Linux PPP Howto

par Robert Hart, hart@interweft.com.au Version 3.0, 31 mars 1997

(Adaptation française par Antoine Levavasseur `levavass@efrei.fr`, le 22 octobre 1997). Ce document montre comment connecter votre PC Linux à un serveur PPP, comment utiliser PPP pour relier deux réseaux locaux ensemble, et fournit une méthode pour configurer votre machine Linux comme serveur PPP. Ce document fournit également une aide pour déboguer les connections PPP qui ne fonctionnent pas.

Contents

Copyright

Distribution

Le document original sera posté dans comp.os.linux.answers lorsque de nouvelles versions du document arriveront. Il est également disponible en HTML à :

- *Linux Howto Index* `http://sunsite.unc.edu/mdw/linux.html#howto`
- *PPP-HOWTO* `http://www.interweft.com.au/other/ppp-howto/ppp-howto.html`

Les autres formats (SGML, ASCII, PostScript, DVI) sont disponibles à *Howtos - other formats* `ftp://sunsite.unc.edu/pub/Linux/docs/HOWTO/other-formats` .

Comme sunsite.unc.edu est très chargé, vous êtes encouragés à utiliser un miroir plus proche de chez vous. En particulier la version française de ce HOWTO est posté régulièrement dans fr.comp.os.linux.annonce, et est disponible sur ftp.lip6.fr et http://www.freenix.fr/.

Remerciements

Un nombre important et croissant de personnes m'ont aidé à préparer ce document. Remerciements spéciaux à Al Longyear pour l'aide sur PPP lui-même (si il y a des erreurs ici, ce sont les miennes et pas les

siennes), Greg Hankins (qui maintient les Howto Linux) et Debi Tackett (de Maximum Access.com) pour ses nombreuses suggestions sur le style, l'organisation du contenu, la logique et la clarté des explications.

Pour finir, je remercie les nombreuses personnes qui m'ont envoyé leurs commentaires par courrier électronique. Comme tous les auteurs de HOWTO, la satisfaction d'aider est le seul salaire que nous recevons, et cela nous suffit. En écrivant ce HOWTO, je rembourse d'une certaine façon la dette que je - et tous les autres utilisateurs Linux - dois aux personnes qui écrivent et maintiennent le système d'exploitation que nous avons choisi.

Traduction Française de ce document

La traduction du présent document PPP-Howto (et précédemment PPP-FAQ) était assurée par René Cougnenc. C'était l'un des premiers (si ce n'est le premier) utilisateur de Linux en France alors que ce n'était qu'un petit projet un peu fou lancé par Linus Torvalds. René Cougnenc a participé au développement de Linux et a assuré la traduction de nombreux HOWTO et ouvrages sur Linux qui font encore référence. Si vous utilisez Linux aujourd'hui c'est sans doute un peu grâce à lui...
Prendre ses pas dans la traduction de ce HOWTO était un peu délicat car la barre était placée haut, et j'espère que le lecteur que vous êtes ne sera pas déçu par mon travail.
Et si René nous entends là-haut, je suis sûr qu'il lèvera avec nous son verre à la santé de notre système d'exploitation favori ! :-)

Enfin merci à Thomas Parmelan pour la relecture et les conseils sur la forme de la version française de ce document.

1 Introduction

PPP (Point to Point Protocol) est un mécanisme qui permet de faire fonctionner IP (Internet Procotol) et tous les autres protocoles réseaux à travers une liaison série - qui peut être une connexion série directe (avec un câble null-modem), à travers une liaison par un telnet, ou encore une liaison utilisant les modems et les lignes téléphoniques (et bien sûr utilisant les lignes numériques comme RNIS).

Avec PPP, vous pouvez connecter votre PC sous Linux à un serveur PPP et accéder aux ressources réseau aux quelles le serveur est connecté (presque) comme si vous étiez directement connecté à ce réseau.

Vous pouvez également configurer votre PC Linux comme un serveur PPP, de cette façon, d'autres ordinateurs peuvent appeler votre ordinateur et accéder aux ressources de votre PC et/ou réseau local.

Comme PPP est un système qui marche dans les deux sens, vous pouvez également utiliser PPP sur deux PC Linux pour relier ensemble deux réseaux (ou un réseau local à Internet), créant ainsi un réseau à large domaine (WAN : Wide Area Network).

Une différence majeure entre PPP et une connexion Ethernet est bien entendu la vitesse - une connexion Ethernet standard offre à un débit maximal théorique de 10 Mbs (Mega - millions de bits par seconde), alors qu'une liaison analogique par modem offre un maximum de 56 Kbs (kilo - millier de bits par seconde).

Par conséquent, selon le type de votre connexion PPP, il y aura des limitations sur l'utilisation des applications et des services.

1.1 Clients et Serveurs

PPP est un protocole strictement symétrique; il n'y a (techniquement) aucune différence entre la machine qui appelle et la machine qui est appelée. Cependant, pour des raisons de clarté, il est intéressant de penser en terme de **serveur** et **client**.

Quand vous appelez un site pour établir une connexion PPP, vous êtes un **client**. La machine sur laquelle vous-vous connectez est le **serveur**.

Quand vous configurez une machine Linux pour répondre aux appels pour les connexion PPP, vous configurez un **serveur** PPP.

Tous les PC Linux peuvent être à la fois serveur et client PPP - même en même temps si vous avez plus d'un port série (et d'un modem si nécessaire). Comme décrit plus haut, il n'y a pas de réelle différence entre les clients et les serveurs tant que l'on utilise PPP et que la connexion est établie.

Ce document référence la machine qui initie l'appel (qui le compose) comme le **CLIENT**, alors que la machine qui répond au téléphone, vérifie l'authentification de l'appel (en utilisant les noms utilisateurs, les mots de passe et éventuellement d'autres mécanismes) est référencée comme le **SERVEUR**.

L'utilisation de PPP comme client pour relier une ou plusieurs machines à Internet est probablement celle qui intéresse une majorité de gens. Ils utiliseront alors leur PC Linux comme client.

La procédure exposée dans ce document va vous permettre d'établir et d'automatiser votre connexion à Internet.

Ce document va également vous offrir un guide pour permettre de configurer un **serveur PPP** sur votre PC Linux et relier ensemble deux réseaux (avec le routage complet) en utilisant PPP (cela est aussi appelé une liaison WAN - wide area network).

1.2 Différences entre les distributions Linux

Il y a de nombreuses distributions Linux différentes qui on leurs propres idiosyncrasies et façons de faire les choses.

En particulier, il y a deux manières différentes sur une machine Linux (ou Unix) de démarrer, configurer ses interfaces et ainsi de suite.

Il y **l'initialisation BSD** et **l'initialisation SysV system**. Si vous consultez certains forums de discussion Unix, vous trouverez de temps en temps des guerres de religion entre les partisans de chacun des deux systèmes. Si ce genre de chose vous plaît, allez donc les rejoindre pour gâcher de la bande passante avec eux !

Les distributions les plus utilisées sont sans doute :

- Slackware
  qui utilise une initialisation système BSD
- Red Hat (et son précédent associé Caldera)
  qui utilise une initialisation SysV system (bien qu'un peu modifiée)
- Debian
  qui utilise une initialisation SysV system

Les initialisations de type BSD conservent typiquement ces fichiers dans `/etc/` et ces fichiers sont :

```
/etc/rc
/etc/rc.local
/etc/rc.serial
        (et parfois d'autres fichiers)
```

Récemment, certaines initialisation systèmes BSD se mettaient à utiliser un répertoire `/etc/rc.d/` contenant les fichiers de démarrage plutôt que de tout mettre dans `/etc`.

L'initialisation System V conserve ses fichiers d'initialisation dans les répertoires `/etc/` ou `/etc/rc.d` et un certain nombre de sous-répertoires de ces derniers.

```
drwxr-xr-x   2 root     root         1024 Jul  6 15:12 init.d
-rwxr-xr-x   1 root     root         1776 Feb  9 05:01 rc
-rwxr-xr-x   1 root     root          820 Jan  2  1996 rc.local
-rwxr-xr-x   1 root     root         2567 Jul  5 20:30 rc.sysinit
drwxr-xr-x   2 root     root         1024 Jul  6 15:12 rc0.d
drwxr-xr-x   2 root     root         1024 Jul  6 15:12 rc1.d
drwxr-xr-x   2 root     root         1024 Jul  6 15:12 rc2.d
drwxr-xr-x   2 root     root         1024 Jul 18 18:07 rc3.d
drwxr-xr-x   2 root     root         1024 May 27  1995 rc4.d
drwxr-xr-x   2 root     root         1024 Jul  6 15:12 rc5.d
drwxr-xr-x   2 root     root         1024 Jul  6 15:12 rc6.d
```

Si vous essayez de chercher où votre interface et les routes réseau associées sont configurées, vous devez parcourir tous les fichiers pour parvenir à trouver où sont les commandes correspondantes.

1.3 Outils de configuration spécifiques aux distributions

Dans certaines installations (par exemple Red Hat et Caldera), il y a un système de configuration des PPP sous X Window. Ce HOWTO ne traite pas de ces outils spécifiques aux distributions. Si vous avez des problèmes avec ceux-ci, contactez directement les distributeurs !

Pour les utilisateurs de Red Hat 4.x, il y a maintenant un

Red Hat PPP-TIP `http://www.interweft.com.au/` dans la partie Linux ressources et également chez *Red Hat Software* `http://www.redhat.com/` dans la partie support.

2 Adresse IP

Tout matériel connecté à Internet doit avoir sa propre, et unique adresse IP. Elle sont distribuées de façon centralisée par l'autorité désignée pour chaque pays.

Si vous êtes connectés par un réseau local (LAN) à Internet, **VOUS DEVEZ** utiliser une adresse IP de votre propre domaine d'adresses qui vous a été assigné pour les ordinateurs et matériels de votre réseau local. Vous **NE DEVEZ PAS** prendre n'importe quelle adresse IP et l'utiliser lorsque vous vous connectez avec un autre réseau local (et à plus forte raison Internet). Au pire ça ne marchera pas du tout et cela peut causer des ravages puisque votre adresse IP 'volée' va interférer avec les communications d'un autre ordinateur qui utilise déjà l'adresse IP que vous avez prise au hasard.

Remarquez que les adresses IP de ce document appartiennent (à part quelques exceptions) aux adresses de réseaux non connectés qui sont réservées aux réseaux qui ne sont pas (encore) connectés à Internet. (RFC1918)

Il y a des adresses IP qui sont spécifiquement destinées aux réseaux locaux qui ne sont pas connectés à Internet. Ces adresses IP sont :

- Une Adresse Réseau de Classe A
  10.0.0.0 (netmask 255.0.0.0)
- 16 Adresses Réseau de Classe B
  172.16.0.0 - 172.31.0.0 (netmask 255.255.0.0)

- 256 Adresses Réseau de Classe C
  192.168.0.0 - 192.168.255.0 (netmask 255.255.255.0)

Si vous avez un réseau pour lequel vous **n'avez pas** alloué d'adresse IP auprès de l'autorité responsable de votre pays, vous devez utiliser une des adresses réseau de la séquence précédente pour vos machines.

Ces adresses ne doivent **jamais** être utilisées sur Internet.

Cependant, elles peuvent être utilisées pour les réseaux locaux Ethernet sur la machine qui est connectée à Internet. C'est parce que les adresses IP sont en fait allouées aux interfaces réseau et non à l'ordinateur. Ainsi, votre interface Ethernet peut utiliser 10.0.0.1 (par exemple), et quand vous vous connecterez à Internet en utilisant PPP, votre interface PPP récupérera une autre adresse IP (valide) du serveur. Votre PC sera connecté à Internet alors que les autres ne le seront pas.

Cependant, en utilisant Linux, les capacités de IP Masquerading (aussi appelé NAT - Network Adress Translation) de Linux et du programme **ipfwadm**, vous pouvez connecter votre réseau local à Internet (avec quelques restrictions sur les services), même si vous n'avez pas d'adresse IP valide pour vos machines en Ethernet.

Pour plus d'informations sur le sujet, consulter le mini-HOWTO IP Masquerade à *Linux IP Masquerade mini HOWTO* `http://sunsite.unc.edu/mdw/HOWTO/mini/IP-Masquerade`

Pour la plupart des utilisateurs, qui sont connectés avec une seule machine à leur fournisseur d'accès Internet (FAI) par PPP, obtenir une adresse IP (ou plus largement une adresse réseau) n'est pas nécessaire.

Si vous souhaitez connecter un petit réseau local à Internet, de nombreux FAI peuvent vous fournir un sous-réseau dédié (une séquence spécifique d'adresses IP) sur leur espace d'adresses IP existant. Alternativement, utilise IP masquerading.

Pour les utilisateurs qui connectent un seul PC à Internet, la plupart des FAI utilise l'allocation **dynamique** des adresses IP. C'est pendant le processus de connexion que le service PPP que vous contactez va indiquer à votre machine quelle adresse IP utiliser pour l'interface PPP pendant la session en cours. Cette adresse ne sera pas forcément la même à chaque fois que vous vous connecterez à votre FAI.

Avec les adresses IP dynamiques, vous **n'avez pas** la même adresse IP à chaque fois que vous vous connectez. Cela a des répercussions sur les applications de type serveur de votre machine Linux telles que Sendmail, ftpd, httpd et d'autres. Ces services supposent que l'ordinateur offrant le service soit accessible tout le temps à la même adresse IP (ou au moins au même nom de domaine qualifié, et que la résolution des noms par DNS du nom vers l'adresse IP est disponible).

Les limitations de service dues aux allocations dynamiques d'adresses IP (et les façons d'éviter cela, quand c'est possible) sont décrites plus loin.

3 Buts de ce document

3.1 Mettre en place un client PPP

Ce document fournit un guide aux personnes qui souhaitent utiliser Linux et PPP pour appeler un Serveur PPP et configurer une connexion IP avec PPP. Cela suppose que PPP ait été compilé et installé sur votre machine Linux (mais une brève description de la reconfiguration/recompilation de votre noyau avec le support PPP est fournie).

Bien que `dip` (la manière standard de créer des connexions SLIP) puisse être utilisé pour mettre en place un connexion PPP, les scripts `dip` sont généralement un peu complexes. Pour cette raison, ce document ne traite pas l'utilisation de `dip` pour lancer une connexion PPP.

A la place, ce document décrit les programmes standard Linux/PPP (chat/pppd).

3.2 Relier deux Réseaux locaux ou un réseau local à Internet grâce à PPP

Ce document donne des informations (de base) pour relier deux réseaux locaux entre eux ou un réseau local à Internet grâce à PPP.

3.3 Configurer un serveur PPP

Ce document fournit un guide pour configurer votre PC Linux comme serveur PPP (autorisant d'autres personnes à appeler votre PC Linux et établir une connexion PPP).

Notez qu'il y a un grand nombre de façons de configurer Linux comme serveur PPP. Ce document donne une méthode - qui est utilisée par l'auteur pour configurer quelques petits serveurs PPP (chacun avec 16 modems).

Cette méthode est connue pour fonctionner correctement. Cependant, ce n'est pas forcément la meilleure méthode.

3.4 Utiliser PPP avec une connexion Null-Modem directe

Ce document offre un bref aperçu d'utilisation de PPP pour relier deux PC Linux avec un câble null modem. Il est possible de relier d'autres Systèmes d'exploitation à Linux de cette façon également. Pour ce faire, utiliser la documentation du système d'exploitation pour lequel vous êtes intéressé.

3.5 Pour le moment ce document ne couvre pas...

- La compilation du daemon PPP
  Regarder la documentation fournie avec la version de `pppd` que vous utilisez.
- Connecter et configurer un modem sous Linux (en détail)
  Regarder le Serial-HOWTO et pour les initialisations spécifiques au modem, voir

  Modem Setup Information `http://www.in.net/info/modems/index.html` des informations qui pourraient vous aider à configurer votre modem.
- Utiliser `dip` pour faire des connexion PPP
  utiliser `chat` à la place
- Utiliser `socks` ou IP Masquerading
  Il y a d'excellents documents qui couvrent déjà ces paquetages.
- Utiliser EQL pour regrouper deux modems pour une seule connexion PPP.
- Les méthodes de connexion PPP spécifiques aux distributions (comme l'outil de configuration réseau de la Red Hat 4.x). Voir votre distribution pour la documentation sur les méthodes utilisées.
- Le nombre croissant d'outils disponibles pour automatiser les connexions PPP
  Voir la documentation appropriée.

4 Versions des logiciels traités

Ce HOWTO suppose que vous utilisez un noyau Linux 1.2.X avec le programme PPP 2.1.2 ou Linux 1.3.X/2.0.x et PPP 2.2.

Au moment de la rédaction, la dernière version officielle de PPP disponible est ppp-2.2.0f. La nouvelle version (ppp-2.3) est encore en beta.

Il est possible d'utiliser PPP 2.2.0 avec le noyau 1.2.13. Cela nécessite des patchs pour le noyau. Il est recommandé aux utilisateurs du noyau 1.2.13 de passer à ppp-2.2 puisqu'il inclut quelques extensions et corrections de bogues.

Vous devez également être particulièrement vigilant et ne pas utiliser le programme PPP 2.1.2 avec un noyau Linux 2.0.X.

Veuillez remarquer que ce document **ne traite pas** des problèmes venant de l'utilisation de modules chargeables pour votre noyau Linux 2.0.x. Consulter le mini-HOWTO kerneld et la documentation kerneld/module 2.0.x (dans les sources de Linux 2.0.x à `/usr/src/linux/Documentation/..`).

Puisque ce document est destiné à aider les nouveaux utilisateurs, il est fortement recommandé d'utiliser une nouvelle version du noyau Linux et la version appropriée de PPP qui sont connus pour fonctionner ensemble.

5 D'autres documents utiles/importants

Je conseille aux utilisateurs de lire :

- la documentation accompagnant le paquetage PPP;
- les pages de manuel de `pppd` et `chat`;
  (utiliser `man chat` et `man pppd` pour les lire)
- La page de configuration des modems - voir *Modem Setup Information* `http://www.in.net/info/modems/index.html`
- Les excellents livres Unix/Linux publiés par O'Reilly and Associates. (*O'Reilly and Associates On-Line Catalogue* `http://www.ora.com/`). Si vous êtes nouveau à Unix/Linux, **courez** (ne marchez pas) au plus proche marchand de livres informatiques et investissez dans un certain nombre d'entre eux immédiatement !
- La PPP-FAQ maintenue par Al Longyear, disponible sur *Linux PPP-FAQ* `ftp://sunsite.unc.edu/pub/Linux/docs/faqs` .
  Elle contient un grand nombre d'informations utiles sous forme de questions/réponses indispensable lorsque l'on cherche pourquoi PPP ne fonctionne pas (correctement).
- Le nombre croissant de livre Linux venant des différents éditeurs et auteurs;
  Vous êtes encouragés à vérifier la date de parution de ces livres. Le développement et la distribution de Linux tendent à être plutôt rapides, alors que la mise à jour des livres est (généralement) bien plus lente ! Acheter un excellent livre (et il y en a) qui est déjà dépassé entraine une confusion et une frustration considérable pour les nouveaux utilisateurs.

Le meilleur point de départ pour la documentation Linux est *The Linux Documentation Project Home Page* `http://sunsite.unc.edu/mdw/` . Les HOWTO sont mis à jour raisonnablement régulièrement.

Bien que vous puissiez utiliser ce document pour créer votre liaison PPP sans lire aucun de ces documents, vous devriez avoir une bien meilleure compréhension de ce qu'il se passe si vous le faites ! Vous pouvez également exposer vous-même vos problèmes (ou du moins poser les questions les plus intelligentes sur les groupes de discussion comp.os.linux... ou fr.comp.os.linux/unix... ou les mailing list Linux).

Ces documents (ainsi que quelques autres, comme les RFCs correspondants) fournissent des explications complémentaires et plus détaillées que ce qui est possible dans les HOWTO.

Si vous connectez un réseau local à Internet avec PPP, vous avez besoin de connaître un certain nombre de choses sur les réseaux TCP/IP. En complément des documents précédents, vous trouverez dans les livres O'Reilly "TCP/IP Network Administration" et "Building Internet Firewalls" un bénéfice considérable !

