Linux PPP, Foire Aux Questions

Par Alfred Longyear, adaptation Française René Cougnenc 27 Janvier 1994

Ce document contient les réponses aux questions fréquemment posées concernant l'implémentation de PPP sous Linux, il s'agit de la traduction Française et d'une mise en forme SGML du fichier contenu dans l'archive de distribution du paquetage PPP.

Depuis, Al Longyear diffuse un "PPP-HOWTO". Ce document ayant été réalisé avant, il conserve le nom de PPP-FAQ et est un peu différent du HOWTO original; en fait il est plus complet.

Table des matières

1 Mentions légales

2 Introduction

Voici les réponses aux questions fréquemment posées concernant l'utilisation de PPP sous Linux.

Envoyez toute correction que vous jugerez nécessaire à l'adresse `longyear@netcom.com`. **Attention !** consultez le texte original Anglais auparavant; l'auteur, Alfred Longyear, n'est pas concerné par les éventuelles erreurs introduites dans cette adaptation Française.

Ce document sera occasionnellement remis à jour, mais sûrement pas très régulièrement, réaliser ce genre de choses prend un temps non négligeable et n'est pas un travail passionnant. Sa diffusion sera tout aussi peu précise, nous tâcherons de le rendre disponible sur les principaux sites diffusant Linux en France. Le source original est réalisé en SGML et formaté par le système `Linux-doc-SGML` mis au point par Matt Welsh pour la documentation du système **Linux**. Vous disposerez par conséquent de ce texte dans différents formats, qui sont:

- Sortie DVI après formatage LaTeX, que vous pourrez par exemple afficher sous X-Window par la commande `xdvi`,
- Sortie PostScript, permettant directement l'impression papier du plus bel effet,
- Sortie HTML, qui vous autorisera à parcourir ce document sous forme hypertexte dans Chimera, Mosaic ou tout autre client WWW,
- Sortie ASCII, désaccentuée et pas très jolie, mais si vous ne pouvez pas faire autrement elle sera lisible sur n'importe quoi,
- Et bien sûr du texte source SGML ayant permis la réalisation de ces précédents formats.

La mise en page n'est pas extraordinaire, le kit Linux-doc-sgml n'étant pas destiné à réaliser des productions aussi peaufinées que celles faites directement sous LaTeX.

Tout au long de ce document, il est fait usage du terme "distant" pour désigner "système situé à l'autre bout de la connexion modem", que l'on appelle souvent "peer". Son adresse IP est l'adresse 'P-t-P' indiquée par la commande 'ifconfig'.

3 Questions et réponses

3.1 Qu'est-ce que PPP ?

PPP, ou "Point-to-Point Protocol", est un protocole internet "officiel". Il est destiné à l'échange de paquets IP (ou autre type de réseau) sur une liaison série. Sa description se trouve dans le RFC 1661.

Contrairement à ce que beaucoup de personnes pensent, PPP ne signifie pas du tout "Peer to Peer Processing", bien qu'il soit possible d'effectuer des communications "peer-peer" en utilisant TCP/IP sur une liaison PPP.

3.2 Mon entreprise (université) ne supporte pas PPP. Puis-je l'utiliser ?

En général, non. Une implémentation "classique" de PPP nécessite des modifications aux routages et interfaces réseau du système d'exploitation. Cela peut nécessiter la recompilation du noyau du système distant.

Ce n'est pas une tâche de simple utilisateur. Si vous arrivez à convaincre vos administrateurs système que PPP est une "bonne chose", alors vous aurez une chance de voir ce protocole installé. Si ils sont réfractaires à cette idée, vous ne pourrez probablement pas utiliser PPP.

Malgré tout, si vous utilisez un système qui est supporté par les gens qui vendent "TIA" ("The Internet Adapter") il y a un petit espoir. Nous n'avons pas beaucoup d'informations sur ce paquetage, mais nous avons entendu dire qu'ils comptent supporter PPP dans les prochaines versions. (Ces informations sont sans doute très anciennes, contactez-les directement. Vous trouverez des renseignements sur TIA sur `ftp.marketplace.com` dans le répertoire `/pub/tia`).

Un portage Linux est prévu.

Si votre système n'est pas supporté par TIA et que vous ne pouvez convaincre vos administrateurs de l'utilité de PPP, essayez d'utiliser le paquetage "term".

3.3 Où et sous quelle forme se trouve PPP ?

Il se décompose en deux parties, dont la première est dans le noyau, incluse depuis Linux 1.1.13.

NE REMPLACEZ PAS LE PILOTE FOURNI AVEC LE NOYAU PAR UNE VERSION TROUVEE DANS LE PAQUETAGE PPPD !!!

La seconde partie est le "démon pppd". Ce programme est INDISPENSABLE, ses sources se trouvent dans le fichier `ppp-2.1.2b.tar.gz` sur les sites diffusant Linux.

Pour les noyaux de version inférieure à 1.1.13, le pilote nécessaire est inclus dans un sous-répertoire de cette archive. C'est le seul cas où vous devez utiliser ces fichiers système, mais il est plutôt préférable que vous mettiez à jour votre noyau.

3.4 Je viens de récupérer PPP. Quelle est la suite des opérations ?

Lisez la documentation fournie. Elle a été écrite pour vous !

Commencez par lire les fichiers README et README.linux. Diverses autres sources de documentation sont indiquées ci-après.

