IP.

6.2. Symphony (Proxim)

Type de réseau : FHSS, Ad-hoc seulement, et avec le protocole propriétaire OpenAir.

Site web : http://www.komacke.com/distribution.html où vous devrez télécharger la documentation et les pilotes Linux. Les pilotes pour les systèmes Windows sont disponibles sur http://www.proxim.com.

Les pilotes sont fournis avec un code source à compiler.

Configuration

1. Décompressez le fichier tar dans un répertoire vide.
2. Tapez **make** pour obtenir de l'aide.
3. **`make modules; make module_install`** pour installer le pilote `rlmod.o` et l'utilitaire **proxcfg**.
4. Pour lancer le pilote (après avoir modifié `/etc/conf.modules` : voir la Section 6.1, « Informations générales sur la configuration »), vous devrez simplement relancer l'interface avec la commande **ifconfig**.

L'utilitaire **proxcfg** [http://www.komacke.com/proxcfg.html] vous permet de modifier les paramètres de la liaison données :

1. **`proxcfg`** ***`ethx`*** **`sta`** pour la configurer en station (esclave)
2. **`proxcfg`** ***`ethx`*** **`msta`** pour la configurer en station maître (maître)
3. **`proxcfg`** ***`ethx`*** **`alt`** pour la configurer en mode automatique

C'est tout ce que vous devez savoir pour la faire fonctionner correctement.

Pour plus d'information, consultez http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Linux.Wireless.drivers.html#RangeLan2.

6.3. Aviator 2.4 et AviatorPro (Webgear)

Type de réseau : FHSS, Ad-hoc seulement pour la Aviator 2.4 et Infrastructure pour la AviatorPro.

Ce sont des cartes PCMCIA, donc vous devrez compiler les sources pcmcia pour pouvoir les utiliser. Le pilote permettant d'utiliser ces cartes est livré avec les sources PCMCIA (version 3.1.33 minimum) et est inclus dans les noyaux Linux à partir de la version 2.4.19.

Configuration

1. Vous devez télécharger les sources pcmcia, les décompresser dans `/usr/src/pcmcia`, les compiler et les installer (voir la Section 5.2, « Configuration bas-niveau du noyau ») ;
2. Le fichier de configuration `/etc/pcmcia/config.opts` devrait[6] contenir une ligne comme celle-ci :

   ```
   module "ray_cs" opts …
   ```

   À cet endroit, vous devez modifier certains paramètres de la liaison données. Vous devez indiquer ces paramètres sur la même ligne, juste après `opts`. Les valeurs possibles de ces paramètres sont :

   - **`pc_debug=`*`x`***, où *x* est le niveau de journalisation ;
   - **`net_type=`*`x`***, avec *x*=0 pour le mode Ad-hoc et *x*=1 pour le mode Infrastructure ;

[6] Si ce n'est pas le cas, le fichier original, nommé `etc/config.opts` se trouve dans les sources pcmcia.

- **`essid=x`**, *x* est le ESSID.

Finalement, pour vérifier votre configuration avec , donnez à `pc_debug` une valeur supérieure à zéro. Vous verrez dans votre console des messages relatifs à la liaison données (comme `network started` lors de la création d'un nouveau réseau sans-fil et `network joined` lors de la jonction avec un réseau sans-fil existant).

Le fichier `/proc/ray_cs` peut également vous aider : si le BSSID qu'il indique est nul, vous ne pourrez recevoir de données d'aucune carte. En effet, le BSSID indique à quel réseau vous appartenez.

Pour plus d'informations, consultez http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Linux.Wireless.drivers.html#Raylink

6.4. Produits Cabletron et Wavelan I, II, Orinoco de Lucent

Type de réseau : DSSS, Ad-hoc et Infrastructure.

Les produits Lucent sont très professionnels. NdT : La gamme des produits sans-fil de Lucent a été revendue à Proxim [http://www.proxim.com]

Site web : http://www.lucent.com et http://www.orinoco.net.

Le pilote (`orinoco_cs`) permettant d'utiliser ces cartes est livré avec les sources PCMCIA (version 3.1.34 minimum) et est inclus dans le noyaux Linux à partir de la version 2.4.21.