5.1 Les Mailing Listes utiles

Il y a de nombreuses mailing listes (listes de diffusion) Linux qui servent de moyen de communication entre les utilisateurs dans de nombreux domaines. Souscrivez absolument à celles qui vous intéressent pour donner votre point de vue.

Remarques de bon sens: certaines listes sont spécifiquement destinés au utilisateurs "experts" et/ou à certains sujets spécifiques. Même si personne ne se plaindra que vous soyez voyeur (souscrire sans poster de messages), vous recevrez certainement des commentaires chauds (voire des flammes) si vous postez des questions de débutant dans les listes innappropriées.

Ce n'est pas parce que les utilisateurs gourous détestent les nouveau utilisateurs, mais parce que ces listes sont là pour contenir des remarques spécifiques à un certain niveau de difficulté.

Vous êtes largement encouragés à souscrire librement à différentes listes mais faites absolument des commentaires qui cadrent avec le sujet (et le niveau) de la liste !

Un bon point de départ pour les mailing-list Linux est

Linux Mailing List Directory `http://summer.snu.ac.kr/~djshin/linux/mail-list/index.shtml`

6 Présentation de ce qui doit être fait pour faire fonctionner PPP comme client

Ce document contient un grand nombre d'informations - qui augmente à chaque version !

Par conséquent, cette section a pour but de fournir une présentation concise des actions pour connecter votre système Linux comme client à un serveur PPP.

6.1 Obtenir/installer les programmes

Si votre distribution Linux ne contient pas les programmes PPP, vous devez les obtenir sur *the Linux PPP daemon* `ftp://sunsite.unc.edu/pub/Linux/system/Network/serial/ppp/ppp-2.2.0f.tar.gz`

C'est la dernière version officielle lors de la rédaction du document. Cependant, prenez la dernière version disponible sur ce site (ppp-2.3 était en béta lors de la rédaction de ce document et devrait sortir bientôt).

Le paquetage PPP contient les instructions sur la façon de compiler et d'installer le logiciel **donc ce HOWTO n'en parlera pas**. !

6.2 Compiler le support PPP dans le noyau

L'installation de PPP Linux se divise en deux parties

- le daemon PPP mentionné ci-dessus
- le support PPP dans le noyau

De nombreuses distributions semblent fournir un support PPP dans les noyaux installés par défaut, mais certaines ne le font pas.

Si lors du boot, le noyau donne un message ressemblant à :

```
PPP Dynamic channel allocation code copyright 1995 Caldera, Inc.
PPP line discipline registered.
```

Votre noyau à le support PPP compilé.

Toutefois, vous devez recompiler votre propre noyau quelque soit votre distribution pour fournir l'utilisation la plus efficace possible des ressources disponibles et de votre configuration matérielle personnelle. Il est bon de savoir que le noyau ne peut être swappé hors de la mémoire. Avoir un noyau le plus petit possible est donc intéressant surtout pour une machine limitée en mémoire.

Ce document fournit les instructions minimales pour la re-compilation du noyau à la section 7 (Configurer votre noyau Linux).

Pour plus de détails, voir le Kernel-HOWTO à *The Linux Kernel HOWTO* `http://sunsite.unc.edu/mdw/HOWTO/Kernel-HOWTO.html`

6.3 Obtenir les informations pour votre fournisseur d'accès à Internet

Il y a presque une infinité de manières de configurer un serveur PPP. Pour se connecter chez votre fournisseur d'accès (ou un serveur PPP pour accéder à votre intranet), vous devez obtenir des informations sur la façon dont le serveur fonctionne.

Puisque vous utilisez Linux, vous pourriez avoir quelques difficultés avec le support de certains FAI (et les serveurs Intranet d'entreprise) qui ne connaissent que les clients MS-Windows.

Toutefois, un nombre croissant de FAI utilisent Linux pour offrir leur service - et Linux pénètre également l'environnement industriel, vous devriez réduire ainsi les chances de rencontrer des problèmes.

La section 8 (Fournir les informations dont vous avez besoin pour votre serveur PPP) vous explique ce dont vous devez savoir sur le serveur PPP sur lequel vous vous connectez et comment obtenir les informations dont vous avez besoin.

6.4 Configurer votre modem et votre port série

Pour se connecter à un serveur PPP et obtenir le meilleur taux de transfert possible, votre modem doit être configuré correctement.

De la même façon, les ports série de votre modem ainsi que votre ordinateur doivent être configurés correctement.

La section 9 (Configurer votre modem et votre port série) fournit des informations à ce sujet.

6.5 Mettre en place la résolution de noms (DNS)

En plus des fichiers qui lancent PPP et automatisent la connexion sur le serveur PPP, il y a un grand nombre de fichiers de configuration qui sont mis en place pour que votre ordinateur puisse résoudre les noms comme `www.interweft.com.au` vers l'adresse IP qui est réellement utilisée pour contacter cet ordinateur. Ces fichiers sont :

- `/etc/resolv.conf`
- `/etc/host.conf`

Voir la section 10 (Configurer la résolution de nom vers l'adresse) pour une configuration détaillée de cela.

En particulier, vous **n'avez pas besoin** de lancer un serveur de nom sur votre PC Linux pour le connecter à Internet (sauf si vous le souhaitez). Tout ce dont vous avez besoin est de connaître l'adresse IP d'au moins l'un des serveurs de noms que vous pouvez utiliser (plutôt celui de votre fournisseur d'accès).

6.6 PPP et les privilèges root

Puisqu'établir une liaison PPP entre votre ordinateur Linux et un autre serveur PPP nécessite la manipulation des périphériques réseau (l'interface PPP est une interface réseau) et des tables de routages du noyau, `pppd` nécessite les privilèges **root**.

Pour des détails là-dessus, voir la section 11 (Utiliser PPP et les privilèges root).

6.7 Vérifier les fichiers de votre distribution PPP et configurer les options de PPP

Il y a un certain nombre de fichiers de configuration et d'appel qui doivent être configurés pour rendre PPP opérationnel. Des exemples sont fournis avec la distribution de PPP et cette section montre quels fichiers il doit y avoir :

```
/etc/ppp/options
/etc/ppp/scripts/ppp-on
/etc/ppp/scripts/ppp-on-dialer
/etc/ppp/options.tpl
```

Vous devrez créer des fichiers supplémentaires en fonction de ce que vous souhaitez exactement utiliser avec PPP :

```
/etc/ppp/options.ttyXX
/etc/ppp/ip-up
/etc/ppp/pap-secrets
/etc/ppp/chap-secrets
```

En plus, le daemon PPP peut traiter un grand nombre d'options en ligne de commande et il est important d'utiliser les bonnes; ainsi cette section montre les options standard de PPP et vous aide à choisir les options que vous utiliserez.

Pour les détails à ce sujet, voir en section 12 (Configurer les fichiers de connexion de PPP).

6.8 Si votre serveur PPP utilise PAP (Password Authentication Protocol)

De nombreux fournisseurs d'accès et serveurs PPP utilisent PAP. Si votre serveur **n'a pas** besoin d'utiliser PAP (si vous pouvez vous connecter à la main et recevoir les messages textes de nom utilisateur/mot de passe il n'utilise pas PAP), vous pouvez tranquillement ignorer cette section.

Plutôt que se connecter dans de tels serveurs en utilisant le nom utilisateur et le mot de passe quand vous êtes interrogés par le serveur pour les entrer, un serveur PPP utilisant PAP n'a pas besoin de login basé sur du texte.

Les informations d'authentification de l'utilisateur sont échangées à la place comme partie prenante du protocole de contrôle de connexion qui est la première partie de l'établissement d'une liaison PPP.

La section 13 (Si votre serveur PPP utilise PAP (Password Authentification Protocol)) fournit des informations sur les fichiers dont vous avez besoin pour établir une connexion PPP en utilisant PAP.

6.9 Se connecter au serveur PPP à la main

Une fois les fichiers de base configurés, c'est une bonne idée que d'essayer ceux-ci en se connectant (avec `minicom` ou `seyon`) et de lancer `pppd` sur votre PC Linux à la main.

Voir la section 14 (Configurer une connexion PPP à la main) pour des détails complets.

6.10 Automatiser votre connexion PPP

Une fois que vous pouvez vous connecter à la main, vous allez maintenant commencer à configurer les scripts qui vont automatiser la mise en place de la connexion.

La section 15 (Automatiser votre connexion - Créer les scripts de connexion) couvre la configuration des fichiers nécessaires, et la mise en place des scripts et du processus de login avec le serveur PPP, ainsi que des détails complets sur `chat`.

Cette section traite des scripts d'authentification du nom utilisateur/mot de passe ainsi que des scripts d'authentification pour les serveurs PAP/CHAP.

6.11 Terminer la connexion

Une fois que votre connexion est activée et fonctionne, vous devez pouvoir désactiver la liaison.

Consultez la section 17 (Terminer une connexion PPP).

6.12 Si vous avez des problèmes

De nombreuses personnes ont des problèmes pour faire fonctionner PPP directement. Les différences entre les serveurs PPP et la façon dont il faut configurer la connexion sont énormes. De plus, PPP a de nombreuses options - et certaines combinaisons d'entre elles ne fonctionnent absolument pas ensemble.

En plus des problèmes de journalisation et de démarrage du service PPP, il y a des problèmes liés aux modems et aux lignes de téléphones qui s'ajoutent !

La section 18 (Résoudre les problèmes) fournit des informations de base au sujet des erreurs classiques, et la façon de les isoler et de les corriger.

Elle **n'est toutefois pas** destinée à offrir plus que les informations de base. Al Longyear maintient la PPP-FAQ qui contient bien plus d'informations à ce sujet !

6.13 Une fois que la connexion est lancée

Une fois que la liaison PPP est opérationnelle (spécifiquement, une fois que les trames IP fonctionnent), `pppd` peut lancer automatiquement (comme utilisateur `root`) un script pour accomplir n'importe quelle fonction qu'un script peut accomplir.

La section 23 (Une fois que la connexion est lancée) fournit des informations sur le script `/etc/ppp/ip-up`, les paramètres que PPP lui envoie et comment l'utiliser pour récupérer votre courrier électronique de votre compte de votre fournisseur d'accès, ou envoyer les messages en attente de transmission sur votre machine.

6.14 Problèmes avec les services IP standards et l'adresse IP dynamique de la liaison PPP.

Comme cela est mentionné dans l'introduction, les adresses IP dynamiques affectent la capacité de votre PC Linux à agir comme un serveur sur Internet.

La section 21 (Problèmes avec les services IP standard et l'adresse IP dynamique de la liaison PPP) fournit des informations sur les (principaux) services affectés et ce que vous pouvez faire (si c'est possible) pour contourner cela.

7 Configurer votre noyau Linux

Pour utiliser PPP, votre noyau Linux doit être compilé avec le support PPP. Récupérez le code source de votre noyau Linux si vous ne l'avez pas encore - il se situe dans `/usr/src/linux` dans un système de fichiers Linux standard.

Vérifiez ce répertoire - de nombreuses distributions installent les sources (fichiers et répertoires) pendant leur processus d'installation.

Au démarrage, votre noyau écrit une grande quantitée d'informations. Parmi elles, est indiqué si le noyau contient le support PPP. Pour afficher ces informations, regardez votre fichier syslog, ou utilisez `dmesg | less` pour afficher ces informations à l'écran. Si votre noyau contient le support PPP, vous verrez quelques lignes du genre

```
PPP Dynamic channel allocation code copyright 1995 Caldera, Inc.
PPP line discipline registered.
```

(pour les noyau Linux version 2.0.x)

Les sources du noyau Linux sont disponibles par ftp à `sunsite.unc.edu` ou ses nombreux miroirs, en particulier `ftp.lip6.fr`.

7.1 Installer les sources du noyau

La suite regroupe de brèves instructions pour obtenir et installer les sources du noyau Linux. Des informations complètes sont disponibles a *The Linux Kernel HOWTO* `http://sunsite.unc.edu/mdw/HOWTO/Kernel-HOWTO.html` .

Pour installer et compiler le noyau Linux vous devez être `root`.

1. Changer de répertoire vers `/usr/src/`
   `cd /usr/src/`

2. Vérifier dans `/usr/src/linux` pour voir si les sources sont déjà installées.

3. Si vous n'avez pas les sources, récupérez les à *Linux kernel source directory* `ftp://sunsite.unc.edu/pub/Linux/kernel/v2.0` ou sur un miroir plus proche comme `ftp.lip6.fr`.
   Si vous cherchez une version plus ancienne du noyau (comme la 1.2.X), elles sont conservées dans
   Old Linux kernel source directory `ftp://sunsite.unc.edu/pub/Linux/kernel/old` .

4. Choisissez le noyau approprié - généralement le plus récent disponible est le meilleur. Le récupérer et mettre le fichier tar dans `/usr/src`.
   Remarque: un fichier 'tar' est une archive - parfois compressée (comme le sont les fichiers tar des sources du noyau Linux) il contient de nombreux fichiers répartis dans plusieurs répertoires. C'est l'équivalent Linux des fichiers multi-répertoires sous MS-DOS.

5. Si vous avez déjà les sources de Linux installées, mais que vous voulez les mettre à jour, il faut effacer les anciens fichiers. Utiliser la commande
   `rm -rf /usr/src/linux`

6. Maintenant décompresser et extraire les sources avec la commande
   `tar xzf linux-2.0.XX.tar.gz`

7. Maintenant, `cd /usr/usr/linux` et lire le fichier README. Il contient une excellente explication de la façon de faire pour configurer et compiler un nouveau noyau. Lisez ce fichier (c'est une bonne idée de l'imprimer pour en garder une copie sous la main quand vous compilerez votre noyau jusqu'à ce que vous l'ayez fait suffisamment pour vous débrouiller tout seul).

7.2 Connaître votre matériel

Vous **DEVEZ** savoir quelles cartes/périphériques vous avez dans votre PC si vous commencer à recompiler votre noyau !!! Pour certains périphériques (comme les cartes sons) vous devez également connaitre quelques paramètres (comme les IRQ et les adresse I/O et des choses du genre).

7.3 Compilation du noyau - le noyau Linux 1.2.13

Pour commencer le processus de configuration, suivre les instructions du fichier README pour installer les sources correctement. Vous lancez le processus de configuration du noyau avec

```
make config
```

Pour utiliser PPP, vous devez configurer le noyau pour inclure le support PPP (PPP a besoin DE `pppd` ET DU support de PPP dans le noyau).

```
PPP (point-to-point) support (CONFIG_PPP) [n] y
```

Répondre aux autres questions du `make config` en accord avec le matériel de votre PC et les caractéristiques que vous souhaitez pour votre système d'exploitation Linux. Continuer ensuite à suivre les instructions de compilation et d'installation du `README` pour votre nouveau noyau.

Le noyau 1.2.13 créait seulement 4 périphériques PPP. Pour les cartes séries multiports, vous devez éditer les sources du noyau PPP pour avoir plus de ports. (Voir le fichier `README.Linux` accompagnant la distribution de PPP-2.1.2 pour des détails complets des éditions simples que vous devez faire).

Remarque : la configuration du noyau 1.2.13 ne permet pas de revenir en arrière - si vous faites une erreur en répondant à une des questions dans le `make config` quitter avec CTRL-C et recommencer.

7.4 Compilation du noyau Linux 1.3.x et 2.0.x

Pour Linux 1.3.x et 2.0.x, vous pouvez utiliser le même processus que pour Linux 1.2.13. De même, suivez les instructions du fichier `README` pour installer les sources correctement. Vous lancez la configuration du noyau avec

```
make config
```

Cependant, vous pouvez aussi utiliser

```
make menuconfig
```

Qui fournit un système de configuration avec une aide en ligne et vous pouvez avancer et reculer dans la configuration.

Il y a aussi l'interface de configuration sous X-Window chaudement recommandée

```
make xconfig
```

Vous pouvez compiler le support PPP directement dans votre noyau ou sous la forme d'un module chargeable dynamiquement.

Si vous utilisez PPP occasionnellement, alors compiler le support PPP en module chargeable est recommandé. Avec `kerneld`, votre noyau chargera automatiquement le(s) module(s) nécessaires pour fournir le support PPP quand vous lancerez une liaison PPP. Cela permet d'économiser une partie de l'espace mémoire : le noyau ne peut être swappé, alors que les modules chargeables sont automatiquement enlevés si ils ne sont pas utilisés.

Pour cela, vous devez activer le support des modules chargeables :

```
        Enable loadable module support (CONFIG_MODULES) [Y/n/?] y
```

Pour ajouter le support PPP à votre noyau, répondre à la question suivante :

```
        PPP (point-to-point) support (CONFIG_PPP) [M/n/y/?]
```

Pour un module chargeable, répondre **M**, sinon pour le compiler directement dans le noyau, répondre **Y**.

A la différence du 1.2.13, le noyau 2.0.x crée les périphériques PPP au vol et selon les besoins, et il n'y a absolument pas besoin de bricoler le source pour augmenter le nombre de périphériques PPP.

7.5 Remarque sur PPP-2.2 et `/proc/net/dev`

Si vous utilisez PPP-2.2, vous trouverez qu'un problème de la création des périphériques PPP 'au vol' est qu'aucun périphérique n'est trouvé dans le fichier `/proc/net` tant qu'un périphérique n'est pas créé avec le lancement de `pppd` :

```
[hartr@archenland hartr]$ cat /proc/net/dev
Inter-|   Receive                    |  Transmit
 face |packets errs drop fifo frame|packets errs drop fifo colls carrier
    lo:  92792    0    0    0    0    92792    0    0    0     0    0
  eth0: 621737   13   13    0   23   501621    0    0    0  1309    0
```

Une fois que vous avez un (ou plus) services ppp lancés, vous verrez des entrées comme celles-ci (sur un serveur PPP) :

```
[root@kepler /root]# cat /proc/net/dev
Inter-|   Receive                  |  Transmit
 face |packets errs drop fifo frame|packets errs drop fifo colls carrier
    lo: 428021    0    0    0    0  428021    0    0    0     0    0
  eth0:4788257  648  648  319  650 1423836    0    0    0  4623    5
  ppp0:   2103    3    3    0    0    2017    0    0    0     0    0
  ppp1:  10008    0    0    0    0    8782    0    0    0     0    0
  ppp2:    305    0    0    0    0     297    0    0    0     0    0
  ppp3:   6720    7    7    0    0    7498    0    0    0     0    0
  ppp4: 118231  725  725    0    0  117791    0    0    0     0    0
  ppp5:  38915    5    5    0    0   28309    0    0    0     0    0
```

7.6 Considérations générales sur la configuration du noyau pour PPP

Si vous configurez votre PC Linux comme un serveur PPP, vous devez activer le support IP Forwarding. Il est également nécessaire si vous voulez utiliser Linux pour relier deux réseaux locaux entre eux ou un réseau local à Internet.

Si vous reliez un réseau local à Internet (ou reliez deux réseaux entre eux), vous devez être concernés par la sécurité. Ajouter le support pour IP firewall dans votre noyau est alors sans doute obligatoire !

Vous aurez également besoin, d'utiliser IP Masquerade pour relier un réseau local qui utilise les adresses IP non connectées mentionnées plus haut.

Pour activer IP Masquerade et IP firewalling, vous **DEVEZ** répondre Oui à la première question de `make config` :

```
Prompt for developement and/or incomplete code/drivers (CONFIG_EXPERIMENTAL)?
```

Bien que cela puisse sembler un peu bizarre aux nouveaux utilisateurs, je confirme que de nombreux utilisateurs utilisent les possibilités de IP Masquerading et de IP firewalling du noyau Linux 2.0.X sans problèmes.

Une fois que vous avez installé et booté votre nouveau noyau, vous pouvez commencer à configurer et à tester votre liaison PPP.

8 Fournir les informations dont vous avez besoin pour votre serveur PPP

Avant d'établir la connexion PPP avec un serveur, vous devez obtenir les informations suivantes (de l'administrateur système/support utilisateur/ personnes gérant le serveur PPP) :

- Le numéro de téléphone à composer pour le service
  Si vous êtes derrière un standard, vous avez aussi besoin du numéro qui vous permet de sortir à l'extérieur du standard - c'est généralement le chiffre zéro (0) ou neuf(9).