3.5 Quelles autres documentations sur PPP sont disponibles ?

(Où est la doc ? Où est la FAQ ? etc.)

Il existe plusieurs sources d'information concernant le protocole PPP tel qu'il est implémenté sous Linux.

- Le fichier README contenu dans l'archive des sources.

- Le fichier README.linux au même endroit.
- Le document Linux Net-2-HOWTO.
- Le "Network Admin Guide".
- La page de manuel du programme "pppd".
- La FAQ de PPP (Ce n'est pas le document que vous lisez ici).

Le fichier HOWTO peut se télécharger sur les sites habituels diffusant Linux, par exemple sur `ftp.ibp.fr` dans le répertoire `/pub/linux/docs/HOWTO`.

Le `Network Admin Guide` fait partie du projet de documentation Linux, vous trouverez ce livre sous forme électronique au même endroit. Il sera prochainement publié par O'Reilly et disponible en version Française. Si vous désirez un document de présentation professionnelle, vous pourrez alors acheter cet ouvrage plutôt que de le télécharger et l'imprimer vous-même.

Les pages de manuel sont fournies avec les sources. Il vous faudra les placer dans les répertoires standard, `/usr/man/man8` pour que la commande `man` puisse les traiter. Vous pouvez bien entendu les examiner directement à l'aide de `groff`.

La FAQ de PPP décrit le protocole en lui-même et ses diverses implémentations sur différents systèmes. Vous trouverez ce document dans le forum Usenet `comp.protocols.ppp`, ou encore dans `news.answers` et il est bien entendu archivé sur `rtfm.mit.edu` dans le répertoire `/usenet`. Il est à l'heure actuelle composé de huit parties.

3.6 Où dois-je poser des questions relatives à PPP ?

Il est préférable de les poser dans le groupe `comp.protocols.ppp`. Il est fait pour cela. Hélas, beaucoup d'utilisateurs ont tendance à exposer d'abord leur problèmes dans les groupes `comp.os.linux.*`, où ils reçoivent des réponses bien que ce ne soit pas l'endroit idéal.

Très peu de questions sur PPP sont directement liées à son implémentation sous Linux. La plupart des problèmes posés sont d'ordre général, concernent l'utilisation du paquetage PPP et les réponses sont applicables à n'importe quel autre système d'exploitation.

Aussi, pour réduire le trafic dans les groupes Linux, et pour toucher les vrais spécialistes de PPP, si vous devez utiliser Usenet à ce sujet, utilisez absolument le groupe `comp.protocols.ppp`.

3.7 PPP ne marche pas, pppd se termine après l'appel. SOS !!!

C'est l'une des questions les plus irritantes. Nous nous rendons bien compte que c'est un appel à l'aide, mais il est inutile de poster un tel message SANS AUCUNE AUTRE INFORMATION. La plupart des gens susceptibles de pouvoir vous aider l'ignoreront, purement et simplement.

Il vous faut fournir les logs système (syslog) produits lorsque vous lancez le programme pppd avec l'option `debug`. De plus, si vous utilisez `chat` pour l'appel du service, utilisez son option `-v` pour obtenir une trace du dialogue.

Fournissez également le log de démarrage de votre noyau Linux. Il informe des différentes configurations propres à votre système, comme la version de PPP, le type de port série, etc.

Enfin, rajoutez toute information que vous pensez relative à votre problème. Bien entendu, votre machine, disques durs, boutons de souris, terminaux, n'ont aucun intérêt... Ce qui compte est surtout le système que vous tentez de contacter, la version de PPP ou de serveur de terminaux qu'il utilise, les types et vitesses de modem de part et d'autre, etc.

Faites attention, éditez soigneusement ces informations, en supprimant éventuellement tout numéro de téléphone, login et mot de passe susceptibles d'apparaître. Ils n'ont aucune utilité pour le déboguage, et les diffuser sur Usenet ne serait pas très malin. Supprimez aussi les lignes qui ne viennent pas du noyeau ou de pppd.

Ne postez jamais intégralement le log du programme `pppd` lancé avec l'option `kdebug 7` !

Si le problème demande une analyse du flux de données, vous serez contacté par courrier électronique et l'on vous demandera la trace par ce biais. Usenet coûte déjà bien trop cher pour beaucoup de gens.

L'information est écrite sous plusieurs niveaux. Les messages de déboguages sont au niveau `debug`. Ceux d'information le sont au niveau `info`, les erreurs au niveau `error`. Incluez tous les niveaux par le groupe `local2`, issu du processus `pppd`.

De plus, surtout ne supprimez pas les informations horaires, elles sont très importantes.

3.8 Je cherche une implémentation de PPP ailleurs que sous Linux.

Existe-t-il des versions pour HP-UX, AIX, ou... bien d'autres systèmes ?

Lisez les documentations mentionnées plus haut.

AIX sera supporté dans la version 2.2 du paquetage `pppd`. HP-UX n'est à notre connaissance supporté que par la version commerciale MorningStar.

Si vous ne trouvez vraiment aucun renseignement, demandez dans `comp.protocols.ppp`, mais surtout pas dans un groupe Linux, qui n'est en aucun cas concerné.