Alors que les pilotes précédents étaient configurés via des paramètres indiqués au module lors de son chargement, la configuration de ce pilote sera réalisée via les *extensions sans-fil* du noyau (*wireless extensions*). Les extensions sans-fil sont un jeu d'appels systèmes permettant de communiquer avec les pilotes des cartes réseaux sans-fil. Les utilitaires permettant d'exploiter ces extensions sont regroupés dans le paquet *wireless-tools*. Parmi ceux-ci se trouve la commande **iwconfig** qui sera spécifiquement utilisée pour la configuration d'une carte (voir http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Tools.html).

Configuration

1. Vous devez télécharger les sources pcmcia, les décompresser dans `/usr/src/pcmcia`, les compiler et les installer (voir la Section 5.2, « Configuration bas-niveau du noyau ») ;

2. Placez dans le fichier de configuration `/etc/pcmcia/wireless.opts` une configuration vierge comme celle-ci :

```
*,*,*,*) # <- Motif de sélection
         # Définit à quelles cartes cette configuration s'applique
         # (Le 4e composant de ce motif correspond à l'adresse MAC)

    INFO="Carte WaveLan ou Orinoco"
    # ESSID (nom de réseau étendu) : "Mon reseau", "any"
    ESSID=""
    # Mode de fonctionnement :
    # ------------------------
    # Ad-Hoc, Managed, Master, Repeater, Secondary, auto
    MODE=""
    # Canal : 1, 2, 3
    # ---------------
    # Cette option n' est applicable qu'au mode Ad-hoc
```

```
CHANNEL=""
# Débit en bits : auto, 1M, 11M
# -----------------------------
# Ce paramètre sert a définir la vitesse de transmission.
# attention à ce paramètre pour la compatibilité avec les cartes
# Cabletron.
RATE=""
;;
```

Il ne vous reste plus qu'à remplir ce fichier pour configurer la liaison données.

Note

Pour utiliser 2 cartes Wavelan avec des configurations différentes sur la même machine (par exemple, l'une en mode Ad-hoc et l'autre en mode infrastructure), il suffit d'indiquer dans le fichier de configuration `/etc/pcmcia/wireless.opts` une configuration pour chaque carte. En effet, la quatrième partie du motif de sélection de chaque bloc `*,*,*,`*`XX:XX:XX:XX:XX`*) sert à spécifier à quelle carte (i.e. à quelle adresse MAC) cette configuration s'applique. Il suffit donc de remplacer dans chaque cas cette quatrième composante (*`XX:XX:XX:XX:XX`*) par l'adresse MAC de la carte à laquelle cette configuration s'applique.

Les pilotes pour Lucent peuvent aussi être utilisés pour les cartes Cabletron http://www.cabletron.com

Pour plus d'information, consultez

.

6.5. YDI

Type de réseau : DSSS, Ad-hoc ou Infrastructure.

YDI vend des cartes très professionnelles avec des antennes, des amplis, et cætera.

Site web : http://www.ydi.com. Les pilotes Linux sont disponibles sur : ftp://ftp.linux-wlan.org/pub/linux-wlan/.

Pour installer :

1. Décompressez le fichier tar dans un répertoire vide ;
2. Tapez **make** pour compiler ;
3. **`make install`** pour installer le pilote `am930_isa` et l'utilitaire **wlanctl**.

Une fois ceci fait, vous pourrez choisir de réaliser une configuration manuelle de la liaison données en utilisant **wlanctl** ou une configuration automatique en utilisant les scripts `scripts/wlan` ou `scripts/rc.wlan`.

Les paramètres principaux pour un réglage manuel :

1. **`wlanctl scan …`** pour rechercher des BSS déjà présents ;
2. **`wlanctl netlist`** vous montrera ce qu'a trouvé la commande **`wlanctl scan …`** ;
3. **`wlanctl bsscreate …`** ***`ssid`*** pour créer un nouveau réseau identifié par le *`SSID`* indiqué ;
4. **`wlanctl bssjoin`** ***`bssid`*** pour rejoindre le réseau identifié par le *`BSSID`* indiqué ;
5. **`wlanctl authen`** et **`wlanctl assoc`** pour les services d'authentification

Pour plus d'information, consultez http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Linux.Wireless.drivers.html#Prism.