- Est-ce que le serveur utilise des adresse IP DYNAMIQUES ou STATIQUES ?
  Si le serveur utilise des adresses IP STATIQUES, vous devez savoir quelle adresse IP utiliser à la fin de la connexion PPP. Si votre FAI vous fournit une adresse de sous-réseau valide, vous devez connaître le masque de sous-réseau à utiliser (netmask).

  La plupart des FAI utilisent des adresses IP DYNAMIQUES. Comme cela est mentionné plus haut, cela a des impliquations sur les services que vous pouvez utiliser.

  Cependant, même si vous utilisez des adresses IP STATIQUES, la plupart des serveurs PPP ne laisseront jamais (pour des raisons de sécurité) les clients spécifier leur adresse IP car c'est un risque pour la sécurité. Vous **devez** connaître aussi cette information !
- Quel sont les adresses IP des serveur de noms des fournisseurs d'accès ?
  Il doit y en avoir au moins deux, même si un seul suffit.

  Il peut y avoir un problème à ce niveau. Le service PPP de MS Windows 95 permet que l'adresse DNS soit transmise au client dans le processus de connexion. Ainsi, votre fournisseur d'accès pourrait vous répondre que vous n'avez pas besoin de l'adresse IP du serveur DNS.

  Avec Linux, vous **devez** connaître l'adresse d'au moins un DNS. L'implémentation PPP de Linux ne permet pas de configurer dynamiquement à la connexion l'adresse du serveur DNS - et ne pourra certainement jamais.

  Remarque: bien que Linux (comme client PPP) ne peut pas recevoir d'adresse DNS du serveur, il peut, en tant que serveur, envoyer l'information au client en utilisant l'option `dns-addr` de pppd.
- Est-ce-que le serveur a besoin d'utiliser PAP/CHAP ?
  Si tel est le cas, vous devez connaître le id et le secret que vous avez pour vous connecter. (c'est probablement votre nom utilisateur et le mot de passe de votre fournisseur d'accès).
- Le serveur est-il un système Microsoft Windows NT et, dans ce cas, utilise-t-il le système PAP/CHAP ?
  De nombreux fournisseurs d'accès semblent utiliser MS-Windows NT de cette façon pour améliorer la sécurité.

Attention à l'information suivante - elle risque d'être utile !

9 Configurer votre modem et votre port série

Vous devez vous assurer que votre modem est correctement configuré et que vous savez sur quel port série il est branché.

Se souvenir que :

- DOS COM1: = Linux /dev/cua0 (et /dev/ttyS0)
- DOS COM2: = Linux /dev/cua1 (et /dev/ttyS1)
  etc.

Il est également bien de se souvenir que si vous avez 4 ports séries, la configuration standard d'un PC est d'avoir COM1 et COM3 partageant IRQ4 et COM2 et COM4 partageant IRQ3.

Si vous avez des périphériques sur vos ports séries qui partagent une IRQ avec votre modem vous allez avoir des problèmes. Vous devez vous assurer que le port série de votre modem est sur sa propre et unique IRQ. De nombreuses cartes séries modernes (et les ports séries des cartes mères de bonne qualité) permettent de changer les IRQ des ports série.

Si vous utilisez le noyau Linux 2, vous pouvez afficher des IRQ actuellement utilisés avec `cat /proc/interrupts`, qui produira une sortie ressemblant à :

```
 0:    6766283   timer
 1:      91545   keyboard
 2:          0   cascade
 4:     156944 + serial
 7:     101764   WD8013
10:     134365 + BusLogic BT-958
13:          1   math error
15:    3671702 + serial
```

Cela montre que le le port série en IRQ4 (une souris) et le port série en IRQ15 (un modem permanent basé sur PPP pour une liaison Internet). (Il y a également un port série sur le port COM2, IRQ3 et COM4 est dans l'IRQ14, mais comme ils ne sont pas utilisés, ils ne sont pas affichés).

Avertissement - vous devez vous rendre compte de ce que vous faites lorsque vous voulez jouer avec vos IRQ ! Non seulement, vous devez ouvrir votre ordinateur, retirer vos cartes et jouer avec les jumpers, mais vous devez savoir ce qu'il y a sur chaque IRQ. Dans mon cas, j'ai un PC totalement SCSI, et j'ai donc désactivé les interfaces IDE de la carte-mère qui utilisent normalement les IRQ14 et 15 !

Je tiens également à préciser que si votre PC utilise un autre système d'exploitation, changer les IRQ pourrait bien signifier que votre OS ne bootera plus correctement, ou plus du tout !

Si vous déplacez vos ports séries vers des IRQ non-standard, vous devez spécifier à Linux quel IRQ utilise chaque port. `setserial` permet de le faire, et de préférence lors du processus de lancement dans `rc.local` ou `rc.serial` qui est appelé dans `rc.local` ou comme une partie de l'initialisation SysV system. Pour la machine présentée ci-dessus, les commandes utilisées sont

```
/bin/setserial -b /dev/ttyS2 IRQ 11
/bin/setserial -b /dev/ttyS3 IRQ 15
```

Toutefois, si vous utilisez le module série chargé dynamiquement au besoin avec le processus `kerneld`, vous ne pouvez pas définir et perdre les IRQ à chaque redémarrage. Si le module série est effacé, Linux perds alors les configurations spéciales.

Donc, si vous chargez le module série à la demande, vous devez reconfigurer les IRQ à chaque fois qu'il est chargé.

9.1 Remarque au sujet des ports séries et des capacité de vitesse

Si vous utilisez un modem haut débit (externe) (14.400 Bits par seconde ou plus), votre port série doit être capable de suivre le débit qu'un tel modem est capable de produire, en particulier si le modem compresse les données.

Cela demande que votre port série utilise un UART(Universal Asynchronous Receiver Transmitter) moderne comme un 16550(A). Si vous utilisez une vieille machine (ou une vieille carte série), il est tout à fait possible que votre port série ait seulement un UART 8250, ce qui amènera des problèmes considérables avec un modem haute vitesse.

Utiliser la commande

```
setserial -a /dev/ttySx
```

Pour que Linux vous affiche le type d'UART que vous avez. Si vous n'avez pas un UART de type 16550, acheter une nouvelle carte série (disponible pour moins de 300 F). Quand vous acheterez une nouvelle carte, assurez vous que vous pouvez changer les IRQ !

Remarque : les premières versions des puces UART 16550 avaient une erreur. Elle a été rapidement découverte et une mise à jour de la puce est sortie - l'UART 16550A. Un nombre relativement faible de puces erronées ont cependant été mises en circulation. Ça serait de la malchance que d'un récupérer une, en particulier sur certaines cartes d'occasion, mais il suffit de chercher la réponse qui s'appelle 16550A.

9.2 Noms des ports série

Historiquement, Linux utilise les périphériques `cuaX` pour les appels sortants et `ttySx` pour les appels entrants.

Le code du noyau a eu besoin de changer cela dans la version 2.0.x du noyau, et vous devez maintenant utiliser `ttySx` pour les appels entrants et sortants. Cela semble annoncer que les noms de périphériques `cuaX` devraient disparaître dans les futures versions du noyau.

9.3 Configuration de votre modem

Vous devez configurer votre modem correctement pour PPP - pour cela, **LISEZ LE MANUEL DE VOTRE MODEM** ! La plupart des modems ont **une configuration par défaut en usine** avec les options requises par PPP. La configuration minimale spécifie :

- Contrôle de flux matériel (RTS/CTS) (`&`K3 sur les modems Hayes)

Les autres paramètres (en commandes Hayes standard) que vous devez vérifier sont :

- La commande E1 Echo ON (nécessaire pour que `chat` fonctionne)
- S0=0 Auto Answer OFF (à moins que vous ne vouliez que votre modem réponde au téléphone)
- `&`C1 Carrier Detect ON seulement après la connexion
- `&`S0 Data Set Ready (DSR) toujours ON
- (variable) Data Terminal Ready

Un site fournit la configuration des modems pour une variété de fabricants et de modèles de modems croissante à *Modem setup information* `http://www.in.net/info/modems/index.html`

qui devrait vous simplifier la configuration.

Il est aussi intéressant de vérifier comment fonctionne l'interface série entre votre ordinateur et votre modem. La plupart des modems modernes vous permettent d'utiliser l'interface série à une vitesse FIXE permettant à l'interface avec la ligne téléphonique de changer de vitesse pour atteindre la plus haute vitesse que les deux modems peuvent supporter.

Cela s'appelle l'opération de division de la vitesse. Si votre modem le supporte, bloquer l'interface série du modem à la vitesse disponible la plus rapide (d'habitude 115.200 bauds mais parfois 38.400 pour les modems 14.400).

Utiliser les outils de communication (c.-à-d. `minicom` ou `seyon`) pour accéder à la configuration de votre modem, et l'adapter à ce qui est nécessaire à PPP. De nombreux modems renvoient leurs réglages courants après la commande AT&V, mais vous devez consulter le manuel de votre modem.

Si vous avez complètement modifié les réglages, vous pouvez repartir sur des bases saines (généralement) en envoyant AT&F - ça retourne aux réglages usine. (Pour la majorité des modems que j'ai rencontré, les réglages usine on tout ce qu'il faut pour faire du PPP - mais vous devez vérifier).

Un fois que vous avez la commande nécessaire pour configurer votre modem, écrivez-là. Vous devez maintenant prendre une décision : vous pouvez stocker ces paramètres dans la mémoire non-volatile de votre modem, comme cela vous pourrez la relancer en envoyant une commande AT appropriée. Alternativement, vous pouvez passer le paramétrage correct à votre modem lors du processus de connexion de PPP.

Si vous utilisez uniquement votre modem sous Linux pour appeler votre FAI ou un serveur d'entreprise, la manière la plus simple est de sauver la configuration de votre modem dans la mémoire non-volatile.

D'un autre coté, si votre modem est utilisé par d'autres applications et systèmes d'exploitation, il est plus sûr de passer ces informations au modem à chaque appel que vous passez, comme cela, c'est sûr que le modem sera bien configuré lors de l'appel. (Cela a aussi l'avantage de conserver la ligne de configuration du modem au cas où le modem perdrait le contenu de sa NV-RAM, ce qui peut parfois arriver).

9.4 Remarque sur le contrôle de flux série

Lorsque les données transitent sur les lignes de communication série, il peut arriver que les données arrivent plus vite que l'ordinateur ne peut les traiter (l'ordinateur peut être occupé à faire autre chose - n'oubliez pas que Linux est un système Multi-tâches Multi-utilisateurs). Pour s'assurer que des données ne se sont pas perdues (les données n'arrivent pas dans le tampon d'entrée pour se perdre ensuite), certaines méthodes de contrôles sont nécessaires.

Il y a deux façons de faire sur des lignes séries :

- Utiliser des signaux matériels (Clear To Send/Request To Send CTS/RTS)
- Utiliser des signaux logiciel (control S et control Q, aussi appelés XON/XOFF).

Bien que le dernier soit correct pour les liaisons vers des terminaux(texte), les données sur une liaison PPP utilisent les 8 bits - et il est fort probable que quelque part dans les données, un octet se traduis par un control S ou un control Q. Donc, si un modem est configuré pour utiliser le contrôle de flux, les choses vont rapidement s'envenimer !

Pour les liaisons à haut débit utilisant PPP (avec l'utilisation des 8 bits de données) le contrôle de flux matériel est vital, et c'est la raison pour laquelle vous devez utiliser le contrôle de flux matériel.

9.5 Tester votre modem pour composer un numéro

Maintenant que vous en avez terminé avec la configuration du modem et du port série, il serait utile de vérifier si cette configuration marche vraiment en appelant votre fournisseur d'accès et voir si vous pouvez vous connecter.

En utilisant le paquetage du terminal de communication (tel que `minicom`), la commande d'initialisation de votre modem pour PPP, appelez le serveur PPP sur lequel vous voulez vous connecter pour une session PPP.

(Remarque : à ce niveau, nous **n'essayerons pas** d'établir la connexion PPP - mais juste d'être sûr que nous pouvons composer un numéro et ainsi trouver **exactement** ce que le serveur nous envoie pour se connecter et lancer PPP).

Pendant le processus, soit vous capturez (dans un fichier) la totalité du processus de login soit vous notez attentivement (*très attentivement*) **exactement** ce que le serveur envoie pour savoir quand est le moment d'entrer votre nom utilisateur et votre mot de passe (et quelques autres commandes nécessaires pour établir la connexion PPP).

Si votre serveur utilise PAP, vous ne verrez pas de message de login, mais vous verrez à la place la (représentation texte) du protocole de contrôle de liaison (une suite incohérente de caractères) s'afficher sur votre écran.

Quelques mots d'avertissements :

- certains serveurs sont relativement intelligents : vous pouvez vous connecter en utilisant un nom utilisateur /mot de passe basé sur du texte OU en utilisant PAP. Ainsi, si votre fournisseur d'accès ou votre serveur d'entreprise utilise PAP mais n'affiche pas le message directement, cela ne signifie pas que vous avez fait une erreur.
- certains serveurs ont besoin que vous entriez du texte avant de lancer *ensuite* la séquence PAP standard.
- Certains serveurs PPP sont passifs - il attendent et n'envoient rien tant que le client avec lequel il dialogue ne lui envoie un paquet LCP valide. Si le serveur PPP sur lequel vous vous connectez fonctionne en mode passif, vous ne verrez jamais de message !
- Certains serveurs ne lancent pas PPP tant que vous n'avez pas pressé `ENTREE` - essayez donc cela si vous êtes connecté et ne voyez pas de message !

Il est intéressant de dialoguer au moins deux fois - certains serveurs changent leur message d'accueil (c.-à-d. avec le temps !) chaque fois que vous vous connectez. Les deux messages critiques que votre machine Linux doit pouvoir identifier chaque fois que vous vous vous connectez sont :

- le message qui vous demande d'entrer votre nom utilisateur ;
- le message qui vous demande d'entrer votre mot de passe ;

Si vous avez tapé une commande pour lancer PPP sur le serveur, vous devez aussi rechercher le message que le serveur vous envoie une fois que vous êtes connecté pour vous informer que vous pouvez entrer la commande pour lancer ppp.

Si votre serveur lance automatiquement PPP, une fois que vous êtes connecté, vous allez voir des caractères s'afficher sur votre écran - c'est le serveur PPP qui envoie à votre machine les informations pour démarrer et configurer la connexion PPP.

Ça doit donner quelque chose comme ceci :

(et ça ne s'arrête pas !)

Sur certains systèmes, PPP doit être explicitement lancé sur le serveur. Cela vient généralement du fait que le serveur a été configuré pour permettre aux connexions PPP et aux connexions Shell a utiliser les mêmes paires nom utilisateur/mot de passe. Si c'est le cas, envoyez cette commande lorsque vous êtes connecté. Une nouvelle fois vous verrez des caractères que le serveur envoie à la fin du lancement de la connexion PPP.

Si vous ne voyez pas directement après la connexion (et le processus de connexion et le lancement du serveur PPP si besoin), tapez **Entrée** pour voir si le serveur PPP démarre...

A ce niveau, vous pouvez raccrocher votre modem (généralement, taper +++ rapidement et envoyer la commande ATHO une fois que votre modem vous a répondu OK).

Si vous n'arrivez pas à faire fontionner votre modem, lisez le manuel du modem, les pages de manuel des programmes de communication, et le Serial HOWTO ! Une fois que vous aurez fait tout cela reprenez l'étape précédente.

10 Configurer la résolution de nom vers l'adresse (DNS)

Alors que les humains aiment donner des noms aux choses, les ordinateurs préfèrent les nombres. Sur un réseau TCP/IP (comme Internet), nous appelons les machines avec un nom particulier et chaque machine se trouve dans un "domaine" particulier. Par exemple, ma station de travail Linux s'appelle **archenland** et se trouve dans le domaine **interweft.com.au**. Son adresse lisible par un humain est donc archenland.interweft.com.au (qui s'appelle le FQDN - nom de domaine totalement qualifié).

Cependant, pour que cette machine soit trouvée par les autres ordinateurs sur Internet, elle est réellement connue par son adresse IP lorsque les ordinateurs communiquent à travers Internet.

Traduire (résoudre) les noms de machines (et de domaine) en nombres réellement utilisés sur Internet est le travail de machines qui offrent le DNS.

Ce qui se passe en fait :

- Votre machine à besoin de savoir l'adresse IP d'un ordinateur particulier. L'application qui a besoin de cette information interroge le resolveur de votre PC Linux pour obtenir cette information;
- Le résolveur interroge le fichier local `/etc/hosts` et/ou les serveurs de noms des domaines qu'il connaît (le comportement exact du résolveur est déterminé dans `/etc/hosts.conf`);
- si la réponse se trouve dans le fichier `hosts`, la réponse est renvoyée;
- si un serveur de nom est spécifié, votre PC interroge alors cette machine
- si la machine DNS connaît déjà l'adresse IP du nom demandé, elle le retourne. Si elle ne le connaît pas, elle interroge un autre serveur de noms sur Internet pour trouver l'information. Le serveur de noms redescend alors les informations au résolveur qui les a demandé - qui les envoie à l'application qui l'a interrogé.

Quand vous utilisez une connexion PPP, vous devez indiquer à votre machine Linux où elle peut récupérer l'adresse IP à partir du nom (résolution des adresses), comme cela **vous pourrez utiliser le nom des machines** mais votre ordinateur pourra traduire ces derniers vers les adresses IP dont il a besoin pour travailler.

Une solution est de rentrer tous les hôtes avec qui vous voulez dialoguer dans le fichier `/etc/hosts` (ce qui est en réalité totalement impossible si vous êtes connecté à Internet); une autre façon est d'utiliser les adresses IP plutôt que les noms (une tâche de mémorisation impossible sauf pour les petits réseaux).

La meilleure façon est de configurer Linux pour qu'il connaisse ou récupère l'information de conversion des noms en adresses - automatiquement. Ce service est fourni par le système DNS. La seule chose à faire est d'entrer l'adresse IP des serveurs de noms dans le fichier /etc/resolv.conf.

10.1 Le fichier `/etc/resolv.conf`

Le support client de votre serveur PPP doit vous fournir deux adresses IP de DNS (un seul est nécessaire - mais deux permettent une redondance en cas d'échec).

Comme c'est mentionné plus haut, Linux ne peut pas définir l'adresse IP de son serveur de noms comme MS Windows 95 le fait. Ainsi, vous devez **insister** (poliment) auprès de votre FAI pour qu'il vous fournisse cette information !

Votre `/etc/resolv.conf` doit ressembler à quelque chose comme :

```
domain votre.nom.de.domaine
nameserver 10.25.0.1
nameserver 10.25.1.2
```

Éditer ce fichier (le créer si nécessaire) pour contenir les informations que votre FAI vous a donné. Il doit avoir les droits et les permissions comme suit :

```
-rw-r--r--   1 root     root           73 Feb 19 01:46 /etc/resolv.conf
```

Si vous avez déjà configuré un `/etc/resolv.conf` car vous êtes sur un réseau local, ajoutez simplement les adresses IP du serveur DNS PPP dans votre fichier existant.

10.2 Le fichier `/etc/hosts.conf`

Vous devez également vérifier que votre fichier `/etc/hosts.conf` est correctement configuré. Il doit ressembler à

```
order hosts,bind
multi on
```

Il indique au résolveur d'utiliser les informations du fichier host avant d'envoyer les requêtes pour une résolution DNS.

11 Utiliser PPP et les privilèges root

Puisque PPP à besoin de configurer les périphériques réseau, modifier les tables de routage entre autres, il a besoin des privilèges **root**.

Si des utilisateurs autres que **root** doivent lancer des connexions PPP, le programme `pppd` doit avoir l'uid **root** :

```
-rwsr-xr-x   1 root     root         95225 Jul 11 00:27 /usr/sbin/pppd
```

Si `/usr/sbin/pppd` n'est pas défini comme cela, alors **en étant root** taper la commande :

```
chmod u+s /usr/sbin/pppd
```

Cela permet à `pppd` de fonctionner avec les privilèges `root` **même** i si le programme est lancé par un utilisateur ordinaire. Ainsi `pppd` a les bons privilèges pour configurer les interfaces réseau et les tables de routage.

Les programmes qui utilisent le uid root sont autant de trous de sécurité potentiels et vous devez être extrêmement vigilant par rapport aux programmes uid root. De nombreux programmes (comme `pppd`) ont été écrits minutieusement pour minimiser le danger d'utiliser suit root, vous devez donc être tranquille avec celui-ci (mais sans garanties totales).