3.9 Saviez vous qu'il existe une autre implémentation nommée "dp" ?

Oui, nous sommes au courant. Le paquetage "dp" fut envisagé au tout début de l'implémentation de PPP sous Linux. Il est très bien, il supporte l'appel à la demande. Il ne peut fonctionner que sur un système supportant les `streams`, c'est à dire principalement le nouveau SunOS (Solaris).

Linux, pour l'instant, ne supporte pas les `streams`.

Il existe d'autres ensembles PPP disponibles par l'Internet. Le paquetage `Portable ppp` ressemble beaucoup au code TIA; une autre implémentation se nomme tout simplement `ppp`, et on trouve du code pour PPP dans l'excellent paquetage `KA9Q`.

De toutes ces versions, `pppd` était le plus proche de ce que Linux nécessitait pour assurer un portage correct.

(Si vous désirez plus d'informations sur ces autres implémentations, demandez dans `comp.protocols.ppp`.

3.10 Quels RFC décrivent le protocole PPP tel qu'implémenté sous Linux ?

L'implémentation courante de PPP est un mélange de plusieurs. La plus grande partie du code de PPP est réalisée selon les RFC 1331 et 1332. Ces documents sont maintenant obsolètes; le 1331 fut remplacé par le 1548 qui, à son tour, fut remplacé par le 1661 six mois plus tard. Vous trouverez une liste complète dans la FAQ PPP.

La plupart des implémentations de PPP converseront sans problèmes avec celle de Linux.

Tous les RFC 1134, 1171, et 1172 (et, de ce fait, 1055) sont maintenant obsolètes. Ils ne sont intéressants que si vous comptez déboguer une connection avec une ancienne implémentation de PPP, et si vous vous demandez pourquoi, par exemple, elle vous a demandé l'option IPCP 2 avec une longueur de 4 seulement, et un type de compression 0x0037. (Il en circule encore de nombreuses versions, faites attention).

Le PPP de Linux ne supportera pas ces vieilleries.

3.11 PPP peut il converser avec SLIP ?

Non. SLIP fonctionne avec SLIP, PPP fonctionne avec PPP.

Quelques distributeurs peuvent offrir des produits fonctionnant à la fois en SLIP et PPP. Toutefois, ils doivent être configurés dans l'un de ces deux modes. Il n'existe actuellement aucune méthode permettant de déterminer à la connexion quels types de protocoles SLIP, ou PPP, est attendu.

3.12 Quel est le meilleur ? PPP ou SLIP ?

CELA DEPEND DE NOMBREUX FACTEURS !

Les gens qui posent ce genre de question n'ont en général pas lu le moindre document sur ces protocoles, à commencer par le `Net-2-HOWTO`.

Un articles très intéressant sur ce sujet est disponible sur le serveur Web de Morning Star, www.morningstar.com.

3.13 Quel est le mieux ? L'authentification CHAP ou PAP ?

Si vous avez le choix, prenez `CHAP`. Sinon, `PAP` sera mieux que rien.

3.14 Que doit contenir le fichier `/etc/ppp/pap-secrets` ?

Le protocole `pap` est le plus souvent implémenté sous la forme de votre nom d'utilisateur associé à un mot de passe. Vous devez mettre le nom du système distant, votre nom d'utilisateur, et le mot de passe.

Prenons un exemple: L'utilisateur `abbot` veut appeler `costello`. L'entrée sera la suivante:

```
#remote   account   password   IP address list
costello  abbot     firstbase
```

3.15 Que doit contenir le fichier /etc/ppp/chap-secrets ?

Le problème le plus courant est que les gens ne reconnaissent pas cette négociation `CHAP` par un couple de `secrets`. Chacun des deux ordinateurs devant se relier doivent connaitre les deux `secrets` pour que la communication aboutisse.

Par exemple, si `abbot` veut communiquer avec `costello`, le fichier sur `abbot` contiendra:

```
#local      remote        secret       IP address list
abbot       costello      firstbase
costello    abbot         who
```

Et celui de `costello` sera:

```
#local      remote        secret       IP address list
abbot       costello      firstbase
costello    abbot         who
```

3.16 J'ai des erreurs lors de la compilation...

J'ai des erreurs lors de la compilation du noyau quand j'inclus le pilote ppp.c. Ce noyau est en version 1.1.8. Que faire ?

Avez-vous édité le fichier ppp.c pour lui indiquer la configuration du noyau que vous utilisez ? Il y a deux `#define` qui doivent être correctement définis si vous comptez compiler le code pour PPP.

```
.---------------------.------------------.-------------------------.
|                     |                  |                         |
|  version du noyau   |     NET02D       |     NEW_TTY_DRIVERS     |
|_____________________|__________________|_________________________|
|                     |                                            |
|  < 1.0.0            |        MISE A JOUR NECESSAIRE !!!!         |
|_____________________|__________________.________________________|
|  1.0.0  - 1.0.*     |     #define      |     #undef              |
|  1.1.0  - 1.1.3     |     #define      |     #undef              |
|  1.1.4  - 1.1.12    |     #undef       |     #undef              |
|_____________________|__________________|_________________________|
|                     |                                            |
|  1.1.13             |       MISE A JOUR NECESSAIRE !!!           |
|_____________________|__________________.________________________|
|                     |                  |                         |
|  1.1.14 - ... NOTE  |     #undef       |     #define             |
|_____________________|__________________|_________________________|
```

Le troisième `#define` au début du fichier s'appelle `OPTIMIZE_FLAG_TIME`. Il peut être ou non défini avec n'importe quelle version du noyau supportant PPP. Vous trouverez plus de détails en lisant ppp.c.