7. Liens utiles

7.1. Liens sur des logiciels libres

Le projet Linux-wlan [http://www.linux-wlan.org] ;

Le *Wireless Howto* de Jean Tourrilhes [http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux].

7.2. Liens commerciaux

Fatamorgana Computers [http://www.fatamorgana.com/] ;

Site web de Lucent [http://www.lucent.com/] ;

Site web de YDI [http://www.ydi.com/] ;

Site web de Siemens [http://www.siemens.com/].

7.3. Liens généraux

Le site consacré au WiFi [http://rubb.free.fr/802-11/index.php] de l'université de Corse ;

Les articles consacrés aux réseaux sans fil [http://www.commentcamarche.net/wireless/wlintro.php3] du site de vulgarisation informatique Comment ça marche ? [http://www.commentcamarche.net].

8. FAQ — Foire aux questions

Q :
: Quelle est la différence entre un BSSID et un ESSID, et quand aurais-je besoin d'un ESSID ?

R :
: Un BSSID est un nombre de 48 bits utilisé pour identifier une petite zone BSS, où chaque hôte est capable de communiquer avec tous les autres (éventuellement avec un point d'accès).

  Un ESSID est une chaîne de longueur variable qui permet de faire communiquer plusieurs BSS afin de les étendre pour former un Ensemble de services étendus (ESS — *Extended Service Set*). Il faudra un point d'accès pour chaque BSS et ceux-ci ne communiqueront tous ensemble que s'ils ont le même ESSID. Un ESSID ne sera nécessaire que si vous avez un grand réseau ayant au moins deux points d'accès.

Q :
: Quel point d'accès dois-je acheter ?

R :
: Le moins cher que vous trouverez : l'important, c'est que le point d'accès et les cartes sans-fil utilisés aient les mêmes spécifications au niveau matériel : tous compatibles FHSS ou tous compatibles DSSS. Attention aux cartes RangeLan2 de Proxim, car elles utilisent le protocole propriétaire OpenAir, et ne peuvent donc pas communiquer avec des cartes FHSS aux normes.

Q :
: Dans quels cas utiliser le paramètre canal ?

R :
: Quand vous aurez plusieurs réseaux avec des BSS différents (et des fabricants différents), vous risquez d'avoir des problèmes d'interférences : changer le canal utilisé par les points d'accès (en mode infrastructure) ou changer le canal des hôtes (en mode Ad-hoc) pourra vous éviter ce genre de problèmes.

Q :
: Pourquoi n'est-il pas possible de changer le canal sur un hôte en mode Infrastructure ?

R :
: Car le canal est choisi par le point d'accès.

Le masque 255.255.255.255, le mandataire ARP et la fonction de pont

Ici sont traités certains avantages de Linux dans l'interconnexion de réseaux sans-fil.

Linux vous permet de choisir un masque de sous-réseau ayant la valeur `255.255.255.255`, ce qui pourra vous aider à attribuer les adresses IP aux interfaces de votre choix, par exemple une pour `eth0`, une autre pour `eth1`, et ainsi de suite…

Ceci n'a pas d'effet secondaire particulier.

Vous pouvez de plus utiliser le paramètre `proxy_arp` sous `/proc/sys/net/ipv4/conf/ethx/proxy_arp` où *ethx* est votre interface.

Si vous faites **echo 1 > proxy_arp**, vous activerez le mandataire ARP (*proxy arp*) pour cette interface. Vous le désactiverez avec **echo 1 > proxy_arp**.

Qu'est-ce qu'un mandataire ARP ? En résumé, un mandataire ARP vous sera utile lorsque vous voudrez qu'un routeur Linux réponde à une requête ARP alors que l'adresse de destination se trouve sur une autre de ses interfaces.