Selon ce que vous voulez faire de votre système - particulièrement si vous voulez qu'un utilisateur puisse initialiser une liaison PPP, vous devez mettre vos scripts `ppp-on/off` lisibles et exécutables par tout le monde. (c'est sans doute génial si vous êtes le SEUL utilisateur du PC).

Cependant, si vous NE voulez PAS que n'importe qui puisse lancer une connexion PPP (par exemple, votre enfant qui a un compte sur votre PC Linux et que vous ne voulez pas qu'il se connecte à Internet sans votre permission), vous devez faire un groupe PPP (en tant que `root`, éditer le fichier `/etc/group`) et :

- Rendre `pppd` 'suid root', possédé par l'utilisateur **root** et le groupe PPP, avec ces nouvelles permissions sur le fichier. Il doit ressembler à

  ```
  -rwsr-x---   1 root     PPP        95225 Jul 11 00:27 /usr/sbin/pppd
  ```

- Mettre les scripts `ppp-on/off` possédés par l'utilisateur **root** et le groupe **PPP**
- Rendre le script `ppp-on/off` lisible/exécutable par le groupe PPP

  ```
  -rwxr-x---   1 root     PPP           587 Mar 14  1995 /usr/sbin/ppp-on
  -rwxr-x---   1 root     PPP           631 Mar 14  1995 /usr/sbin/ppp-off
  ```

- Ajouter les utilisateurs qui utilisent PPP dans le groupe PPP dans `/etc/group`

Même si vous faites cela, les utilisateurs normaux ne pourront pas fermer la connexion avec un programme ! Utiliser `ppp-off` nécessite les privilèges **root**. Même si n'importe quel utilisateur peut éteindre le modem (ou déconnecter la ligne téléphonique sur un modem interne).

Un alternative (et une meilleure méthode) de faire cela est d'utiliser le programme `sudo`, qui offre une meilleure sécurité et vous permettra de choisir quels utilisateurs (autorisés) pourrons activer/désactiver la liaison en utilisant des scripts. Utiliser `sudo` permet aux utilisateurs autorisés d'activer/désactiver la liaison PPP de façon propre et sécurisée.

12 Configurer les fichiers de connexion de PPP

Maintenant, vous devez être connecté en tant que **root** pour créer les répertoires et éditer les fichiers nécessaires à la configuration de PPP, même si vous souhaitez que PPP soit utilisable par tous les utilisateurs.

PPP utilise un certain nombre de fichiers pour configurer les connexions. Ceux-ci diffèrent dans leur nom et les chemins entre PPP 2.1.2 et 2.2.

Pour PPP 2.1.2 les fichiers sont :

```
/usr/sbin/pppd          # le programme PPP
/usr/sbin/ppp-on        # le script d'appel/connexion
/usr/sbin/ppp-off       # le script de déconnexion
/etc/ppp/options        # les options utilisées pour toutes le connexions
/etc/ppp/options.ttyXX  # les options spécifiques pour ce connecter sur ce port
```

Pour PPP 2.2. les fichiers sont :

```
/usr/sbin/pppd                   # Le programme PPP
/etc/ppp/scripts/ppp-on          # le script d'appel connexion
/etc/ppp/scripts/ppp-on-dialer   # première partie du script d'appel
/etc/ppp/scripts/ppp-off         # le script chat réel
/etc/ppp/options                 # options utilisées par pppd pour toutes les
```

```
                                # connexions
/etc/ppp/options.ttyXX          # options spécifiques pour se connecter sur ce
                                # port
```

Je rappelle aux utilisateurs de la RedHat Linux que dans l'installation standard de la Red Hat 4.X les scripts sont placés dans `/usr/doc/ppp-2.2.0f-2/scripts`.

Dans votre répertoire `/etc`, il devrait y avoir un répertoire **ppp** :

```
drwxrwxr-x   2 root     root         1024 Oct  9 11:01 ppp
```

S'il n'existe pas - le créer avec ces possessions et permissions.

Si le répertoire existe déjà, il devrait contenir un fichier d'options modèle appelé **options.tpl**. Ce fichier est inclus plus bas au cas où il n'existerait pas.

Imprimez-le il contient une explication de presque toutes les options de PPP et c'est pratique de le lire en même temps que les pages de manuel de ppp). Même si vous pouvez utiliser ce fichier comme base à votre `/etc/ppp/options`, il est sans doute mieux de créer votre propre fichier d'options qui ne contiendra pas tous les commentaires du modèle - il sera bien plus court, et plus facile à lire/maintenir.

Si vous avez une ligne série/modems multiple (c'est typiquement le cas des serveurs PPP), créer un fichier général `/etc/ppp/options` contenant les options communes à tous les ports séries dans lequel vous supportez la composition et la réception d'appels et un fichier de configuration individuel pour chaque ligne série sur laquelle vous établirez la configuration individuelle nécessaire pour chaque port.

Ces fichiers d'options spécifiques aux ports sont appelés `options.ttyx1`, `options.ttyx2` et ainsi de suite (ou x est la lettre appropriée pour votre port série).

Toutefois, pour une connexion simple, vous pouvez tranquillement utiliser le fichier `/etc/ppp/options`. Vous pouvez aussi mettre toutes les options comme argument de la commande `pppd` elle-même.

Il est plus simple de maintenir une configuration qui utilise les fichiers `/etc/ppp/options.ttySx`. Si vous utilisez PPP sur différents sites, vous pouvez créer un fichier d'options pour chaque site dans `/etc/ppp/options.site`, et ainsi spécifier le fichier d'options en paramètre à la commande PPP lorsque vous vous connectez (en utilisant l'option `file option-file` de `pppd` dans la ligne de commande de `pppd`).

12.1 Le fichier options.tpl fourni

Certaines distributions de PPP semblent avoir perdu le fichier options.tpl, donc voici le fichier complet. Je suggère de NE PAS éditer ce fichier pour créer votre fichier(s) `/etc/ppp/options`. Copiez plutôt celui-ci dans un nouveau fichier et éditez-le alors. Si vous vous trompez dans votre édition, vous pourrez toujours revenir en arrière en recommençant sur l'original.

```
# /etc/ppp/options -*- sh -*- fichiers d'options generales pour pppd
# cree le 13-Jul-1995 jmk
# autodaté: 01-Aug-1995
# autotime: 19:45
# adaptation francaise par Antoine Levavasseur levavass@efrei.fr

# Utilise l'executable ou la commande shell specifiee pour configurer la ligne
# serie. Ce script va en fait utiliser le programme "chat" pour appeler le
# modem et le debut de la liaison distante ppp.
#connect "echo Vous devez installer une commande connect."
```

```
# Lance l'executable ou la commande shell specifie, une fois que ppp a termine
# la liaison. Ce script va, par exemple, envoyer les commandes au modem pour le
# faire raccrocher si les signaux de controles matériels ne sont pas
# disponibles.
#disconnect "chat -- \d+++\d\c OK ath0 OK"

# table de caractere async -- 32-bit hex; chacun des bit est un caractere qui
# doit etre echappe pour que pppd le recoive.  0x00000001 represente '\x01',
# et 0x80000000 represente '\x1f'.
#asyncmap 0

# Demande a votre correspondant de s'authentifier avant de pouvoir envoyer ou
# recevoir des paquets reseaux
#auth

# Utilise le controle de flux materiel (i.e. RTS/CTS) pour controler le flux
# de donnees du port série
#crtscts

# Utilise le controle de flux logiciel (i.e. XON/XOFF) pour controler le flux
# de donnees du port serie.
#xonxoff
# Ajouter une route par defaut aux tables de routage systeme, pour utiliser
# votre correspondant comme gateway, lorsque la negociation ICP reussi. Cette
# entree est supprimee lorsque la connexion PPP se termine.
#defaultroute

# Specifie que certains caracteres doivent etre echapes sur la transmission
# (en fonction de la facon dont le client souhaite qu'ils soient echappes avec
# la table de controle async). Les caracteres qui sont echappes sont specifies
# comme une liste de nombres hexa separes par des guillemets. La plupart des
# caracteres peuvent etres specifies avec l'option escape, a moins que l'option
# asyncmap soit activee puisque dans ce cas seuls les caracteres de controle
# peuvent etres specifies. Les caracteres qui ne peuvent etre echapes ont les
# valeurs hexa 0x20 - 0x3f ou 0x5e.
#escape 11,13,ff

# Ne pas utiliser les lignes de controle du modem
#local

# Specifie que pppd doit utiliser un verouillage du peripherique serie style
# UUCP pour s'en assurer un acces exclusif.
#lock

# Utilise les lignes de controle du modem. Sur un Ultrix, cette option implique
# un controle de flux materiel, comme avec l'option crtscts. (Cette option
# n'est pas completement implementee.)
#modem
```

```
# Definit la valeur MRU [Unite de Reception Maximum] a n pour la negociation.
# pppd demandera alors de se faire renvoyer les paquets pour au plus <n> octets
# La valeur MRU minimale est de 128. La valeur par defaut est 1500. La valeur
# 296 est recommandee pour les liaisons lentes (40 octets pour l'entete TCP/IP
# + 256 octets de donnees).
#mru 542

# Definit le masque reseau avec <n> qui est un masque reseau 32 bit en
# "chiffres decimaux" note (e.g 255.255.255.0).
#netmask 255.255.255.0

# Desactive le comportement par defaut lorsque aucune adresse IP n'est
# specifiee, et qu'elle est determine (quand c'est possible) a partir du
# hostname. Avec cette option, votre correspondant devra vous fournir votre
# adresse IP locale pendant la negotiation IPCP (a moins que cela ne soit
# specifie explicitement dans la ligne de commande ou un fichier d'options)
#noipdefault

# Active l'option passive sur le LCP. Avec cette option, pppd, essayera
# d'initier la connexion; si aucune reponse n'est recue de votre correspondant,
# pppd attendra passivement un packet LCP (au lieu de s'arreter comme il fait
# sans cette option)
#passive

# Avec cette option, pppd ne va pas transmettre les paquets LCP pour
# initialiser la connexion tant qu'un paquet LCP valide n'est pas envoye
# par votre correspondant (comme avec l'option "passive" pour des versions
# plus anciennes de pppd).
#silent

# Ne requiert et ne permet pas de negociation pour n'importe quelle option de
# LCP et IPCP (utilise  les valeurs par defaut).
#-all

# Desactive la negociation de la compression d'adresses/controle (utilise les
# reglages par defaut, i.e. le champ adresse/controle desactive)
#-ac

# Desactive la negociation asyncmap (utilise le reglage asyncmap par defaut,
# i.e. echappement de tout les caracteres de controle)
#-am

# Ne forke pas pour devenir un processus en tache de fond (sinon pppd le fera
# si un peripherique serie est specifie).
#-detach

# Desactive la negociation d'adresses IP (avec cette option, l'adresse IP
# distante doit etre specifiee avec une option en ligne de commande ou dans
# un fichier d'options).
#-ip
```

```
# Desactive la negociation des nombres magiques. Avec cette option, pppd ne
# peut pas detecter une ligne avec une boucle.
#-mn

# Desactive la negociation MRU [Unite de Reception Maximum] (utilise la valeur
# par defaut, i.e. 1500).
#-mru

# Desactive la negociation du protocole de compression par champ (utilise la
# valeur par defaut, i.e le protocole de compression par champs desactive)
#-pc

# Impose a votre corresondant de s'autentifier avec PAP. Cela necessite une
# authentification des DEUX COTES - NE PAS utiliser pour une liaison standard
# authentifiee par PAP vers un FAI puisqu'il faudra que la machine du FAI
# soit authentifie aupres de votre machine (et ce qu'elle ne pourra pas faire).
#+pap

# Refuse d'utiliser l'authentification avec PAP.
#-pap

# Oblige votre correspondant a s'authentifier avec une authentification
# CHAP [Cryptographic Handshake Authentication Protocol]
# Cela necessite une authentification des DEUX COTES - ne PAS utiliser pour une
# liaison avec un FAI puisqu'il faudra que la machine du FAI
# soit authentifie aupres de votre machine (et ce qu'elle ne pourra pas faire).
#+chap

# Refuse d'utiliser l'authentification avec CHAP.
#-chap

# Desactive la negociation de la compression de l'entete IP style Van Jacobson
# (utilise la valeur par defaut, i.e. pas de compression).
#-vj
# Augmente de niveau de deboguage (comme -d). Si cette option est passee, pppd
# enverra dans le log le contenu de tous les paquets de controle envoyes ou
# recus dans une forme lisible. Les paquets seront envoyes dans syslog avec les
# possibilites des daemons et du niveau de deboguage. Ces informations peuvent
# etre diriges dans un fichier en configurant /etc/syslog.conf correctement
# (voir syslog.conf(5)). (Si pppd est compile avec des options de deboguage
# supplementaires, il enverra les messages de log dans local2 au lieu du
# daemon).
#debug

# Ouvre le nom de domaine <d> pour permettre l'authentification de nom d'hote
# local. Par exemple, si gethostname() retourne le nom porsche,
# mais le nom de domaine qualifie est porsche.Quotron.COM, vous devriez
# utiliser l'option domain pour mettre le nom de domaine a Quotron.COM.
#domain <d>
```

```
# Active le code de deboguage du pilote PPP au niveau du noyau. L'argument n
# est un nombre qui est la somme des valeurs suivantes : 1 active les messages
# de deboguage generaux, 2 pour que le contenu des paquets recus soient
# affiches, et 4 pour que le contenu des paquets envoyes soient affiches.
#kdebug n

# Configure la valeur MTU [Unitee de Transmission Maximum] a <n>. A moins
# que votre correspondant ne demande une valeur plus petite lors de la
# negociation MRU, pppd demande que les codes reseaux du noyau envoie des
# paquets d'au plus n octets dans l'interface reseau PPP.
#mtu <n>

# Definit le nom du systeme local avec <n> dans un but d'authentification.
# Cela doit certainement avoir ete configure avec votre nom utilisateur de votre
# FAI si vous utilisez PAP/CHAP.
#name <n>

# Definit le nom utilisateur avec <u> pour l'authentification avec le
# correspondant utilisant PAP .
# Ne pas utiliser si vous utilisez le 'name' au-dessus !
#user <u>

# Force l'utilisation du nom d'hote comme nom du systeme local pour les
# authentification (masque l'option name).
#usehostname

# Configure le nom du systeme distant pour les authentifications a <n>.
#remotename <n>

# Ajoute une entree au systeme des tables ARP [Protocole de Resolution
# d'Adresses] avec l'adresse IP de votre correspondant et l'adresse Ethernet
# de ce syteme.
#proxyarp

# Utilise le systeme de base de donnes de mots de passe pour l'authentification
# de votre correspondant avec PAP.
#login

# Si cette option est passee, pppd va envoyer une requete-echo LCP a votre
# correspondant toutes les n secondes. Sous Linux, une requete-echo est envoyee
# lorsqu'aucun paquet n'a ete recu du correspondant depuis n secondes.
# Normalement, le correspondant doit repondre a la requete-echo en envoyant une
# reponse-echo. Cette option peut etre utilise avec l'option lcp-echo-failure
# pour detecter que le correspondant n'est plus connecte.
#lcp-echo-interval <n>

# Si cette option est passe, pppd va supposer que le correspondant est mort
# si n requetes-echo LCP sont envoyees sans recevoir de reponse-echo LCP
# valide. Si cela arrive, pppd va terminer la connexion. Utiliser cette option
```

```
# necessite une valeur non nulle pour le parametre lcp-echo-interval.
# Cette option peut etre utilisee pour permettre a pppd de terminer une fois
# qu'une connexion pppd ait ete cassee (c.-a-d. le modem a ete raccroche) dans
# les situations ou aucune ligne de controle materiel du modem n'est
# disponibles
#lcp-echo-failure <n>

# Definit l'intervale relance LCP (delay de retransmission) a <n> secondes
# (defaut 3).
#lcp-restart <n>

# Definit le nombre maximal de transmission de requete-fin de LCP a <n>
# (defaut 3).
#lcp-max-terminate <n>

# Definit la nombre maximum de transmission de requete-configure LCP a <n>
# (defaut 10).
# Certains serveurs PPP sont lents a demarrer. Vous devrez alors augmenter
# ceci si vous avez des erreurs 'serial line looped back' et que vous etes
# SUR que vous etes connecte correctement et que PPP doit demarrer sur le
# serveur.
#lcp-max-configure <n>

# Definit le nombre maximum de retours LCP configure-NAKs avant de commencer
# a envoyer des configure-Rejects a la place a <n> (defaut 10).
#lcp-max-failure <n>

# Definit l'intervale de relancce de IPCP (temps de retransmission) a <n>
# secondes (defaut 3).
#ipcp-restart <n>

# Definit le nombre maximum de transmission de requete-fin IPCP a <n>
# (defaut 3).
#ipcp-max-terminate <n>

# Definit la nombre maximum de transmission de requete-configure IPCP a <n>
# (defaut 10).
#ipcp-max-configure <n>

# Definit la nombre maximem de retour configure-NAKs IPCP avant de commencer
# a envoyer des configure-Rejects a la place a <n> (defaut 10).
#ipcp-max-failure <n>

# Definit l'intervalle de relance PAP (temps de retransmission) a <n> secondes
# (defaut 3).
#pap-restart <n>

# Definit le nombre maximum de transmission de requete-d'authentification PAP
# a n (defaut 10)
#pap-max-authreq <n>
```

```
# Definit les intervales de relance de CHAP (temps de retransmission) a <n>
# secondes (defaut 3).
#chap-restart <n>

# Definit le nombre maximal d'essai de  transmission de CHAP a <n>
# (defaut 10).
#chap-max-challenge

# Si cette option est passee, pppd va relancer votre correspondant toutes les
# <n> secondes.
#chap-interval <n>

# Avec cette option, pppd va accepter la proposition de votre correspondant
# pour votre adresse IP locale, meme si l'adresse IP est specifiee en option.
#ipcp-accept-local

# Avec cette option, pppd va accepter la proposition d'adresse IP (distante),
# meme si l'adresse distante est specifiee en option.
#ipcp-accept-remote
```

12.2 Quelles options dois-je utiliser ? (Sans PAP/CHAP)

En fait, comme toujours, ça dépend. Les options spécifiées ici devraient marcher avec la plupart des serveurs.

Cependant, si ça NE fonctionne PAS LISEZ LE FICHIER MODELE (`/etc/ppp/options.tpl`) **et** les pages de manuel de `pppd` **et** demandez à votre administrateur/support client qui fait marcher le serveur sur lequel vous vous connectez.

Je rappelle également que les scripts de connexion présentés ici utilisent également des options en ligne de commande pour `pppd` et rendre les choses plus facilement modifiables.

```
# /etc/ppp/options (NO PAP/CHAP)
#
# Empeche pppd de forker en tache de fond
-detach
#
# utilise les lignes de controle du modem
modem
# utilise les verrous style uucp pour assurer une utilisation exclusive du
# port serie
lock
# utilise le flux de controle materiel
crtscts
# creer une route par defaut pour cette connexion dans la table de routage
defaultroute
# ne configure aucune sequence de controle echapee
asyncmap 0
# utlise une taille maximale de paquet transmis de 552 octets
mtu 552
# utilise une taille maximale de paquets recus de 552 octets.
```

```
mru 552
#
#-------FIN D'EXEMPLE /etc/ppp/options (sans PAP/CHAP)
```

13 Si votre serveur PPP utilise PAP (Password Authentification Protocol)

Si le serveur sur lequel vous vous connectez a besoin de l'authentification PAP ou CHAP, vous avez un petit peu plus de travail.