NOTE: A partir du noyau 1.1.14, ne remplacez pas les fichiers `drivers/net/ppp.c` et `include/linux/ppp.h`. Ils sont adaptés à votre version.

Donc, comme vous le voyez dans le tableau ci-dessus, pour le noyau 1.1.8 par exemple, il faut définir à la fois NET02D et NEW_TTY_DRIVERS. De même, une version 1.1.18 contient déjà tout ce qu'il faut, vous n'avez aucune modification à faire.

3.17 Que signifie le message: `unable to create pid file` ?

Vous devez créer le répertoire `/var/run`. Dans certaines distributions, il s'agit d'un lien symbolique vers le répetoire `/etc`.

Ce n'est qu'un avertissement, qui n'empêche pas PPP de fonctionner correctement. Toutefois, le script `ppp-off` dépend de ce fichier, il est donc préférable de régler le problème.

L'entête POSIX `path.h` définit l'endroit du fichier `pid` sous le nom `_VAR_RUN`. Si vous désirez utiliser un autre répertoire pour PPP ou d'autres programmes, changez la valeur de cette définition et recompilez les programmes.

3.18 Que veut dire `/etc/ppp/options: no such file or directory` ?

Vous devez créer le répertoire `/etc/ppp` et y mettre un fichier nommé `options`. Il doit être lisible par le processus `pppd` (root).

Ce fichier peut être vide, tout simplement. Dans ce cas, utilisez la commande `touch` pour le créer. Lisez la page de manuel de `pppd` pour obtenir une description de ce fichier.

Il est possible de compiler `pppd` pour qu'il n'utilise pas ce fichier; dans ce cas il faut supprimer (commenter) la ligne `SECURE_FLAGS = -DREQ_SYSOPTIONS=1` dans le `Makefile`.

3.19 Le programme “dip” référence PPP...

Lorsque je tente d'utiliser `mode ppp` depuis `dip` j'obtiens une erreur disant que le mode PPP n'est pas supporté. Est-il prévu d'utiliser `dip` avec PPP ?

Le programme `dip` contrôle l'établissement d'une liaison SLIP. Puisque le processus `pppd` doit faire plusieurs choses à des moments différents que dans un lien SLIP, il y a fort peu de chances pour que `dip` et `pppd` puissent s'entendre un jour.

La commande `dip` peut néanmoins être utilisée pour l'appel téléphonique et le lancement de PPP sur le système distant. Dans ce cas il est préférable de le lancer en tant qu'argument `connect` de `pppd`.

3.20 Comment arrêter PPP ? Existe-t-il un “dip -k” pour PPP ?

Non, il n'y en a pas.

Dans le répertoire `chat`, vous trouverez un script appelé `ppp-off`. Il arrêtera la liaison PPP de la même façon que votre `dip -k` .

Le voici ci-dessous. Copiez-le dans un fichier que vous rendrez exécutable par la commande `chmod +x` et le tour sera joué.

```
#!/bin/sh
DEVICE=ppp0
#
# Si le fichier pid de ppp0 est present, alors c'est que le programme
# tourne. Arretons-le.
if [ -r /var/run/$DEVICE.pid ]; then
        kill -INT `cat /var/run/$DEVICE.pid`
#
# Si ca n'a pas marche, c'est qu'il n'y a plus de processus actifs
# correspondant a ce pid. Cela peut aussi signifier que le fichier
# de verrouillage a ete laisse en place. Il faut supprimer ce verrou.
        if [! "$?" = "0" ]; then
                rm -f /var/run/$DEVICE.pid
                echo 'ERROR: Removed stale pid file'
                exit 1
        fi
#
# Succes. Laissons pppd faire son propre menage.
        echo 'PPP link to $DEVICE terminated.'
        exit 0
fi
#
# Le processus pppd n'est pas actif sur ppp0
echo 'ERROR: PPP link is not active on $DEVICE'
exit 1
```

3.21 Le processus pppd ne se termine pas lorsque le modem se déconnecte. Je dois l'arrêter manuellement pour que getty reprenne la main.

Avez-vous lancé le programme avec le paramètre `modem` ?

Ce paramètre indique à `pppd` d'honorer les signaux relatifs au statut du modem sur la ligne, qu'il ignore par défaut. Il est décrit dans la page de manuel de `pppd`.

Est-ce que votre modem reflète bien l'état de la communication par la ligne DCD ?

La commande `Hayes` positionnant ce comportement est généralement `&C1`. Si vous remettez l'appareil à zéro par `ATZ` lors de la communication, assurez-vous bien qu'il gère toujours le signal DCD. La configuration d'usine fournie par beaucoup de constructeurs est hélas d'ignorer ce `Carrier Detect`.

Le signal DTR est généré par l'ordinateur et indique au modem de raccrocher la ligne. La séquence Hayes pour cela est généralement `&D1` ou `&D2`, la seconde étant préférée pour utiliser PPP. La configuration par défaut des constructeurs étant souvent d'ignorer DTR.

N'utiliseriez-vous pas un câble de liaison bon marché, incomplet, ne transmettant pas la ligne DCD ? Les câbles de **Macintosh classic**, par exemple, ont la réputation de poser ce problème. Cet ordinateur n'utilise pas ce signal.