Exemple :

```
192.168.1.1 ---- 192.168.1.2 routeur Linux 192.168.2.2 ----192.168.2.1
```

Pour faire fonctionner cet exemple, vous devez :

Sans mandataire ARP

1. Sur l'hôte `192.168.1.1` : choisir `192.168.1.2` comme passerelle
2. Sur l'hôte `192.168.2.1` : choisir `192.168.2.2` comme passerelle
3. Effectuer une commande **ping** avec succès d'un côté vers l'autre.

Avec un mandataire ARP

1. Sur l'hôte `192.168.1.1` : choisir `192.168.1.2` comme passerelle
2. Ne pas choisir de passerelle sur l'hôte `192.168.2.1`, mais activer le mandataire ARP sur la bonne interface du routeur
3. Effectuer une commande **ping** avec succès d'un côté vers l'autre.

Dans le second cas, le mandataire ARP permet au routeur Linux de répondre lorsque vous effectuez une commande **ping** depuis l'hôte `192.168.2.1`, en disant qu'il connaît l'hôte `192.168.1.1` et qu'il peut donc répondre à sa place. Ensuite, quand la source commence à envoyer des paquets ICMP, le routeur Linux sait qu'il doit les rediriger vers le véritable hôte `192.168.1.1`.

Dans un réseau sans-fil, un mandataire ARP peut vous aider si vous avez beaucoup de machines Linux utilisées comme routeurs IP et que vous ne voulez pas avoir à paramétrer un grand nombre de routes statiques sur chaque hôte.

Vous pouvez aussi essayer d'utiliser une machine Linux comme pont sur un réseau sans-fil :

1. Installez un noyau stable et récent
2. Téléchargez de bons utilitaires de pontage sur http://bridge.sourceforge.net/.

Un pont devrait être plus simple à gérer.

Le Modem Radio Siemens DECT

Site web : http://www.siemens.com/

Qu'est-ce ? Ces deux composants ne sont pas réellement des cartes PC mais plutôt des modems que vous utilisez comme des périphériques externes.

```
Hôte1-PortSérie - ModemRadio1 - - - - - ModemRadio2 - PortSérie-Hôte2
```

Comment les connecter ?

Si vous vouliez les imaginer d'une manière abstraite, vous pourriez les modéliser comme ceci :

```
Hôte1-PortSérie - - CÂBLE NULL MODEM - - PortSérie-Hôte2
```

Vous aurez donc une connexion entre deux ports série distant avec deux configurations possibles :

1. Linux avec Windows, Linux utilise un démon qui répond aux appels ppp, alors que Windows utilise un connexion d'accès au réseau à distance.
2. Linux avec Linux, vous pouvez lancer sur les 2 hôtes une connexion ppp avec des adresses IP inversées.

Pour le cas n°1, vous pouvez utilisez ce scripts Linux simple :

```
/usr/sbin/pppd -detach lock idle 300 crtscts \
               connect "/usr/sbin/chat -v TIMEOUT 5 AT OK AT OK AT OK AT OK" \
               IPLINUX:IPWINDOWS /dev/ttySx 115200 \
               disconnect "/usr/sbin/chat -v AT OK" \
               ms-dns IPDNS
```

Où :

- `/dev/ttySx` est votre port série ;
- *IPDNS* est l'adresse IP de votre serveur DNS ;
- *IPLINUX* est l'adresse IP de l'hôte Linux et *IPWINDOWS* est l'adresse IP de l'hôte Windows.

Ce script est nécessaire pour faire croire à Windows qu'il y a un modem sur le port série !

Dans Windows, vous devrez créer une connexion réseau à distance avec contrôle de flux matériel (RTS/CTS) activé, une vitesse de `115 200` et un numéro d'appel bidon (demandé par l'accès à distance, mais effectivement non utilisé).

Avec deux machines Linux, vous devrez simplement lancer ce script très simple sur chaque hôte Linux :

```
/usr/sbin/pppd passive local crtscts IPLINUX1:IPLINUX2 /dev/ttySx 115200 noauth pe
```

Où vous inverserez *IPLINUX1* et *IPLINUX2* sur la seconde machine.

Notez que vous pourrez mettre en place une authentification si vous le souhaitez, autant en Linux-Windows qu'en Linux-Linux.