Pour les fichiers de configuration précédents, ajouter les lignes suivantes :

```
#
# force pppd a utiliser le nom utilisateur de FAI comme 'nom d'hote' pendant
# le processus d'authentification
name <le nom de votre FAI>       # vous devez editer cette ligne
#
# Si vous avez un *serveur* PPP vous devez forcer PAP ou CHAP en decommentant
# l'une des lignes suivantes. Ne pas les utiliser si vous etes un client qui
# se connecte sur un serveur PPP (meme s'il utilise PAP ou CHAP) puisque cela
# forcera le serveur a s'authentifier aupres de votre machine (alors que cela
# ne sera certainement pas possible - et la liaison echouera).
#+chap
#+pap
#
# Si vous utilisez des secrets ENCRYPTES dans le fichier /etc/ppp/pap-secrets,
# alors decommentez la ligne suivante.
# Remarque: ce n'est pas pareil que d'utiliser les mots de passe MS encryptes
# comme ils peuvent etres avec MS RAS sur Windows NT.
#+papcrypt
```

13.1 Utiliser MSCHAP

Microsoft Windows NT RAS peut être configuré pour utiliser une variation particulière de CHAP (Challenge/Handshake Authentication Protocol). Dans l'archive des sources de PPP, vous devriez trouver un fichier appelé README.MSCHAP80 qui parle de cela.

Vous pouvez déterminer si le serveur demande une authentification utilisant ce protocole en activant le deboguage dans `pppd`. Si le serveur demande une authentification MS CHAP, vous devriez voir les lignes

```
rcvd [LCP ConfReq id=0x2 <asyncmap 0x0> <auth chap 80> <magic 0x46a3>]
```

L'information critique ici est **auth chap 80**.

Pour utiliser MS CHAP, vous devez recompiler pppd. Veuillez consulter les instructions dans le fichier README.MSCHAP80 des sources de PPP pour obtenir les instructions sur la façon de compiler en utilisant cette variante.

Vous devez remarquer que le code présent supporte uniquement des clients Linux PPP connectés sur un serveur MS Windows NT. Mais NE supporte PAS la configuration d'un serveur PPP Linux pour utiliser l'authentification MSCHAP80 pour les clients.

13.2 Le fichier de secrets de PAP/CHAP

Si vous utilisez l'authentification PAP ou CHAP, alors vous devez créer les fichiers . Ce sont

```
/etc/ppp/pap-secrets
/etc/ppp/chap-secrets
```

Il doivent être possédés par **root**, le groupe **root** et avoir les permissions 740 pour la sécurité.

Le premier point à remarquer au sujet de PAP et CHAP est qu'ils sont conçus pour authentifier **les ordinateurs** mais pas **les utilisateurs**

"Humm? Qu'est ce que ça change ? " Me demandez vous.

Hé bien, une fois que l'ordinateur a établi une connexion PPP avec le serveur, **n'importe quel** utilisateur de votre système peut utiliser cette connexion - pas seulement vous. C'est pourquoi vous pouvez configurer un réseau à large domaine en reliant deux réseaux locaux avec PPP.

PAP peut (et CHAP **DOIT**) demander une authentification **bidirectionnelle** - c'est un nom et un code secret qui est nécessaire pour chaque ordinateur avec lequel il est impliqué. Cependant, ce n'est pas la façon dont tous les serveurs PPP fonctionnent pour offrir l'authentification PAP des appels PPP.

Il a été dit que votre FAI vous donnerait certainement un nom utilisateur et un mot de passe pour vous permettre de vous connecter sur son système et ensuite sur Internet. Votre FAI se fiche de connaître le nom de votre ordinateur, vous devrez donc utiliser votre nom utilisateur chez votre FAI comme nom pour votre machine.

Cela est possible avec l'option `name username` de `pppd`. Donc, si vous utilisez le nom donné par votre FAI, ajoutez la ligne

```
name votre_utilisateur nom_chez_votre_FAI
```

dans votre fichier `/etc/ppp/options`.

Techniquement, vous devrez utiliser `user votre_utilisateur nom_chez_votre_FAI` pour PAP, mais `pppd` est suffisamment intelligent pour interpréter l'option `name` comme `user` si nécessaire pour utiliser PAP. L'avantage d'utiliser l'option `name` est qu'elle est également valide pour CHAP.

Puisque PAP permet d'authentifier les **ordinateurs**, techniquement, vous devez aussi spécifier un nom d'ordinateur distant. Toutefois, comme de nombreuses personnes ont plus d'un seul FAI, vous pouvez utiliser un caractère générique (*) pour le nom d'ordinateur distant dans le fichier de secrets.

Il est aussi intéressant de remarquer que de nombreux FAI utilise des banques de modems multiples connectés sur différents serveurs de terminaux - tous ont un nom différent, mais sont accessibles avec un seul numéro d'appel. Il peut alors être difficile dans certaines circonstances de savoir à l'avance quel est le nom de l'ordinateur distant, puisque cela dépendra du serveur de terminal sur lequel vous vous connecterez !

13.3 Le fichier de secrets de PAP

Le fichier `/etc/ppp/pap-secrets` ressemble à

```
# Authentication Secrete avec PAP
# client        serveur       secret     adresses_IP_locale_acceptable
```

Les quatre champs sont séparés par des espaces et le dernier peut être omis (c'est l'adresse dynamique ou sans doute statique de votre FAI).

Supposons que votre FAI vous ait donné un nom utilisateur `fred` et un mot de passe `flintstone` vous aurez l'option `name fred` dans `/etc/ppp/options[.ttySx]` et vous configurerez votre `/etc/ppp/pap-secrets` comme suit

```
# Authentication Secrete avec PAP
# client        serveur       secret     adresses_IP_locale_acceptable
fred            *       flintstone
```

Cela signifie que pour une machine locale `fred` (`pppd` lui impose de s'appeler comme cela même si ça n'est pas son nom de machine réel) et pour **n'importe quel** serveur, utiliser le mot de passe (secret) `"flintstone"`.

Remarquez que vous ne devez pas spécifier d'adresse IP locale, à moins que vous ne deviez FORCER une adresse locale statique particulière. Même si vous essayez cette option, il y a peu de chances qu'elle marche puisque la plupart des serveurs PPP (pour des raisons de sécurité) ne permettent pas au système distant de configurer l'adresse IP qu'ils reçoivent.

13.4 Le fichier de secrets de CHAP

Cette méthode nécessite une méthode authentification mutuelle - c'est à dire que vous devez permettre à votre machine d'authentifier le serveur distant **ET** le serveur distant d'authentifier votre machine.

Donc, si votre machine s'appelle `fred` et que le serveur distant s'appelle `barney` il faut définir sur fred `name fred remotename barney` et sur la machine distante `name barney remotename fred` dans les fichiers `/etc/ppp/options.ttySx` respectifs.

Le fichier `/etc/chap-secrets` de `fred` ressemblera à

```
# Authentication Secrete avec CHAP
# client        serveur       secret     adresses_IP_locale_acceptable
fred            barney      flintstone
barney          fred        wilma
```

et pour `barney`

```
# Authentication Secrete avec CHAP
# client        serveur       secret     adresses_IP_locale_acceptable
barney          fred    flintstone
fred            barney  wilma
```

Remarquez en particulier que les deux machines doivent avoir des entrées pour une authentification bidirectionnelle. Cela permet à la machine locale d'authentifier elle-même la machine distante **ET** la machine distante d'authentifier elle-même la machine locale.

13.5 Supporter plusieurs connexions authentifiées avec PAP

Certains utilisateurs ont plus d'un serveur sur lesquels ils se connectent en utilisant PAP. A partir du moment où votre nom utilisateur est différent sur chacune des machines sur lesquelles vous vous connectez, il n'y a pas de problème.

Mais, nombreux sont les utilisateurs qui ont le même nom utilisateur sur deux (ou plus, voire même tous les) systèmes où ils se connectent. Le problème revient donc à choisir la bonne ligne du fichier `/etc/ppp/pap-secrets`.

Comme vous pouvez le supposer, PPP fournit un mécanisme pour régler cela. PPP permet de définir un << pseudonyme >> pour le serveur distant final de la connexion en utilisant l'option **remotename** de pppd.

Supposons que vous vous connectiez sur deux serveurs PPP utilisant le même nom utilisateur `fred`. Vous configurerez votre `/etc/ppp/pap-secrets` avec

```
fred    pppserveur1       barney
fred    pppserveur2       wilma
```

Maintenant, pour se connecter sur `pppserveur1` vous utiliserez `name fred remotename pppserver1` dans votre `ppp-options` et pour `pppserveur2` `name fred remotename pppserver2`.

Puisque vous pouvez sélectionner le fichier d'options à utiliser avec `pppd` en utilisant l'option `file filename`, vous pouvez définir un script pour se connecter sur chacun de vos serveurs PPP, qui choisira le fichier d'options correct et mettra la bonne option `remotename`.

14 Configurer une connexion PPP à la main

Maintenant que vous avez créé les fichiers `/etc/ppp/options` et `/etc/resolv.conf` (et si besoin le fichier `/etc/ppp/pap/chap-secrets`), vous pouvez tester les paramètres en établissant une connexion PPP à la main. (Une fois que la connexion manuelle marchera, nous automatiserons le processus).

Pour ce faire, votre programme de communication doit pouvoir être quitté sans réinitialiser le modem. `Minicom` peut le faire avec Alt-Q (ou dans les versions plus ancienne de `minicom` CTRL-A Q)

Vérifiez que vous êtes **root**.

Lancez le programme de communication (`minicom` par exemple), appelez le serveur PPP et connectez-vous comme d'habitude. Si vous avez besoin d'envoyer une commande pour démarrer PPP sur le serveur, faites-le. Vous devriez maintenant voir les messages que vous aviez déjà vus.

Si vous utilisez PAP ou CHAP, se connecter simplement sur le système distant devrait démarrer PPP et vous devriez voir des caractères s'afficher sans toutefois se connecter (même si ça peut ne pas marcher sur certains serveurs - essayez alors d'appuyer sur **Entrée** et les caractères devraient s'afficher).

Maintenant quittez le programme de communication *sans initialiser le modem* (ALT-Q ou CTRL-A Q dans `minicom`) et au message d'accueil de Linux (étant **root**) tapez

```
pppd -d -detach /dev/ttySx 38400 &
```

L'option -d active le déboguage - la conversation de démarrage de la connexion ppp sera retranscrite dans votre journal système - ce qui est très utile si des problèmes apparaissent.

Les lumières de votre modem devraient alors clignoter puisque la connexion PPP est établie. Cela peut prendre un peu de temps avant que la connexion soit terminée.

A ce niveau, vous pouver afficher l'interface PPP en tapant la commande :

```
ifconfig
```

En plus des éventuels périphériques loopback et Ethernet, vous devriez avoir quelque chose qui ressemble à :

```
ppp0      Link encap:Point-Point Protocol
          inet addr:10.144.153.104  P-t-P:10.144.153.51 Mask:255.255.255.0
          UP POINTOPOINT RUNNING  MTU:552  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0
          TX packets:0 errors:0 dropped:0 overruns:0
```

Où

- inet addr:10.144.153.10 est l'adresse IP de votre connexion.
- P-t-P:10.144.153.5 est l'adresse IP du serveur.

(Naturellement, `ifconfig` ne renvoie pas ces nombres là, mais ceux qui sont utilisés par le serveur PPP.)

Remarque : `ifconfig` indique également que votre liaison fonctionne correctement !

Si vous aucun périphérique PPP n'est affiché ou quelque chose du genre

```
ppp0      Link encap:Point-Point Protocol
          inet addr:0.0.0.0  P-t-P:0.0.0.0  Mask:0.0.0.0
          POINTOPOINT  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0
          TX packets:0 errors:0 dropped:0 overruns:0
```

Votre liaison PPP n'est pas correcte... voir la section suivante sur le débuggage !

Vous devez aussi pouvoir voir une route vers l'hôte distant (et même plus). Pour cela, entrez la commande

```
route -n
```

Vous devriez obtenir quelque chose comme :

```
Kernel routing table
Destination     Gateway         Genmask         Flags MSS    Window Use Iface
10.144.153.3    *               255.255.255.255 UH    1500   0        1 ppp0
127.0.0.0       *               255.0.0.0       U     3584   0       11 lo
10.0.0.0        *               255.0.0.0       U     1500   0       35 eth0
default         10.144.153.3    *               UG    1500   0        5 ppp0
```

Ce qui est important ici, c'est d'avoir DEUX entrées pointant vers votre interface ppp.

La première est une route d'hôte (indiqué par le drapeau H) et qui permet de voir l'hôte sur lequel vous êtes connecté - mais pas plus.

La seconde est la route par défaut (établie en donnant à `pppd` l'option `defaultroute`). C'est cette route qui indique à votre PC Linux d'envoyer tous les paquets qui ne sont pas destinés à vos interfaces Ethernet - vers lesquelles vous avez des routes réseau spécifiques - au serveur PPP lui-même. Le serveur PPP est alors responsable de router vos paquets vers Internet et de vous renvoyer les paquets chez vous.

Si vous ne voyez pas une table de routage avec deux entrées, quelque chose ne marche pas. En particulier si votre syslog contient des messages indiquant que `pppd` ne remplace pas la route par défaut, vous devez avoir une route par défaut qui pointe sur votre interface Ethernet - qui **DOIT** être remplacée par une route réseau spécifique : **VOUS NE POUVEZ AVOIR QU'UNE SEULE ROUTE PAR DEFAUT !!!**

Vous devez explorer vos fichiers d'initialisation de votre système pour trouver où est configuré votre route par défaut (elle doit utiliser la commande `route add default...`). Changer cette commande par quelque chose du genre `route add net ....`

Maintenant testez la liaison en 'pingant' le serveur à l'adresse IP affiché par ifconfig, comme suit :

```
ping 10.144.153.51
```

Vous devriez obtenir

```
PING 10.144.153.51 (10.144.153.51): 56 data bytes
64 bytes from 10.144.153.51: icmp_seq=0 ttl=255 time=328.3 ms
64 bytes from 10.144.153.51: icmp_seq=1 ttl=255 time=190.5 ms
64 bytes from 10.144.153.51: icmp_seq=2 ttl=255 time=187.5 ms
64 bytes from 10.144.153.51: icmp_seq=3 ttl=255 time=170.7 ms
```

Ce listing doit continuer sans jamais s'arrêter - pour le stopper, taper CTRL C, vous recevrez alors des informations complémentaires :

```
--- 10.144.153.51 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 170.7/219.2/328.3 ms
```

Jusqu'ici tout va bien, jusqu'ici tout va bien.

Maintenant, essayez de pinger un nom d'hôte (pas le nom du serveur PPP lui-même) mais un hôte d'un autre site dont vous SAVEZ qu'il y a de très fortes chances qu'il marche...). Par exemple

```
ping sunsite.unc.edu
```

ou lip6.fr

Dans ce cas, il va y avoir une petite pause le temps que Linux obtienne l'adresse IP du nom de hôte que vous avez 'pingé' par le DNS que vous avez spécifié dans le `/etc/resolv.conf` - ne vous inquiétez pas (les lumières de votre modem devraient clignoter). Rapidement, vous devriez avoir l'affichage suivant

```
 PING sunsite.unc.edu (152.2.254.81): 56 data bytes
64 bytes from 152.2.254.81: icmp_seq=0 ttl=254 time=190.1 ms
64 bytes from 152.2.254.81: icmp_seq=1 ttl=254 time=180.6 ms
64 bytes from 152.2.254.81: icmp_seq=2 ttl=254 time=169.8 ms
64 bytes from 152.2.254.81: icmp_seq=3 ttl=254 time=170.6 ms
64 bytes from 152.2.254.81: icmp_seq=4 ttl=254 time=170.6 ms
```

Là encore, arrêtez le en tapant CTRL C pour obtenir les statistiques...

```
--- sunsite.unc.edu ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 169.8/176.3/190.1 ms
```

Si vous n'avez pas de réponse, essayer de pinger l'adresse de votre serveur DNS de votre FAI. Si vous n'avez pas de résultat, alors il doit y avoir un problème avec votre `/etc/resolv.conf`.

Si ça ne fonctionne pas, vous devez avoir un problème de routage, ou votre FAI a un problème à vous renvoyer les paquets. Vérifiez les tables de routages présentées plus haut et si tout est OK, contacter votre FAI. Si cela se trouve après votre FAI, alors le problème est de votre coté.

Si tout marche correctement, fermez la connexion en tapant

```
ppp-off
```

Après une courte pause, le modem devrait raccrocher.

Si ça ne marche pas, soit vous éteignez votre modem, soit vous lancez votre programme de communication et vous interrompez votre modem avec +++ puis vous raccrochez avec ATH0 quand vous recevez le message OK de votre modem.

Vous devez également effacer le fichier créé par pppd

```
rm -f /var/lock/LCK..ttySx
```

15 Automatiser votre connexion - Créer les scripts de connexion

Même si vous pouvez continuer à vous connecter à la main comme montré ci-dessus il est plus sympa de faire quelques scripts pour automatiser tout ça.

Quelques scripts vont automatiser la connexion et le lancement de PPP, vous n'aurez alors plus qu'à envoyer une seule commande (en étant **root** ou membre du groupe PPP) pour lancer votre connexion.

15.1 Scripts de connexion pour l'authentification nom utilisateur/mot de passe

Si votre FAI n'a pas besoin d'utiliser PAP/CHAP, ces scripts sont faits pour vous !

Si le paquetage ppp est correctement installé, vous devez avoir deux fichiers d'exemple. Pour PPP 2.1.2 il sont dans `/usr/sbin` et pour PPP 2.2 il sont dans `/etc/ppp/scripts`. Ils s'appellent

pour PPP-2.1.2

```
ppp-on
ppp-off
```

et pour PPP 2.2

```
ppp-off
ppp-on
ppp-on-dialer
```

Maintenant, si vous utilisez PPP 2.1.2, je vous presse fermement de détruire ces fichiers d'exemple. Il y a pleins de problèmes potentiels avec ceux-ci - et ne me dites pas qu'ils marchent correctement - je les ai également utilisé pendant des années (et les recommandait même dans la première version de ce HOWTO) !

Pour les utilisateurs de PPP 2.1.2, voici de BIEN MEILLEURS modèles, pris dans la distribution 2.2 de PPP. Je vous conseille de les copier et d'utiliser ces scripts à la place des anciens de PPP-2.1.2.

15.2 Le script ppp-on

C'est le premier des DEUX scripts qui lance la connexion.

```
#!/bin/sh
#
# Script pour initialiser une connexion PPP. C'est la premiere partie de deux
# scripts. Ce ne sont pas des scripts securises, puisque les codes sont
# visibles avec la commande 'ps'. Cependant ils sont simples.
#
# Ce sont des parametres a changer selon les besoins
TELEPHONE=555-1212    # Le numero de telephone de la connexion
ACCOUNT=george        # Le nom du compte (comme dans 'George Burns')
PASSWORD=gracie       # Le mot de passe du compte (et 'Gracie Allen')
LOCAL_IP=0.0.0.0      # L'adresse IP locale si elle est connue Dynamic = 0.0.0.0
REMOTE_IP=0.0.0.0     # Adresse IP distante. Normally 0.0.0.0
NETMASK=255.255.255.0 # Le masque reseau si besoin
#
# Les exporter pour qu'ils soit disponible avec 'ppp-on-dialer'
export TELEPHONE ACCOUNT PASSWORD
#
# C'est l'emplacement du script qui compose le numero et se connecte.
# Utilisez les noms de fichiers absolus, car la variable $PATH n'est pas
# utilisee avec l'option connect. (Faire ca en etant 'root' entraine un
# trou de securite.)
#
DIALER_SCRIPT=/etc/ppp/ppp-on-dialer
#
# Commencer la connexion
#
#
exec /usr/sbin/pppd debug /dev/ttySx 38400 \
        $LOCAL_IP:$REMOTE_IP \
        connect $DIALER_SCRIPT
```

Voici le script ppp-on-dialer :

```
#!/bin/sh
#
# C'est la deuxieme partie du script ppp-on. Il contient le protocole de
# connexion de la connexion desiree.
#
/usr/sbin/chat -v                                                  \
        TIMEOUT         3                               \
        ABORT           '\nBUSY\r'                      \
        ABORT           '\nNO ANSWER\r'                 \
        ABORT           '\nRINGING\r\n\r\nRINGING\r'    \
        ''              \rAT                            \
        'OK-+++\c-OK'   ATH0                            \
        TIMEOUT         30                              \
```

```
        OK              ATDT$TELEPHONE                  \
        CONNECT         ''                              \
        ogin:--ogin:    $ACCOUNT                        \
        assword:        $PASSWORD
```

Avec PPP-2.2, le script ppp-off ressemble à :

```
#!/bin/sh
######################################################################
#
# Determine le peripherique a terminer.
#
if [ "$1" = "" ]; then
        DEVICE=ppp0
else
        DEVICE=$1
fi

######################################################################
#
# Si le fichier pid ppp0 est present alors le programme est lance. Le stopper.
if [ -r /var/run/$DEVICE.pid ]; then
        kill -INT `cat /var/run/$DEVICE.pid`
#
# Si le kill ne marche pas, alors il n'y a pas de processus lance pour ce pid.
# Cela peut signifier que le fichier de lock a ete laisse. Vous souhaitez
# peut-etre detruire le fichier de lock en meme temps.
        if [ ! "$?" = "0" ]; then
                rm -f /var/run/$DEVICE.pid
                echo "ERROR: Removed stale pid file"
                exit 1
        fi
#
# Succes. pppd a enlever tout ce qu'il avait cree.
        echo "PPP link to $DEVICE terminated."
        exit 0
fi
#
# Le processus ppp ne marche pas pour ppp0
echo "ERROR: PPP link is not active on $DEVICE"
exit 1
```

15.3 Editer les scripts de démarrage de PPP fournis

Puisque les nouveaux scripts sont en deux parties, nous allons les éditer chacun leur tour.