Pour les connexions en mode serveur, avez-vous fait exécuter pppd correctement ? Le processus doit être lancé par un appel `exec`; si vous le lancez depuis le shell comme une simple commande, ce sera le shell qui recevra SIGHUP et non pas le processus pppd.

Le script doit avoir un format du type suivant:

```
#!/bin/sh
exec pppd -detach modem ...
```

3.22 Les transferts FTP échouent quand je fais un "put" mais se passent bien lorsque je fais un "get".

Est-ce que le contrôle de flux est bien en service ? Il se valide par l'option `crtscts` de pppd, pour ce qui est du contrôle de flux matériel RTS/CTS, et par l'option `xonxoff` pour XON/XOFF. Sans contrôle de flux, vous perdrez des caractères et le résultat sera désastreux pour la compression vj, entre autres.

3.23 Comment utiliser le contrôle de flux XON/XOFF ?

Le mieux est de ne pas l'utiliser, et de choisir RTS/CTS. Toutefois, si vous ne pouvez faire autrement, vous devez opérer selon les trois points suivants:

- Vous devez spécifier l'option `xonxoff` à pppd. Cela indiquera au programme de configurer le port série pour le contrôle de flux XON/XOFF, en indiquant ces caractères au pilote.
- Vous devez spécifier les deux caractères XON et XOFF dans le paramètre `asyncmap` de pppd. Cela indiquera au système distant qu'il doit encoder XON et XOFF lorsqu'il doit les envoyer en tant que caractères normaux. Il suffit en principe d'indiquer `asyncmap a0000`.
- Bien sûr, n'oubliez pas d'indiquer au modem d'utiliser ce type de contrôle de flux. Consultez la documentation de votre appareil pour déterminer la commande Hayes nécessaire.

3.24 Le modem ne semble jamais se connecter aux vitesses rapides.

Mettez la vitesse désirée dans la commande de pppd. Si vous ne l'indiquez pas, ce sera celle qui est initialisée au moment ou vous lancez le programme qui sera utilisée, et elle peut être quelconque : tous les programmes ne remettent pas correctement les paramètres du port série qu'ils ont trouvé en arrivant. Certains peuvent même laisser des valeurs étranges amenant un fonctionnement curieux (Une vitesse de 0, par exemple !).

3.25 Pourquoi ne puis-je lancer `pppd` si je ne suis pas root ?

Le processus `pppd` a besoin de modifier le paramétrage du réseau, et ceci n'est autorisé qu'au super-utilisateur. Si vous désirez pouvoir lancer `pppd` sous un autre nom d'utilisateur, le programme doit alors être "suid root":

```
chown root pppd
chmod 4755 pppd
```

Si vous voulez n'autoriser `pppd` qu'à un petit groupe de personnes, attribuez-le à ce groupe et interdisez à tous les autres son exécution.

3.26 Je peux atteindre la machine distante, mais aucune autre.

N'auriez-vous pas oublié le paramètre `defaultroute` ? Il rajoute une route par défaut dans vos tables de routage, de manière à ce que les paquets vers toutes les autres adresses IP soient envoyés par l'interface PPP.

Le programme ne remplacera pas une route par défaut déjà existante. Ceci est volontaire, cela évite de détruire la route par défaut vers d'éventuels routeurs Ethernet par accident. Dans ce cas, un message d'avertissement est envoyé au système de "log" de la machine, via `syslog`, et ce sera à vous d'effectuer les opérations que vous jugerez nécessaires.

3.27 J'ai une route par défaut mais je ne peux toujours pas me connecter ailleurs !

Cela ne provient pas de votre système Linux local, mais probablement d'un problème de routage sur la machine distante.

Le système distant a besoin d'une route vers vous, tout comme vous en avez une vers lui. Ceci peut être réalisé de quatre façons différentes. Chaque méthode a ses avantages et inconvénients; vous devez en choisir une, et une seule.

1. Utiliser une route de type "host". Sur chaque machine du site distant, rajouter une route vers votre adresse IP Linux, avec comme passerelle le serveur de terminaux que vous utilisez pour votre accès local. Cette méthode fonctionnera si vous avez un nombre limité de machines et un réseau très simple.

2. Utiliser une route de type "network". Subdivisez les adresses IP distantes de manière à ce que l'adresse IP locale à votre machine Linux, celle de votre serveur distant, et celle de l'interface Ethernet de ce dernier soient dans le même domaine IP.

   Cette méthode fonctionnera si vous possédez des adresses à distribuer. Elle sera parfaite si vous possédez un réseau de classe B et pouvez assigner toutes les adresses distantes dans le même domaine IP. Ajoutez alors une route réseau sur chacune des passerelles et routeurs de façon que toute adresse du réseau distant soit accédée par le serveur de terminaux. Beaucoup de configurations possèdent beaucoup de machines mais très peu de routeurs. (A `sii.com`, nous avons plus de 300 systèmes actifs et seulement 3 routeurs).

3. Utilisez le programme `gated` sur toutes les passerelles et sur le serveur de terminaux. Cela fera diffuser par le serveur de terminaux des informations vers les passerelles, indiquant qu'il peut accepter des paquets pour votre adresse IP. Puisque les machines auront une route par défaut vers l'une de ces passerelles, ces dernières génèreront les requêtes ICMP de redirection et le routage s'établira automatiquement.