15.3.1 Le script ppp-on

Vous devez éditer ces fichiers pour contenir VOTRE nom utilisateur de votre FAI, VOTRE mot de passe de votre FAI, et le numéro de téléphone de votre FAI.

Chacune des lignes ressemblant à `TELEPHONE=` définissent une variable du shell contenant les informations à droite du '=' (mise à part les commentaires évidemment). Éditer donc toutes les lignes de ce fichier pour qu'il corresponde a votre FAI et votre connexion.

De plus, si vous configurez l'adresse IP (si vous avez besoin) dans le fichier `/etc/ppp/options`, DETRUISEZ les lignes mentionnant

```
$LOCAL_IP:$REMOTE_IP \
```

Assurez-vous également que la variable shell DIALER_SCRIPT pointe sur le chemin complet du script d'appel qui sera réellement utilisé. Ainsi, si vous déplacez ou renommez ce script, assurez-vous que vous modifier correctement cette ligne dans le script `ppp-on` !

15.3.2 Le script ppp-on-dialer

C'est le deuxième des scripts qui lance votre liaison PPP.

Remarque : un script `chat` est normalement sur une ligne complète. Les backslashs sont utilisés pour permettre à la ligne de se répartir sur plusieurs lignes physiques (pour une lisibilité plus humaine) et ne font pas partie du script lui-même.

Cependant, il est très utile de le regarder en détail, comme cela nous comprenons ce qu'il est réellement supposé faire !

15.4 Ce que script Chat signifie...

Un script `chat` est une séquence de paires de "chaînes d'attentes" et de "chaînes d'envoi ". En particulier, remarquez que nous attendons **TOUJOURS** quelque chose avant d'envoyer quoi que ce soit.

Si nous envoyons quelque chose **SANS** avoir rien reçu avant, nous devons attendre une chaîne vide avant (indiquée par un `""`) Et de la même manière attendre quelque chose sans avoir rien envoyé ! Ainsi, si les chaîne consistent en quelques mots, (c.-à-d. NO CARRIER), vous devez entourez les chaînes par des quotes pour qu'elles soient vues comme une seule entité par chat.

La ligne `chat` de notre modèle est :

```
exec /usr/sbin/chat -v
```

Invoque chat, le -v indique à `chat` de mettre toutes ses entrée/sorties dans le journal système (généralement /var/log/messages). Une fois que votre script `chat` marchera correctement, éditez cette ligne et enlevez le -v pour économiser des messages inutiles dans votre syslog.

```
TIMEOUT         3
```

Ça définit un temps d'attente de trois secondes avant la réception d'une entrée attendue. Vous devrez l'augmenter à 5 ou 10 secondes si vous utilisez un modem extrêmement lent !

```
ABORT           '\nBUSY\r'
```

Si la chaîne BUSY est reçue, abandonner l'opération.

M

```
ABORT           '\nNO ANSWER\r'
```

Si la chaîne NO ANSWER est reçue, abandonner l'opération.

```
ABORT           '\nRINGING\r\n\r\nRINGING\r'
```

Si la chaîne RINGING est reçue de manière répétée, abandonner l'opération. C'est parce que quelqu'un vous appelle !

```
"              \rAT
```

Ne rien attendre du modem et envoyer la chaîne AT.

```
OK-+++\c-OK   ATH0
```

C'est un petit plus compliqué car cela utilise certaines capacités de chat à récupérer les erreurs.

Ça veux dire...Expect OK, si on ne l'a pas reçu (parce que le modem n'est pas en ligne de commande) alors envoyer +++ (la chaîne standard des modems compatible Hayes pour retourner en ligne de commande) et attendre OK. Ensuite envoyer ATH0 (la chaîne pour raccrocher). Cela permet au script de répondre à la situation ou votre modem avait laissé la ligne décrochée !

```
TIMEOUT         30
```

Mettre le temps d'attente à 30 secondes pour le reste du script. Si vous avez des soucis avec des abandons du script `chat` à cause de timeout, augmentez cette valeur à 45 secondes ou plus.

```
OK              ATDT$TELEPHONE
```

Attends un OK (la réponse du modem à la commande ATH0) et compose le numéro que nous voulez appeler.

```
CONNECT         ''
```

Attendre CONNECT(ce que le modem nous envoie lorsque le modem distant décroche) et n'envoie rien en réponse.

```
ogin:--ogin:    $ACCOUNT
```

Une fois encore nous avons une récupération d'erreur ici. Attendre le message de login (...ogin:) mais si nous ne le recevons pas au bout d'un certain temps, envoyer un retour chariot et attendre une nouvelle fois le message de login. Lorsque le message est reçu, envoyer le nom utilisateur (stocké dans la variable shell $ACCOUNT).

```
assword:        $PASSWORD
```

Attendre le message du mot de passe et envoyer le mot de passe (là aussi stocké dans une variable shell).

Ce script `chat` à une capacité de récupération d'erreurs résonnable. Chat possède bien d'autres possibilités que celles détaillés. Pour plus d'informations, consulter la page de manuel de `chat` (`man 8 chat`).

15.4.1 Démarrer PPP sur le serveur final

Bien que le script ppp-on-dialer soit parfait pour les serveurs qui lancent automatiquement `pppd` sur le serveur final une fois que vous êtes connectés, certains serveurs nécessitent que vous lanciez PPP explicitement sur le serveur.

Si vous avez besoin d'envoyer une commande de lancement de PPP sur le serveur, vous DEVEZ éditer le script ppp-on-dialer.

A la fin du script (après la ligne password) ajouter une paire **attente envoi** en plus - elle cherchera un message de login (une suite de caractères qui ont une signification particulière dans le shell Bourne - comme $ et [ou] (les crochets ouverts et fermés).

Une fois que `chat` à obtenu le prompt du shell, `chat` doit envoyer la commande de démarrage de ppp nécessaire sur le serveur PPP de votre FAI.

Pour ma part, mon serveur PPP utilise un prompt Bash standard

```
[hartr@kepler hartr]$
```

et nécessite que je tapes

```
ppp
```

pour lancer ppp sur le serveur.

Une idée judicieuse peut être d'ajouter un peu de récupération d'erreur, dans mon cas j'utilise

```
        hartr--hartr    ppp
```

Cela signifie que si je ne reçois pas le prompt dans le temps imparti, j'envoie un retour chariot et je recherche le prompt une nouvelle fois.

Une fois que le message est reçu, j'envoie la chaîne `ppp`.

Remarque : n'oubliez pas d'ajouter un \ à la fin de la ligne précédente pour que `chat` pense que le script de connexion de `chat` est toujours sur une seule ligne !

Malheureusement, certains serveurs génèrent un nombre de messages très variable ! Vous aurez peut-être besoin de vous connecter plusieurs fois avec `minicom` pour comprendre comment ça marche et récupérer une chaîne "d'attente" stable.

15.5 Un script chat pour les connections authentifiées avec PAP/CHAP

Si votre FAI utilise PAP/CHAP, alors votre script `chat` est bien plus simple. Tout ce que votre script `chat` doit faire est de composer le numéro, attendre la connexion et ensuite laisser `pppd` s'occuper du processus de connexion !

```
#!/bin/sh
#
# C'est la deuxieme partie du script ppp-on. Il contient le protocole de
# connexion de la connexion desiree.
#
exec /usr/sbin/chat -v                                    \
```

```
TIMEOUT          3                                  \
ABORT            '\nBUSY\r'                         \
ABORT            '\nNO ANSWER\r'                    \
ABORT            '\nRINGING\r\n\r\nRINGING\r'       \
''               \rAT                               \
'OK-+++\c-OK'    ATH0                               \
TIMEOUT          30                                 \
OK               ATDT$TELEPHONE                     \
CONNECT          ''                                 \
```

15.6 Les options `debug` et `file option_file` de pppd

Comme nous l'avons déjà vu, vous pouvez activer les informations de débuggage avec l'option -d de pppd. L'option 'debug' est équivalente à celle-ci.

Puisque vous établissez la connexion avec un nouveau script, laissez pour le moment les informations de débuggage. (Attention : si votre espace disque est réduit, les échanges de connexion de `pppd` peuvent rapidement augmenter votre fichier syslog et vous amener des problèmes - mais pour aller jusque là, il faut que vous essayez et vous échouiez vos connexions un certain temps).

Une fois que tout marche correctement, vous pouvez alors enlever cette option.

Si vous avez appelé votre fichier d'options de ppp autrement que `/etc/ppp/options` ou `/etc/ppp/options.ttySx`, spécifiez le nom du fichier avec l'option `file` de `pppd` :

```
exec /usr/sbin/pppd debug file options.myserver /dev/ttyS0 38400 \
```

16 Testez votre script de connexion

Ouvrez un nouveau Xterm **root** (si vous êtes sous X) ou ouvrez une nouvelle console virtuelle et connectez vous comme **root**.

Dans cette nouvelle session, tapez la commande

```
tail -f /var/log/messages
```

(ou l'endroit exact ou se trouve le fichier log).

Dans la première fenêtre (ou console virtuelle), tapez la commande

```
ppp-on &
```

(ou le nom que vous avez donné à la nouvelle version de votre /usr/sbin/ppp- on). Si vous ne mettez pas le script en tâche de fond en spécifiant & en fin de commande, vous ne récupérerez pas le prompt de votre terminal avant que ppp se termine (ou que la liaison se termine).

Maintenant revenez dans la fenêtre ou vous affichez votre journal système.

L'affichage devrait ressembler à ce qui suit (si vous avez spécifié l'option -vi a `chat` et -d à `pppd`)....c'est le déroulement du script `chat` et des réponses envoyées dans le fichier journal système suivi des informations de démarrage de `pppd` :

```
Oct 21 16:09:58 hwin chat[19868]: abort on (NO CARRIER)
Oct 21 16:09:59 hwin chat[19868]: abort on (BUSY)
Oct 21 16:09:59 hwin chat[19868]: send (ATZ^M)
Oct 21 16:09:59 hwin chat[19868]: expect (OK)
Oct 21 16:10:00 hwin chat[19868]: ATZ^M^M
Oct 21 16:10:00 hwin chat[19868]: OK -- got it
Oct 21 16:10:00 hwin chat[19868]: send (ATDT722298^M)
Oct 21 16:10:00 hwin chat[19868]: expect (CONNECT)
Oct 21 16:10:00 hwin chat[19868]: ^M
Oct 21 16:10:22 hwin chat[19868]: ATDT722298^M^M
Oct 21 16:10:22 hwin chat[19868]: CONNECT -- got it
Oct 21 16:10:22 hwin chat[19868]: send (^M)
Oct 21 16:10:22 hwin chat[19868]: expect (ogin:)
Oct 21 16:10:23 hwin chat[19868]: kepler login: -- got it
Oct 21 16:10:23 hwin chat[19868]: send (hartr^M)
Oct 21 16:10:23 hwin chat[19868]: expect (ssword:)
Oct 21 16:10:23 hwin chat[19868]:  hartr^M
Oct 21 16:10:23 hwin chat[19868]: Password: -- got it
Oct 21 16:10:23 hwin chat[19868]: send (??????^M)
Oct 21 16:10:23 hwin chat[19868]: expect (hartr)
Oct 21 16:10:24 hwin chat[19868]: [hartr -- got it
Oct 21 16:10:24 hwin chat[19868]: send (ppp^M)
Oct 21 16:10:27 hwin pppd[19872]: pppd 2.1.2 started by root, uid 0
Oct 21 16:10:27 hwin pppd[19873]: Using interface ppp0
Oct 21 16:10:27 hwin pppd[19873]: Connect: ppp0 <--> /dev/cua1
Oct 21 16:10:27 hwin pppd[19873]: fsm_sdata(LCP): Sent code 1, id 1.
Oct 21 16:10:27 hwin pppd[19873]: LCP: sending Configure-Request, id 1
Oct 21 16:10:27 hwin pppd[19873]: fsm_rconfreq(LCP): Rcvd id 1.
Oct 21 16:10:27 hwin pppd[19873]: lcp_reqci: rcvd MRU
Oct 21 16:10:27 hwin pppd[19873]: (1500)
Oct 21 16:10:27 hwin pppd[19873]:  (ACK)
Oct 21 16:10:27 hwin pppd[19873]: lcp_reqci: rcvd ASYNCMAP
Oct 21 16:10:27 hwin pppd[19873]: (0)
Oct 21 16:10:27 hwin pppd[19873]:  (ACK)
Oct 21 16:10:27 hwin pppd[19873]: lcp_reqci: rcvd MAGICNUMBER
Oct 21 16:10:27 hwin pppd[19873]: (a098b898)
Oct 21 16:10:27 hwin pppd[19873]:  (ACK)
Oct 21 16:10:27 hwin pppd[19873]: lcp_reqci: rcvd PCOMPRESSION
Oct 21 16:10:27 hwin pppd[19873]:  (ACK)
Oct 21 16:10:27 hwin pppd[19873]: lcp_reqci: rcvd ACCOMPRESSION
Oct 21 16:10:27 hwin pppd[19873]:  (ACK)
Oct 21 16:10:27 hwin pppd[19873]: lcp_reqci: returning CONFACK.
Oct 21 16:10:27 hwin pppd[19873]: fsm_sdata(LCP): Sent code 2, id 1.
Oct 21 16:10:27 hwin pppd[19873]: fsm_rconfack(LCP): Rcvd id 1.
Oct 21 16:10:27 hwin pppd[19873]: fsm_sdata(IPCP): Sent code 1, id 1.
Oct 21 16:10:27 hwin pppd[19873]: IPCP: sending Configure-Request, id 1
Oct 21 16:10:27 hwin pppd[19873]: fsm_rconfreq(IPCP): Rcvd id 1.
Oct 21 16:10:27 hwin pppd[19873]: ipcp: received ADDR
Oct 21 16:10:27 hwin pppd[19873]: (10.144.153.51)
Oct 21 16:10:27 hwin pppd[19873]:  (ACK)
```

```
Oct 21 16:10:27 hwin pppd[19873]: ipcp: received COMPRESSTYPE
Oct 21 16:10:27 hwin pppd[19873]: (45)
Oct 21 16:10:27 hwin pppd[19873]:  (ACK)
Oct 21 16:10:27 hwin pppd[19873]: ipcp: returning Configure-ACK
Oct 21 16:10:28 hwin pppd[19873]: fsm_sdata(IPCP): Sent code 2, id 1.
Oct 21 16:10:30 hwin pppd[19873]: fsm_sdata(IPCP): Sent code 1, id 1.
Oct 21 16:10:30 hwin pppd[19873]: IPCP: sending Configure-Request, id 1
Oct 21 16:10:30 hwin pppd[19873]: fsm_rconfreq(IPCP): Rcvd id 255.
Oct 21 16:10:31 hwin pppd[19873]: ipcp: received ADDR
Oct 21 16:10:31 hwin pppd[19873]: (10.144.153.51)
Oct 21 16:10:31 hwin pppd[19873]:  (ACK)
Oct 21 16:10:31 hwin pppd[19873]: ipcp: received COMPRESSTYPE
Oct 21 16:10:31 hwin pppd[19873]: (45)
Oct 21 16:10:31 hwin pppd[19873]:  (ACK)
Oct 21 16:10:31 hwin pppd[19873]: ipcp: returning Configure-ACK
Oct 21 16:10:31 hwin pppd[19873]: fsm_sdata(IPCP): Sent code 2, id 255.
Oct 21 16:10:31 hwin pppd[19873]: fsm_rconfack(IPCP): Rcvd id 1.
Oct 21 16:10:31 hwin pppd[19873]: ipcp: up
Oct 21 16:10:31 hwin pppd[19873]: local  IP address 10.144.153.104
Oct 21 16:10:31 hwin pppd[19873]: remote IP address 10.144.153.51
```

(Remarque - j'utilise les adresses IP STATIQUES - ma machine l'envoie au serveur PPP - vous ne verrez pas la même chose si vous utilisez des adresses IP DYNAMIQUES.) Ainsi, le server nécessite une commande spécifique pour démarrer ppp à la fin.

Ca semble OK - alors testez comme avant avec des ping des adresses IP et des noms des hôtes.

Lancer votre navigateur web ou n'importe quoi d'autre et allez surfer - vous êtes connecté !

17 Terminer une connexion PPP

Quand vous en avez terminé avec la liaison PPP, utilisez la commande standard ppp-off pour la terminer (rappel - vous devez être **root** ou membre du groupe PPP !).

Dans votre fichier journal de votre système, il devrait y avoir quelque chose ressemblant à :

```
Oct 21 16:10:45 hwin pppd[19873]: Interrupt received: terminating link
Oct 21 16:10:45 hwin pppd[19873]: ipcp: down
Oct 21 16:10:45 hwin pppd[19873]: default route ioctl(SIOCDELRT): Bad address
Oct 21 16:10:45 hwin pppd[19873]: fsm_sdata(LCP): Sent code 5, id 2.
Oct 21 16:10:46 hwin pppd[19873]: fsm_rtermack(LCP).
Oct 21 16:10:46 hwin pppd[19873]: Connection terminated.
Oct 21 16:10:46 hwin pppd[19873]: Exit.
```

Ne pas s'inquiéter du `SIOCDELRT` - c'est juste `pppd` qui indique qu'il se termine, il n'y a pas à s'en inquiéter.

18 Résoudre les problèmes

Il y a de nombreuses raisons pour que votre liaison ne fonctionne pas - `chat` n'a pas réussi à aller jusqu'au bout, vous avez une mauvaise ligne, etc. Regardez votre syslog pour avoir des indications.

18.1 J'ai compilé le support PPP dans le noyau, mais...

Un problème relativement courant est que les gens compilent le support PPP dans leur noyau et après, lorsqu'ils essayent de lancer pppd, le noyau répond qu'il ne supporte pas ppp ! Un grand nombre de raisons peuvent en être la cause.

18.1.1 Bootez-vous avec le bon noyau ?

Même si vous **avez** recomplié votre support de ppp dans le noyau, vous ne bootez pas sur le nouveau noyau. Cela peut arriver si vous n'avez pas mis à jour `/etc/lilo.conf` et relancé lilo.

Une bonne façon de vérifier le noyau est d'envoyer la commande `uname -a`, qui affiche la ligne ressemblant à

```
Linux archenland 2.0.28 #2 Thu Feb 13 12:31:37 EST 1997 i586
```

Cela donne la version du noyau et la date à laquelle il a été compilé - ce qui vous donnera une bonne idée de ce qui se passe.

18.1.2 Avez-vous compilé le support noyau de ppp comme un module ?

Si vous avez compilé le support de ppp comme un module, mais n'avez pas fait d'installation des modules, alors vous pouvez avoir cette erreur. Relisez le kernel-HOWTO et le fichier README de `/usr/src/linux` !

Une autre possibilité concernant les modules est que vous supposez que les modules sont chargés automatiquement, alors que le daemon `kerneld` n'est pas lancé (il charge et décharge les modules au vol). Vérifiez le mini-HOWTO kerneld pour des informations sur la configuration de kerneld.

18.1.3 Utilisez-vous une bonne version de PPP pour votre noyau ?

Vous **devez** utiliser ppp-2.2 avec les versions 2.0.x du noyau. Vous pouvez utiliser ppp-2.2 avec les versions 1.2.x du noyau (si vous le patchez) sinon vous devez utiliser ppp-2.1.2.

18.1.4 Lancez vous pppd en étant root ?

Si vous ne lancez pas `pppd` en étant l'utilisateur **root** (et que `pppd` n'est pas suid vers **root**), vous pourrez recevoir ce message.

18.2 Mon modem se connecte mais PPP ne démarre jamais

Il y a d'innombrables possibilités avec ça (regardez dans comp.os.linux....).

Une erreur **EXTREMEMENT** fréquente est que vous avez mal tapé quelque chose dans vos scripts. La seule chose à faire est de vérifier que vous avez la conversation entre votre PC Linux et le serveur rapporté dans le syslog /var/log/message) allez ensuite dans celui-ci *ligne par ligne*. Vous aurez peut-être besoin d'appeler votre serveur ppp à la main pour revérifier tout ça.

Vous devez vérifier les messages de logs réels très attentivement - et ayez à l'esprit que les hommes on tendance à lire ce qu'ils PENSENT avoir tapé - et non ce qu'ils ont sous leurs yeux !

18.3 Le syslog contient `"serial line is not 8 bit clean..."`

Il y a plusieurs possibilités à cela - comme des retours sur la ligne etc., qui peuvent être la conséquence d'une (ou plusieurs) choses.

Pour comprendre ce qui se passe, il est nécessaire de chercher se qui se passe en coulisses dans `pppd` lui-même.

Lorsque `pppd` démarre, il envoie des paquets LCP (Link Control Protocol) à la machine distante. Si il reçoit une réponse valide il passe ensuite à l'étape suivante (avec IPCP - les paquets de contrôle IP) et c'est seulement une fois que cette négociation est terminée que les trames IP réelles commencent et que vous pouvez utiliser la liaison PPP.

Si il n'y a pas de serveur ppp fonctionnel à l'autre bout quand votre PC envoie les paquets LCP, ceux-ci sont renvoyés par le processus de login final. Comme ces paquets utilisent 8 bits, et sont réfléchis avec le 8ème bit (souvenez vous qu'ASCII est un code 7 bits). PPP s'en aperçoit et s'en plaint.

Il y a plusieurs possibilité pour qu'un réflexion arrive.

18.3.1 Vous n'êtes pas correctement connecté sur le serveur

Lorsque le script `chat` se termine, `pppd` démarre sur votre PC. Cependant, si vous n'avez pas terminé le processus de connexion au serveur (comme envoyer une commande nécessaire à lancer PPP sur le serveur), PPP ne se lancera pas.

Ainsi, le paquet LCP sera réfléchi et vous aurez une erreur.

Vous devez vérifier et corriger (si nécessaire) attentivement votre script `chat` (voir plus haut).

18.3.2 Vous n'avez pas lancé PPP sur le serveur

Certains serveurs PPP nécessitent que vous entriez une commande et/ou un retour chariot après avoir terminé le processus de connexion et avant que ppp soit lancé à l'autre bout.

Vérifiez votre script `chat` (voir plus haut).

Si vous vous connectez à la main et que vous vous rendez compte que vous devez envoyer sur ENTREE après avoir lancé PPP, ajouter simplement une paire attente/envoi vide à la fin de votre script `chat` (un chaîne vide envoie en fait ENTREE).

18.4 Le processus PPP distant est long à démarrer

Là c'est un peu délicat !

Par défaut, votre `pppd` Linux est compilé pour envoyer un maximum de 10 requêtes LCP de configuration. Si le serveur est un peu lent à démarrer, la totalité des 10 requêtes auront été envoyées avant que le serveur PPP distant n'ait eu le temps de les recevoir.

Sur votre machine, `pppd` voit la réflexion des 10 requêtes (avec le 8éme bit inversé) et s'arrête.

Il y a deux façon de résoudre cela :

Ajouter `lcp-max-configure 30` comme option de ppp. Cela augmentera le nombre de packets de configuration LCP que `pppd` enverra avant d'abandonner. Pour les serveurs vraiment lents, vous devez même mettre plus que ça.

Sinon, vous pouvez être un peu malin de votre côté. Vous avez peut-être remarqué que lorsque vous vous connectez à la main au serveur PPP et que vous démarrez alors PPP, le **premier** caractère que ppp envoie et qui apparaît est toujours le caractère tilda (~).

En savant cela, vous pouvez ajouter une nouvelle paire `attente/envoi` à la fin de votre script `chat` qui attendra un tilda et n'enverra rien. Cela ressemblera alors à :

```
\~          ''
```

Remarque : comme le caractère tilda a une signification particulière pour le shell, il doit être échappé (en fait précédé par un backslash).

18.5 La route par défaut n'est pas configurée

Si `pppd` refuse de configurer la route par défaut, c'est parce que (assez justement) il refuse de remplacer/supprimer une route par défaut existante.

La raison habituelle lorsque cette erreur apparaît est que certaines distributions configurent une route par défaut pour votre carte Ethernet au lieu de la configurer comme une route réseau spécifique.

Voir le Linux NAG et le HOWTO Net2/3 pour des informations sur la façon de configurer correctement votre carte Ethernet et les routes associées.

Une alternative à cela est que votre réseau local utilise déjà un gateway/routeur et que votre table de routage est déjà configurée pour envoyer la route par défaut à cet endroit.

Résoudre cette situation peut nécessiter quelques connaissances de réseau IP qui sort du domaine de ce HOWTO. Il est suggéré d'obtenir l'avis d'un expert (par les groupes de news ou quelqu'un que vous pouvez interroger près de vous).

18.6 Autres problèmes

Il y a des tas de raisons autres que celles-ci pour que ppp ne parvienne pas à se connecter et/ou fonctionner correctement.

Lisez la FAQ PPP(qui est une série de questions-réponses). C'est un document très didactique et les réponses y SONT ! De ma propre (et triste) expérience, si la réponse à votre problème n'est pas dedans, votre problème N'est PAS de la faute de ppp ! Pour ma part, j'utilisais un noyau ELF alors que je n'avais pas mis à jour les modules du noyau. J'ai juste perdu 2 jours (et presque une nuit) à chercher comment doit être un serveur PPP parfait avant de faire lumière !

19 Obtenir de l'aide quand on est totalement perdu

Si vous n'arrivez pas à faire fonctionner votre liaison PPP, relisez ce document et tout vérifier - en conjonction avec les sorties de "chat-v..." et "pppd -d" dans votre journal système.

Consultez également la documentation de PPP la FAQ et tout les documents mentionnés ici !

Si vous êtes encore bloqué, essayez les groupes de news comp.os.linux.misc et comp.os.linux.networking qui sont régulièrement parcourus par des gens capable de vous aider sur PPP ainsi que dans comp.protocols.ppp

Vous pouvez essayer de m'envoyer personnellement un email, mais si j'ai une journée (et même une vie) chargé, je ne garanti pas de répondre rapidement (même pas du tout) selon la quantité de travail en cours et l'état de ma vie privée !

En particulier - **NE POSTEZ PAS DES PAGES ET DES PAGES DE SORTIE DE DEBUGGAGE DANS LES NEWS GROUPS NI MEME A MOI PAR EMAIL** - c'est une énorme perte de bande passante, et la majorité sera envoyé dans /dev/null (à moins qu'elle ne soit expressément demandé).

20 Problèmes classiques lorsque la liaison fonctionne

Un des problèmes que vous rencontrerez est que de nombreux fournisseurs d'accès est qu'ils ne supporterons que les programmes de connexions qu'ils distribuent avec les nouveaux comptes. Il sont (habituellement) pour Microsoft Windows :-(- et la plupart des bureau d'aide des fournisseurs d'accès ne sembleront rien savoir au sujet d'Unix (ou Linux). Donc préparez vous à une assistance limité de leur part !

Vous pouvez évidemment leur faire une faveur et leur apprendre ce qu'est Linux (certaines personnes au support de votre fournisseur d'accès seront relativement 'branché' avec le monde Internet est auront une machine Linux chez eux - bien sûr que ça existe !)

20.1 Je n'arrive pas à voir après le serveur PPP ou je suis connecté

OK - votre connexion PPP est lancée et fonctionne vous pouvez pinger le serveur PPP avec son adresse IP (la deuxième ou "remote" adresse IP affichée par `ifconfig ppp0`), mais vous ne pouvez rien atteindre après cela.

Tout d'abord, essayez de pinger les adresses IP que vous avez spécifié dans /etc/resolv.conf comme serveur de noms. Si elles marchent, vous **pouvez** voir après votre serveur PPP (en tout cas ce qui à une adresse IP identique à ce que vous avez spécifié). Maintenant essayez de pinger l'adresse complète de votre fournisseur d'accès :

```
ping mon.fournisseur.internet.fr
```

Si ça ne marche pas, vous avez un problème avec la résolution des noms. C'est probablement à cause d'une faute de frappe dans votre fichier /etc/resolv.conf. Vérifiez ça attentivement à partir des informations que vous aviez obtenues en appelant votre fournisseur d'accès. Si ça semble correct, appelez votre fournisseur d'accès, et vérifiez que vous avez noté l'adresse IP correctement.

Si ça ne marche TOUJOURS pas (et que votre fournisseur d'accès vous confirme que le serveur de noms fonctionne correctement), vous devez avoir un problème ailleurs - et je vous suggère de vérifier votre installation de Linux (en particulier les permissions des fichiers).

Si vous ne pouvez TOUJOURS pas pinger le serveur de nom de votre fournisseur d'accès par l'adresse IP, soit il est éteint (appelez les par téléphone et vérifiez) soit il y a un problème de routage du côté de votre fournisseur d'accès. De même appelez-les et demandez leur.

Un autre possibilité est qu'à l'autre bout le serveur PPP Linux n'ait pas l'option IP forwarding d'activée dans le noyau !

Un bon test est d'essayer de se connecter avec le programme que votre fournisseur d'accès vous a donné pour (oups) Microsoft Windows. Si tout marche sur un autre système d'exploitation avec exactement le même compte, alors le problème vient de votre machine Linux et NON de votre fournisseur d'accès.

20.2 Je peux envoyer du courrier mais pas en recevoir

Si vous avez une adresse IP dynamique, c'est tout à fait normal. Voir "Configurer les services" plus bas.

20.3 Pourquoi les gens ne peuvent pas faire de finger, WWW, gopher, talk etc avec ma machine ?

La aussi, si vous utilisez une adresse IP dynamique c'est tout à fait normal. voir "Configurer les services" plus bas.

21 Utiliser les services Internet avec des adresses IP dynamiques

Si vous utilisez une adresse IP dynamique (et tous les provideurs ne vous donnerons qu'une adresse dynamique à moins que vous ne payez bien plus pour votre connexion), vous devez accepter les limitations qu'elle impose.

Tout d'abord, les requêtes pour les services sortant fonctionneront parfaitement. Cela signifie que vous pourrez envoyer du courrier avec sendmail (si sendmail est configuré correctement), les fichiers ftp pour les sites distants, les fingers vers d'autres machines, surfer sur le web etc.

En particulier, vous pourrez répondre au courrier que vous avez reçu même si votre machine n'est plus connectée. Le courrier restera dans votre queue de messages jusqu'à ce que vous appeliez votre FAI.

Cependant, votre machine N'est PAS connectée à Internet 24H/24, et en plus elle n'a pas la même adresse IP à chaque fois qu'elle se connecte. Il est donc impossible de recevoir du courrier électronique directement sur votre machine, et de créer un serveur web et ftp que vos amis pourraient accéder ! A partir du moment ou ça concerne Internet, votre machine n'est pas unique, ni accessible de manière permanente comme cela serait le cas avec une adresse IP unique (rappel - les autres machines utilisent les adresses IP pour accéder à votre machine).

Si vous configurez en serveur WWW (ou n'importe quoi d'autre), il est totalement inconnu des utilisateurs d'Internet A MOINS QU'ils soient au courant que votre machine est connectée et qu'ils connaissent votre adresse IP courante. Il y a plusieurs façon de fournir cette information, vous demander par téléphone, leur envoyer un e-mail pour leur dire, ou utiliser un fichier ".plan" sur le compte shell de votre fournisseur d'accès (en partant du principe que votre fournisseur d'accès permet les script shell et les finger).

Maintenant, pour la majorité des utilisateurs, ça n'est pas vraiment un problème - la plupart des gens veulent pouvoir envoyer et recevoir leur email (sur le compte de leur fournisseur d'accès) et faire des connexions sortantes vers WWW, ftp et d'autres services d'Internet. Si vous AVEZ un besoin impératif de connexion sortantes, vous feriez mieux de prendre une adresse IP statique. Cependant, vous pouvez aussi explorer les méthodes astucieuses ci dessous...

21.1 Configurer le courrier électronique (email)

Même avec une adresse IP dynamique, vous devrez certainement configurer sendmail sur votre machine pour envoyer le courrier que vous écrivez localement. La configuration de sendmail peut être obscure et difficile - ce document ne vous parleras pas de la façon de faire. Cependant, vous aimeriez sans doute configurer sendmail pour que votre fournisseur d'accès soit considéré comme un hôte "smart relay" (l'options **DS** de sendmail). (Pour plus d'info sur la configuration de sendmail, voir la documentation de sendmail - et regardez la configuration de m4 fourni avec sendmail. Il y a forcement quelque chose qui réponds à ce que vous cherchez).

Il y a un excellent livre sur Sendmail (une véritable 'bible' aux éditions O'Reilly and Associates), mais c'est certainement beaucoup trop pour la plupart des utilisateurs !

Une fois que vous avez configuré sendmail, vous avez probablement envie que sendmail envoie les messages en attente dès que la connexion PPP est lancée. Pour cela, ajouter la commande

```
sendmail -q &
```

dans votre script /etc/ppp/ip-up (voir plus haut).

Les mails entrant sont problématique avec les adresses IP dynamiques. La façon de les supporter est de :

- configurer votre lecteur d'email pour que le courrier contiennent un champ "reply to" dans lequel il y aura votre adresse électronique chez votre fournisseur d'accès.

Si vous le pouvez, configurez également le champ FROM avec votre adresse électronique chez votre FAI.

- utiliser les programmes popclient, fetchmail pour récupérer le courrier chez votre fournisseur d'accès. Ou bien, si votre FAI utilise IMAP, utilisez un lecteur de courrier électronique ayant IMAP d'activé (comme pine).

Vous pouvez automatiser ce processus lors de l'appel en ajoutant la commande nécessaire dans le script `/etc/ppp/ip-up` (voir plus haut).

21.2 Configurer un serveur de noms local

Même si vous êtes content d'utiliser le serveur de noms situé chez votre fournisseur d'accès, vous pouvez en plus configurer un serveur de nom local (secondaire) de cache uniquement qui serait lancé avec le script ip-up. L'avantage d'utiliser un serveur de noms local (cache uniquement) est de gagner du temps (et de la bande passante) si vous contactez souvent les même sites pendant de longues sessions en ligne.

La configuration du DNS pour un serveur de cache seulement (qui utilise une ligne 'forwarders' dans le fichier named.boot pointant sur le DNS de votre FAI) est relativement simple. Le livre O'Reilly (DNS et Bind) explique tout ce que vous devez savoir sur le sujet.

Il existe également un DNS-HOWTO.

Si un petit réseau local accède à Internet grâce à votre PC Linux (avec IP masquerading par exemple), c'est sans doute une excellente idée d'utiliser un serveur de noms (avec la directive forwarders) puisque cela réduira les delais et la bande passante utilisée pour la résolution des noms.

Un point de la Netiquette : demandez la permission à votre fournisseur d'accès avant de mettre en place un serveur de noms secondaire de cache uniquement dans le domaine de votre FAI. Configuré correctement, il n'entrainera aucun problème chez votre FAI, mais si il ne l'est pas, il peut causer de nombreux problèmes.

22 Relier deux réseaux locaux avec PPP

Il y a fondamentalement aucune différence entre le fait de relier un seul PC Linux avec PPP et relier deux réseaux locaux avec PPP et une machine sur chaque réseau. Souvenez-vous PPP est un protocole **réversible**.

Cependant, vous devez **REELLEMENT** comprendre comment le routage se déroule. Lisez le NET-2 HOWTO et le Guide d'administration Linux (NAG). Vous trouverez également une aide inestimable dans le " TCP/IP Network Administration" (publié par O'Reilly and Assoc - ISBN 0-937175-82-X)

Si vous utilisez une adresse de sous-réseau de chaque côté de la liaison, le (brouillon) de mini-howto Linux sub networking pourra vous être utile. Il est disponible à *Linux Sub networking mini-HOWTO* `http://www.interweft.com.au/other/` .

Pour relier deux réseaux locaux, vous **devez** utiliser des adresses IP différentes (ou des sous-réseau d'adresses différentes) et vous devrez utiliser des adresses IP statiques ou l'IP Masquerading. Si vous voulez utiliser l'IP Masquerading, consultez le mini-howto IP masquerade pour avoir des instructions sur la manière de le configurer.

22.1 Configurer les adresses IP

Arrangez-vous avec l'administrateur système de l'autre réseau local pour choisir les adresses IP qui seront utilisés de chaque côté de l'interface PPP. Si vous utilisez des adresses IP statiques, il sera sans doute pratique d'appeler un numéro de téléphone dédié.

Maintenant, éditez le fichier /etc/ppp/options[.ttyXX] approprié - il est judicieux d'avoir un modem et un port dédié de chaque côté de la connexion. Cela nécessite quelques changements dans le fichier /etc/ppp/options et la création d'un fichier options.ttyXX approprié pour les autres connexions !

Spécifiez les adresses IP pour le côté de votre liaison PPP dans les fichiers d'options appropriés exactement comme cela est décris plus haut pour les adresses IP statiques.

22.2 Configurer le routage.

Vous devez vous occuper que les paquets du réseau local soit routés à travers l'interface lorsque la liaison PPP est lancée. Ce processus se fait en deux parties.

Tout d'abord, vous devez configurer une route partant de la machine qui fait marcher la liaison PPP vers le réseau situé à l'autre bout. Si cette liaison donne sur Internet, vous pouvez utilisez la route par défaut mise en place par pppd lui-même à la connexion avec l'option defaultroute.

Si toutefois, la liaison relie seulement deux réseaux locaux, alors une route réseau spécifique doit être ajoutée pour tous les réseaux accessible par cette liaison. C'est la commande 'route' qui le fait pour chaque réseau dans le script /etc/ppp/ip-up script (voir une fois que la liaison fonctionne...) pour des instructions détaillées.

La seconde chose dont vous avez besoin est de prévenir les autres ordinateurs que vous êtes le gateway vers les réseaux situés à l'autre bout de la liaison.

Bien sûr, l'administrateur réseau de l'autre coté de la liaison doit faire les même choses ! Cependant, puisque qu'il route les paquets vers votre réseau particulier, une **route réseau spécifique** est indispensable, et non la route par défaut (à moins que le réseau situé à l'autre bout se connecte chez vous pour accéder à Internet).

22.3 Sécurité réseau

Si vous reliez votre réseau local à Internet avec PPP - ou du moins avec un réseau local "étranger", vous devez réfléchir aux problèmes de sécurité. Je vous conseille fermement de penser à configurer un firewall !

Vous devriez également parler à l'administrateur réseau de votre site **AVANT** de lancer une connexion vers un réseau local étranger voire vers Internet ce cette manière. Ne rien faire peut soit n'amener aucune réaction soit de sérieux problèmes !

23 Une fois que la connexion fonctionne - le script /etc/ppp/ip-up

Une fois que la connexion est établie, pppd cherche un /etc/ppp/ip-up. Si ce script existe et est exécutable, le daemon PPP le lance. Cela permet d'automatiser des commandes de routages spéciales qui peuvent être nécessaires et toutes les autres actions que vous souhaitez lancer une fois que la liaison PPP est activée.

C'est un simple script shell qui fait uniquement ce qu'un script shell peut faire (en fait quasiment tout ce que vous voulez).

Par exemple, vous pouvez demander à sendmail d'envoyer les messages en attentes.

Il y a quelques restrictions avec /etc/ppp/ip-up:

- Il fonctionne délibéremment dans un environnement réduit pour augmenter la sécurité. Cela signifie que vous devez donner le chemin complet des exécutables.
- Techniquement, /etc/ppp/ip-up est un *programme* et non un script. Cela signifie qu'il ne peut pas être exécuté directement - même si il a besoin du (#!/bin/bash) magique au début de la première ligne et qu'il doit être lisible est exécutable par root.