4. Utilisez "`proxy ARP`" sur le serveur de terminaux. Cette méthode ne fonctionnera que si votre adresse IP locale correspond à l'un des domaines IP des cartes réseau.

En résumé, il n'y a pas de solution miracle, simple et unique. Vous devez faire votre choix parmi l'une de ces quatre là.

3.28 Je n'arrête pas de recevoir le message disant que le "magic number" est toujours refusé (NAK). Le système ne se connecte pas.

La probabilité que les deux systèmes aient choisi le même nombre magique est d'une chance sur un billion. Si vous obtenez continuellement cette erreur, prenez un billet de loterie et/ou surveillez votre femme :-)

Les deux causes les plus courantes de ce problème sont:

1. Le modem s'est déconnecté immédiatement après avoir réalisé la connexion et l'accès au système distant. Beaucoup de modems sont configurés pour renvoyer l'écho des données qu'on leur injecte, et vos dialoguez avec l'écho local.
2. Les programmes réalisant PPP ne sont pas en service sur le système distant au moment où vous tentez d'établir le lien. Est-ce que la machine distante est bien configurée pour PPP ? Est-ce que les programmes s'y trouvent bien accessibles, tant en chemin d'accès qu'en permissions ?

   Cela tendrait à indiquer que c'est le shell qui vous réalise un écho local des données que vous envoyez.

   Quoi qu'il en soit, le système Linux envoie des données que l'autre côté retourne immédiatement. Ce n'est pas une condition acceptable, vous avez ce que l'on appelle une "boucle".

3.29 Je tente de me connecter à un Telebit Netblazer, et il termine par le message "`Could not determine local IP address`".

Le Netblazer n'a pas votre adresse IP. Vous n'avez pas non plus votre adresse IP. La liaison ne s'effectuera pas tant que ces deux paramètres ne seront pas connus.

On a du vous donner une feuille de papier sur laquelle ces deux adresses IP sont indiquées. Vous devez signaler au Netblazer l'adresse à utiliser. Indiquez alors ces deux adresses, locale et distante, comme paramètres lors du lancement du programme `pppd`, selon le format suivant:

```
adresse_locale:adresse_distante
```

N'oubliez surtout pas les deux points (`:`) de séparation.

3.30 Je tente de me connecter à un Telebit Netblazer, et il termine par le message "`Could not determine remote IP address`".

Lisez donc la réponse à la question précédente...

3.31 Je ne peux pas "`pinguer`" mon adresse IP locale.

Vous ne pouvez pas parceque vous n'avez pas de route vers cette adresse. C'est le fonctionnement normal. N'essayez pas.

Si vous voulez faire un `ping` sur votre propre système, utilisez l'adresse `loopback`: 127.0.0.1.

Vous devriez pouvoir "`pinguer`" l'adresse distante. Toutefois, certains serveurs interdisent cette opération.

En règle générale, n'essayez pas `ping` sur l'une ou l'autre de ces adresse. Choisissez en une troisième, que vous savez accessible sur le réseau distant.

3.32 Puis-je utiliser la même adresse IP locale pour toutes les lignes de mon serveur PPP ?

Oui.

L'adresse locale n'est pas significative pour le système local. Vous devez avoir une adresse IP **distante** unique. Le routage est fait en fonction de l'adresse distante et non locale.

3.33 J'utilise un serveur de terminaux Xyplex et `pppd` se plaint de se voir refuser le protocole `fffb`. Qu'est-ce que c'est que ça ?

Le serveur de terminaux Xyplex est troublé par la compression d'entêtes Van Jacobson. Utilisez l'option `-vj` de `pppd` pour ne pas l'utiliser.

3.34 Le log contient souvent `Alarm`. Est-ce grave ?

Non.

Cela signifie simplement qu'une mesure de temps s'est écoulée, c'est une partie nécessaire de la phase d'établissement du protocole.

3.35 Le log indique `protocol reject for protocol c025`. Qu'est-ce que c'est que cette histoire ?

Le site distant voudrait mettre en service le protocole d'analyse de qualité de liaison sur la machine Linux. (LQR, "Link Quality Reporting"). Ce protocole n'est pas encore supporté, ce n'est pas une erreur. Linux signale en substance: "L'autre m'a demandé ça, je lui ai dit que je ne sais pas le faire pour l'instant et qu'il ne m'embête pas avec".

L'implémentation MorningStar tentera toujours LQR. C'est normal.

3.36 La connexion s'effectue mais la séquence d'initialisation semble ne jamais se terminer. Le périphérique n'est jamais "UP", et finalement le programme raccroche. Que se passe-t-il ?

Essayez avec l'option `debug` et examinez les traces système. (Vous aurez de toutes façons besoin de ces traces si vous comptez demander de l'aide). Si elles montrent que vous envoyez la séquence " LCP-request" continuellement et que la valeur d'"id" ne s'incrémente pas, c'est qu'il n'y a pas de PPP à l'autre bout.

Les trois causes les plus courantes sont:

- PPP n'est pas lancé sur la machine distante. Vous envoyez des demandes de négociation à un programme quelconque, qui est probablement en train de vous répondre "Qu'est-ce que c'est que ce charabia ?"

  Assurez-vous que le protocole PPP est bien lancé à l'autre bout avant d'entrer dans la phase de négociation du protocole.