23.1 Routages spéciaux

Si vous reliez deux réseaux locaux, vous devez configurer des routes spécifiques vers le réseau 'étranger'. Le script `/etc/ppp/ip-up` peut le faire facilement. La seule difficulté est lorsque votre machine supporte plusieurs liaisons PPP.

`/etc/ppp/ip-up` est utilisé à chaque lancement de ppp, vous devez donc être extrêmement vigilant d'exécuter les commandes de routage correctes pour la liaison qui démarre - et non lorsque n'importe quelle liaison démarre !

23.2 Support du courrier électronique en attente

Lorsqu'une liaison entre deux réseaux locaux démarre, vous voudriez vous assurez que le courrier en attente de chaque coté de la liaison est bien envoyé à son destinataire. Cela est possible en ajoutant un appel correct à `sendmail`.

Utiliser la commande case du bash sur le bon paramètre que passe pppd au script permet cela. Par exemple, voici un script `/etc/ppp/ip-up` que j'utilise pour mes liaisons entre l'entreprise et chez mon réseau Ethernet (également connecté au même serveur ppp).

23.3 Un exemple de script /etc/ppp/ip-up

Le script suivant permet de nombreuses utilisations.

```
#!/bin/bash
#
# Script permettant de supporter les possiblites de routages necessaires a pppd
# Seule la liaison vers Newman necessite ce support.
#
# Quand la liaison ppp demarre, ce script est appele avec les parmetres
# suivants
#       $1      le nom de l'interface utilisee par pppd (c.-a-d. ppp3)
#       $2      le nom du peripherique tty
#       $3      la vitesse du peripherique tty
#       $4      l'adresse IP locale de l'interface
#       $5      l'adresse  IP distante
#       $6      le parametre specifie par l'option 'ipparam' a pppd
#
case "$5" in
# Supporte le routage vers le serveur du Newman Campus
        202.12.126.1)
                /sbin/route add -net 202.12.126.0 gw 202.12.126.1
# et envoie les courriers en attente la-bas !
                /usr/sbin/sendmail -q &
                ;;
        139.130.177.2)
# Notre liaison Internet
# Lorsque notre liaison est lancee, demarre le serveur de temps et
# se synchronise avec le monde si il n'a pas encore bouge.
                if [ ! -f /var/lock/subsys/xntpd ]; then
                        /etc/rc.d/init.d/xntpd.init start &
```

```
                fi
# Demarre le serveur de news (si il ne marche pas encore)
                if [ ! -f /var/lock/subsys/news ]; then
                        /etc/rc.d/init.d/news start &
                fi
                ;;
        203.18.8.104)
# Recupere le courrier electronique sur ma machine tant que la liaison
# fonctionne. Aucun routage n'est necessaire tant que mon reseau local
# Ethernet supporte IP masquerade et le routages proxyarp.
                /usr/sbin/sendmail -q &
                ;;
        *)
esac
exit 0
```

Le résultat du lancement de la liaison PPP vers notre campus Newman et le démarrage de ce script, détermine les entrées de la table de routage (cette machine est aussi notre serveur PPP général ET supporte nos liaisons Internet). J'ai ajouté des commentaires sur la sortie pour expliquer a quoi correspond chaque entrée) :

```
[root@kepler /root]# route -n
Kernel routing table
Destination     Gateway         Genmask         Flags MSS    Window Use Iface
# la route de l'HOTE de notre gateway internet
139.130.177.2   *               255.255.255.255 UH    1500   0      134 ppp4
# la route de l'HOTE vers le serveur Newman campus
202.12.126.1    *               255.255.255.255 UH    1500   0       82 ppp5
# la route de l'HOTE de mon réseau local ethernet
203.18.8.104    *               255.255.255.255 UH    1500   0       74 ppp3
# deux des lignes générales PPP
203.18.8.64     *               255.255.255.255 UH    552    0        0 ppp2
203.18.8.62     *               255.255.255.255 UH    552    0        1 ppp1
# la route réseau specifique vers le réseau Newman campus
202.12.126.0    202.12.126.1    255.255.255.0   UG    1500   0        0 ppp5
# la route de notre réseau local Ethernet (super-netting two adjacent C classes)
203.18.8.0      *               255.255.254.0   U     1500   0     1683 eth0
# la route vers le périphérique loop back
127.0.0.0       *               255.0.0.0       U     3584   0      483 lo
# la route par défaut vers Internet
default         139.130.177.2   *               UG    1500   0     3633 ppp4
```

23.4 Support du courrier électronique

La section précédente montrait comment supporter le mail sortant - en envoyant simplement le courrier lorsque la liaison démarrait.

Si vous utilisez une liaison pour un réseau large, vous pouvez vous arranger avec l'administrateur réseau du site distant pour qu'il fasse exactement la même chose. Par exemple, à l'autre bout de la liaison au Newman Campus, le script `/etc/ppp/ip-up` ressemble à :

```
#!/bin/bash
#
# Script permettant de supporter les possiblites de routages necessaires a pppd
# Seule la liaison vers Hedland necessite ce support.
#
# Quand la liaison ppp demarre, ce script est appele avec les parmetres
# suivants
#       $1      le nom de l'interface utilisee par pppd (c.-a-d. ppp3)
#       $2      le nom du peripherique tty
#       $3      la vitesse du peripherique tty
#       $4      l'adresse IP locale de l'interface
#       $5      l'adresse  IP distante
#       $6      le parametre specifie par l'option 'ipparam' a pppd
#
case "$5" in
        203.18.8.4)
                /usr/sbin/sendmail -q
                ;;
        *)
esac
exit 0
```

Si toutefois, vous n'avez qu'une adresse IP dynamique chez votre FAI, vous devrez recevoir votre courrier sur le compte de votre fournisseur d'accès. Cela est généralement possible en utilisant POP (Post Office Protocol). Ce processus peut être supporté avec le programme 'popclient" - le script ip-up peut automatiser le processus pour vous !

Créez simplement un script `/etc/ppp/ip-up` qui contient un appel correct à popclient. Pour mon portable avec la RedHat Linux (que je prends lorsque je voyage), j'ai

```
popclient -3 -c -u hartr -p <password> kepler.hedland.edu.au |formail -s procmail
```

Vous pouvez aussi utiliser slurp en quelque chose d'autre d'identique pour les news, et d'autres trucs. Souvenez vous que le script ip-up est un script bash standard et qui peut servir à automatiser TOUTES les fonctions que vous devez accomplir chaque fois qu'une liaison PPP correspondante démarre.

24 Utiliser /etc/ppp/ip-down

Vous pouvez écrire un script, qui sera exécuté à chaque fois qu'une liaison se termine. Qui se situe dans `/etc/ppp/ip-down`. Il peut être utilisé pour retirer ce que vous avez ajoutez de spécial avec le script /etc/ppp/ip-up correspondant.

25 Possibilité de routage sur un réseau local

Si vous êtes connecté sur un réseau local et que vous voulez toujours utiliser PPP avec votre machine Linux, vous devez définir les routes que les packets doivent prendre pour atteindre le réseau local à partir de votre machine (par l'interface Ethernet) et également le serveur PPP distant voire plus loin.

Cette section ne prétend pas vous apprendre le routage - elle traite seulement de cas simples et spéciaux de routage (statique) !

Je vous recommande chaudement de lire de Guide d'administration Réseau Linux si vous n'êtes pas habitué avec le routage. De même le livre d'O'Reilly "TCP/IP Network Administration" couvre le sujet de manière très abordable.

La règle de base du routage statique est que la route par DEFAUT pointe à l'endroit où il y le PLUS grand nombre d'adresse réseau. Pour les autres réseau, entrés des routes spécifiques dans la table de routage.

La SEULE situation que je couvre ici est lorsque votre machine Linux est sur un réseau non connecté à Internet - et vous voulez vous connectez à Internet pour une utilisation personnelle tout en restant connecté au réseau local.

Tout d'abord, assurez-vous que votre route Ethernet est bien configuré comme une adresse réseau spécifique disponible sur votre réseau local - et PAS dans la route par defaut !

Vérifiez cela en tapant la commande route, vous devriez avoir quelque chose qui ressemble à ce qui suit :

```
[root@hwin /root]# route -n
Kernel routing table
Destination     Gateway         Genmask         Flags MSS    Window Use Iface
loopback        *               255.255.255.0   U     1936   0        50 lo
10.0.0.0        *               255.255.255.0   U     1436   0       565 eth0
```

Si votre interface Ethernet (eth0) pointe vers la route par défaut, (la première colonne contiendra un "default" dans la ligne eth0) vous devez changer les scripts d'inititalisation d'Ethernet pour qu'ils utilisent des adresses réseau spécifiques plutôt que la route par défaut (consultez le HOWTO Net2 et le NAG).

Cela va permettre à pppd de configurer votre route par défaut comme suit :

```
[root@hwin /root]# route -n
Kernel routing table

Destination     Gateway         Genmask         Flags MSS    Window Use Iface
10.144.153.51   *               255.255.255.255 UH    488    0         0 ppp0
127.0.0.0       *               255.255.255.0   U     1936   0        50 lo
10.1.0.0        *               255.255.255.0   U     1436   0       569 eth0
default         10.144.153.51   *               UG    488    0         3 ppp0
```

Comme vous pouvez le voir, nous avons une route d'hôte vers le serveur PPP (10.144.153.51) par ppp0 qui est également la route par défaut utilisé par le serveur PPP comme gateway.

Si votre configuration est plus compliqué que cela, lisez les documents concernant le routage cités plus haut, et consultez un expert de votre site !

Si votre réseau local à aussi des routeurs, vous devez déjà avoir des gateways pour interconnecter les réseaux disponibles sur votre site. Vous devez TOUJOURS mettre la route par défaut vers l'interface PPP - et utiliser des routes spécifiques pour les autres réseaux que vous accédez.

25.1 Remarques sur la sécurité

Lorsque vous configurez une machine Linux sur un réseau local existant sur Internet, vous ouvrez potentiellement tout votre réseau local vers Internet - ainsi que les pirates qui l'utilise. Avant de commencer, je vous conseille vivement de consulter votre administrateur réseau et le responsable de la sécurité de votre site. Si votre connexion PPP vers Internet est utilisé et permet d'attaquer votre site, vous serez dans la plus mauvaise position face à la colère de vos collègues et votre administrateur système et réseau. Vous pourriez vous-même avoir de sérieux problèmes !

Avant de connecter un réseau local sur Internet, vous devez prendre en compte les problèmes de sécurité même avec une connexion DYNAMIQUE - dans ce cas la référence la plus récente est le "Building Internet Firewalls" d'O'Reilly !

26 Configurer un serveur PPP

Comme c'est mentionné plus haut, il y a des tas de façons de faire. Ce que je présente ici est la façon de faire une liaison tournante sur plusieurs lignes téléphoniques (avec une carte série multi-ports utilisant des Cyclades)

Si vous n'aimez pas la méthode présentée ici, faites donc comme vous préférez. J'aimerais toutefois ajouter des méthodes supplémentaires dans les prochaines versions de ce HOWTO. Envoyez donc moi comment vous faites !

Remarquez que cette section ne concerne que la configuration de Linux comme serveur PPP. Je ne compte absolument pas donner d'informations sur la façon de configurer des serveurs terminaux ou autre.

Même si, j'ai déjà expérimenté l'utilisation de shadow password (et l'utilise de temps en temps). Les informations présentées ici n'utiliserons toutefois pas ce mécanisme.

26.1 Compiler le noyau

Tous les commentaires précédent concernant la compilation du noyau et les versions du noyau et de `pppd` sont encore valables. Cette section considère que vous avez lu les sections précédentes du document !

Pour un serveur PPP, vous **DEVEZ** ajouter le support de IP forwarding dans votre noyau. Vous pouvez aussi inclure d'autres supports (comme le IP fire walls, accounting etc etc).

Si vous utilisez une carte série multi-port, vous devez bien entendu inclure également les pilotes nécessaires dans le noyau !

26.2 Présentation du système de serveur

Nous offrons des comptes PPP (ou SLIP) et des comptes shell en utilisant les mêmes paires nom utilisateur/ mot de passe. Cela permet (pour nous) que les utilisateurs n'aient qu'un seul compte qu'ils peuvent utiliser pour toutes les connexions.

Comme nous sommes une organisation éducative, nous ne faisons pas payer le personnel ni les étudiants pour leur accès, et nous n'avons donc pas besoin de nous inquiéter des problèmes de payement.

Un firewall existe entre notre site et Internet, et cela limite les accès des utilisateurs puisque les lignes d'appels sont sur notre firewall Internet (pour certaines raisons évidentes, le détails des autres firewalls internes ne sera pas présenté ici et serait de toute façon inutile)

Le processus avec lequel un utilisateur établit une liaison PPP avec notre site (une fois qu'ils ont un compte bien sûr) :

- Appeler notre ligne d'appel groupé (qui est un numéro de téléphone qui se connecte sur une banque de modems - le premier modem libre est utilisé).
- Connexion avec une paire valide de nom utilisateur et mot de passe
- Au prompt du shell, lancement de la commande `ppp` pour lancer PPP sur le serveur.

- Démarrer PPP sur le PC (il peut utiliser Windows, DOS, Linux, MAC OS ou n'importe quoi d'autre - ce n'est pas un problème).

Le serveur utilise un fichier `/etc/ppp/options.ttyXX` individuel pour chacune des connexions sur le port où est configuré l'adresse IP distante pour l'allocation des adresses IP dynamiques. Le serveur utilisateur route les clients avec proxyarp (configuré avec l'option appropriée de pppd). Cela oblige la présence de routed ou gated.

Quand l'utilisateur raccroche de son coté, `pppd` le détecte et indique au modem de raccrocher, fermant en même temps la liaison PPP.

26.3 Faire marcher les programmes tous ensemble

Vous avez besoin des programmes suivants :

- Linux, correctement compilé avec les options nécessaires
- La version de `pppd` correspondant à votre noyau
- Un programme 'getty' qui décroche intelligemment les communications par modem.
  Nous utilisons getty_ps2.0.7h, mais mgetty est aussi bien. Je sais que mgetty peut s'apercevoir qu'un appel utiliser PAP/CHAP (PAP est le standard pour Windows 95) et invoquer `pppd` automatiquement, mais je dois encore explorer un peu ça.
- Un serveur de nom (DNS) fonctionnel qui est accessible pour les utilisateurs qui se connectent.

Ce howto ne couvre pas sa configuration. Regardez la documentation de getty de votre choix ou le serial HOWTO pour plus d'informations sur le sujet.

26.4 Configurer les fichiers d'options

Vous avez besoin de configurer la totalité de `/etc/ppp/options` avec les options standard pour les ports d'appel. Les options utiles sont :

```
asyncmap 0
netmask 255.255.254.0
proxyarp
lock
crtscts
modem
```

Remarque - nous n'utilisons AUCUN routage (direct) - et en particulier il n'y a pas d'option defaultroute. La raison de cela est que tout ce dont vous avez besoin (en tant que serveur PPP) à l'endroit où router les paquets **venant** du client ppp vers votre réseau local ou Internet et de router les paquets **vers** le client ppp qui viennent du votre réseau local ou d'Internet.

Tout ce dont vous avez besoin est la route hôte de la machine client et l'utilisation de l'option 'proxyarp' de pppd.

L'option 'proxyarp' configure (surprise) une entrée proxy ARP dans la table ARP du serveur qui signifie en substance 'envoie moi tous les paquets destinés au client'. C'est la manière la plus simple de configurer le routage vers un client PPP unique - mais elle ne peut être utilisé pour le routage entre deux réseaux locaux - vous devez alors utiliser les routes réseau réelles qui ne pourront pas utiliser le proxy ARP.

Vous devez certainement avoir envie de fournir des adresses IP dynamiques aux utilisateurs qui se connectent. Vous pouvez le faire en associant une adresse IP dynamique sur chaque port d'appel. Créez alors pour chaque port d'appel un `/etc/ppp/options.ttyXX`.

Dedans, mettez simplement l'adresse IP locale du serveur et l'adresse IP qui doit être utilisé pour ce port. Par exemple

```
kepler:slip01
```

Remarquez en particulier que vous pouvez utiliser les noms d'hôtes valides dans ce fichier (je trouve que c'est la seule façon de se souvenir des adresses IP des machines importantes de mon réseau, les noms sont bien plus significatifs) !

26.5 Configurer pppd pour permettre aux utilisateurs de le lancer

Comme configurer une liaison ppp implique de configurer un périphérique dans le noyau (une interface réseau) et de manipuler les tables de routage, des privilèges spéciaux sont nécessaires - en fait la totalité des privilèges root.

Heureusement, `pppd` est prévu pour marcher correctement avec set uid root. Vous devez donc

```
chmod u+s /usr/sbin/pppd
```

Quand vous listez le fichier, il doit apparaître comme cela

```
-rwsr-xr-x   1 root     root        74224 Apr 28 07:17 /usr/sbin/pppd
```

Si vous n'avez pas ça, vous ne pourrez pas utiliser votre liaison ppp.

26.6 Créer un alias global pour pppd

Pour simplifier les choses pour les utilisateurs de notre connexion PPP, nous créerons un alias global (dans /etc/bashrc) qui est en fait une simple commande qui lancera ppp sur le serveur dès qu'ils seront connectés. Cela nous donne

```
alias ppp="exec /usr/sbin/pppd -detach"
```

Et il fait ceci

- exec : va remplacer le programme en cours (dans le cas d'un shell) par le programme qui va être lancé.
- pppd -detach : lance `pppd` et le laisse pas en tâche de fond. Cela assure que `pppd` existe et qu'il n'y a pas d'autres processus qui traînent.

Quand un utilisateur se connecte comme cela, il va apparaître après un 'w'

```
  6:24pm  up 3 days,  7:00,  4 users,  load average: 0.05, 0.03, 0.00
User     tty       login@  idle   JCPU   PCPU  what
hartr    ttyC0     3:05am  9:14                -
```

Et c'est tout... Je vous avais dit que c'était simple un serveur PPP de base !

27 Utiliser PPP avec une connexion null modem

C'est très simple - il n'y a pas de modem et ainsi les choses sont bien plus facile.

Tout d'abord, choisissez une machine comme 'serveur', configurez getty sur le port série pour que vous testiez si vous avez une connectivité en utilisant minicom sur le 'client'.

Une fois que ça fonctionne, vous pouvez enlever le getty A MOINS que vous vouliez être sûr que la connexion permet d'utiliser les paires nom utilisateur/ mot de passe pour les connexions entrantes. Comme vous avez un 'contrôle physique' des deux machines, je pense que vous n'avez pas besoin de ça.

Maintenant, sur le serveur, enlevez getty et assurez vous que les ports série sont configurés correctement sur les deux machines avec 'setserial'.

Tout ce que vous avez besoin maintenant est de démarrer `pppd` sur les deux systèmes. Je suppose que les connexions utilise une connexion par /dev/ttyS34. Ainsi sur les deux machines lancez la commande :

```
pppd -detach crtscts lock <local IP>:<remote IP> /dev/ttyS3 38400 &
```

Cela devrait lancer la connexion - mais pour le moment il n'y a aucun routage de spécifié. Vous pouvez tester la connexion en pingant l'autre machine. Si ça marche, éteignez la liaison en tuant le processus pppd

Le routage dont vous avez besoin dépend de ce que vous voulez faire exactement. Généralement, une des machines sera connecté sur un réseau Ethernet (et même plus) le routage nécessaire est donc exactement le même qu'entre un serveur PPP et un client.

Sur la machine équipée d'Ethernet, la commande sera donc

```
pppd -detach crtscts lock proxyarp <local IP>:<remote IP> /dev/ttyS3 38400 &
```

et sur l'autre machine

```
pppd -detach crtscts lock defaultroute <local IP>:<remote IP> /dev/ttyS3 38400 &
```

Si vous reliez deux réseaux locaux (avec une liaison série !) vous aurez besoin d'un routage plus complexe, vous pouvez utiliser /etc/ppp/ip-up exactement de la même façon qu'il est décrit plus haut.

Robert Hart
Port Hedland, Western Australia
Melbourne, Victoria, Australia August/October 1996 January/March 1997