  Essayez d'utiliser un programme de communications normal et simulez la session à la main. Vous devriez alors voir nettement la tentative de négociation PPP vous arriver, c'est facile à identifier: il s'agit de trains d'environ 16 caractères, contenant beaucoup d'accolades ouvrantes. Il ne doit pas y avoir de retour-chariot dedans et ils sont envoyés par saccades.

- La liaison n'est pas "8 bits clean". Cela signifie que pour utiliser PPP, vous devez être configuré en 8 bits de données, pas de parité et un bit de stop. PPP ne peut fonctionner qu'en 8 bits.

  Le programme `pppd` passera automatiquement la ligne sous ce format. La machine distante doit également se conformer à cette configuration sous peine de nombreuses erreurs de format et parité.

  PPP encode certains caractères. Il ne lui est pas possible de travailler au niveau bit comme kermit peut le faire. PPP ne **fonctionne pas** sur une liaison 7 bits.

  Il y a une option de compilation dans le pilote ppp.c (qui fait partie du noyau) appelée CHECK_CHARACTERS, qui ajoute du code destiné à effectuer des tests supplémentaires sur les caractères reçus. Elle pourra vous indiquer si la parité était validée ou si le système distant envoyait du 7 bits.

- Le système distant est configuré pour demander l'authentification PAP ou CHAP. Vous n'avez pas configuré votre système local pour le faire. Par conséquent, l'autre côté refuse tous vos paquets jusqu'à en rencontrer un d'identification, que vous n'enverrez jamais.

  Dans ce cas, il vous fait soit configurer la machine distante pour qu'elle ne demande pas d'authentification, ou configurer le système local pour qu'il le fasse.

3.37 La connexion échoue sur une erreur `ioctl(TIOCSCTTY)`.

Utilisez l'archive ppp-2.1.2b.tar.gz. Il s'agit d'un bug qui n'était pas géré avant la diffusion de ppp-2.1.2a.

3.38 J'essaie d'utiliser `proxyarp`. La fonction proxyarp n'arrive pas à trouver l'adresse MAC avec Linux 1.0.

Utilisez l'archive ppp-2.1.2b.tar.gz. Le programme `pppd` était compilé par erreur sous Linux 1.1.8 et utilisait les définitions Net-3 plutôt que celles de l'ancien Net-2.

3.39 Bon sang ! J'ai récupéré ppp-2.1.2b et il dit qu'il a besoin des versions 4.6 des bibliothèques partagées ! Alors ???

Désolé, on s'est trompé, nous travaillons avec vos programmes du futur, que voulez-vous. Un jour où l'autre, vous disposerez également de ces librairies.

En attendant,il va vous falloir recompiler le code vous-même, avec les bibliothèques dont vous disposez. C'est très facile:

Allez dans le répertoire `pppd`, effacez le mauvais binaire, et tapez la commande "`make`". Faites de même dans le répertoire `chat` si vous comptez utiliser ce programme.

Une autre solution est de vous procurer une version de ces bibliothèques et de jouer les Alpha-testeurs :-)

3.40 La connection échoue avec des erreurs comme `ioctl(TIOCGETD): I/O error` ou bien `ioctl(PPPIOCSINPSIG): I/O error` . Allo, Docteur ?

Regardez les messages du noyau lors de l'amorçage de votre système Linux. Si il affiche `PPP version 0.1.2`, vous avez une version beaucoup trop ancienne du pilote `ppp.c`.

Si vous voyez `PPP version 0.2.7`, vous disposez de la version à jour, mais qui cependant n'a pas été compilée avec les mêmes définitions pour les valeurs d'`ioctl()`.

Vérifiez que vous disposez bien d'un fichier, `ppp.h`, qui doit se trouver dans le répertoire `include/linux` des sources du noyau. Dans ce cas, recompilez le noyau, puis le programme `pppd`.

3.41 Quelquefois, j'obtiens les messages "`ioctl(PPPIOCGDEBUG): I/O error`", "`ioctl(TIOCSETD): I/O error`" et "`ioctl(TIOCNXCL): I/O error`". Pourquoi donc ?

Parceque le système distant a raccroché le téléphone. Les pilotes tty rétablissent alors la discipline correcte sur le port série, et ces erreurs sont le résultat du processus `pppd`, tentant de faire la même chose.

Tout cela est parfaitement normal, il n'y a pas d'inquiétudes à avoir.

3.42 J'essaie d'utiliser le réseau `merit`, et je me fais jeter.

Des utilisateurs de ce réseau ont signalé qu'il nécessite PAP. Avez-vous essayé l'authentification PAP ?

3.43 Mon `ifconfig` affiche autre chose que ce qui suit. Il indique l'interface ppp comme "`unknown`" et une adresse MAC bizarroïde.

```
ppp0      Link encap Point-to-Point Protocol
          inet addr 192.76.32.2  P-t-P 129.67.1.165  Mask 255.255.255.0
```

Ce n'est pas grave. Cet affichage n'est là que pour information. Si vous utilisez un noyeau 1.1. récent, mettez à jours vos programmes réseau.

3.44 Mon système ne marche pas. J'utilise `slattach`,`ifconfig` et `route` et ça ne marche toujours pas ! J'ai lu ça dans le Net-2-HOWTO; où est l'erreur ?

N'utilisez jamais `slattach` et `ifconfig` avec PPP. Ces programmes ne sont utiles que pour SLIP. Le processus `pppd` s'occupe de ces fonctions tout seul au moment opportun, c'est à dire après l'échange des protocoles LCP et IPCP.

Vous ne pouvez pas remplacer `pppd` par `slattach` et `ifconfig`. La plus grande partie du protocole PPP se trouve dans `pppd`. Le noyau ne connait que IP (et IPX dans l'avenir).

La route vers le site distant sera automatiquement ajoutée par `pppd`. Il n'y a pas d'option pour supprimer cette opération. Le programme se terminera si la route ne peut pas être rajoutée a la table de routage.

La route par défaut peut être automatiquement ajoutée, ou non. Ceci est controlé par l'option `defaultroute`. Si vous avez déjà une route par défaut, elle ne sera pas modifiée.

Si vous devez router un réseau entier, mettez la commande dans le script `/etc/ppp/ip-up`. Les paramètres passés à ce script sont:

- $0 - nom du script (/etc/ppp/ip-up ou /etc/ppp/ip-down)
- $1 - nom de l'interface réseau (comme ppp0)
- $2 - nom du périphérique (comme /dev/cua0)
- $3 - vitesse du périphérique en bits par seconde (comme 38400)
- $4 - l'adresse IP locale
- $5 - l'adresse IP distante

3.45 D'accord, j'ai les paramètres. Mais je veux la route vers le réseau et non pas la route vers le host. Comment faire ?

Sur `sunsite.unc.edu` se trouve un paquetage du nom de `devinfo.tar.gz`. Il contient quelques petits utilitaires pratiques qui extraient les données concernant un périphérique et font différentes opérations sur les notations d'adresses IP.

La documentation est fournie sous forme de pages de manuel.

Par exemple, si vous voulez router le domaine IP entier vers le site distant, vous pouvez mettre les choses suivantes dans le script `/etc/ppp/ip-up`.

Bien entendu, si les valeurs ne sont pas variables, il vous suffit d'utiliser les entrées appropriées avec la commande `route`.

```
# Obtention du masque reseau pour l'interface ppp0 (ou autres)
NETMASK = ‘devinfo -d $1 -t mask‘

# Obtention du domaine IP (sans l'adresse host en supprimant les bits ad hoc)
DOMAIN = ‘netmath -a $5 $NETMASK‘

# Rajout de la route vers le reseau que nous connaissons maintenant
route -net add $DOMAIN gw $5
```

3.46 Quand Linux supportera-t-il la numérotation sur demande ? J'en ai absolument besoin !

Mercredi prochain !

Plus sérieusement, la numérotation sur demande devait se trouver dans la prochaine version. Toutefois, le paquetage `pppd` est l'oeuvre de **Paul Mackerras** (nous ne faisons que le portage) et il a préféré commencer par ce qu'il préférait, c'est à dire la compression BSD.

Aussi, cela viendra après que ce code de compression soit diffusé. Il y a déjà quelques volontaires s'étant proposé pour travailler sur le projet. Vous pouvez vous y mettre aussi !

3.47 Quid du filtrage ? Quand allez-vous implémenter ça ?

Rien n'est prévu à ce sujet dans le code de PPP. Utilisez le code `ipfirewall`, c'est disponible sur `sunsite.unc.edu`. Aidez l'auteur à déboguer la chose, et ça fera ce que vous désirez que ça fasse.

3.48 Quid de IPX ?

L'addition du support pour IPX se fait tranquillement. J'ai (Al Longyear) commencé à y travailler pour quelqu'un me l'ayant demandé sur l'IRC (Internet Relay Chat). Je dois d'abord approfondir les manuels Novell pour comprendre quelles sont les valeurs correctes pour les fonctions de routages nécessaires pour IPXCP. (IPXCP est le protocole de contrôle d'IPX).

3.49 Quid de NETBIOS ?

Il existe un protocole PPP Netbios. Toutefois, la meilleure solution serait que vous utilisiez TCP/IP et le code “`samba`”.

Microsoft et compagnie ont utilisé le protocole PPP Netbios. C'est généralement un savant mélange très propriétaire et une implémentation ne fonctionne pas forcément avec une autre.

Je laisse ces machins à quelqu'un d'autre.

3.50 Je viens de regarder /proc/net/dev comme indiqué dans la documentation et c'est vide...Pourquoi ?

Avez-vous juste tapé la commande "`ls -l /proc/net`" en vous étonnant de trouver une taille de zéro octets ? Si c'est le cas, c'est normal. Vous devez faire:

```
cat /proc/net/dev
```

Vous devriez alors voir le contenu de ce fichier, qui ne doit pas être vide. Sa taille est toujours indiquée comme étant nulle, c'est le système de fichiers "`proc`"qui veut ça. Ce ne sont pas de vrais fichiers, ils sont simulés par le noyau. Ne tenez pas compte de la taille, tapez la commande indiquée.

3.51 Je ne peux pas connecter de/vers mon code Windows NT (Daytona) avec le PPP Microsoft. Pourquoi diable ?

Le PPP de Microsoft envoie une authentification de type 0xC207, alors que le PAP normal est 0xC203.

Microsoft a choisi d'utiliser un protocole d'authentification non standard dans Windows NT. C'est leur droit le plus strict, à condition qu'ils aient enregistré leur numéro de protocole avec l'IANA. (C'est le cas). Toutefois, ils sont dans l'obligation de supporter la séquence de refus du protocole, ce qui est logique. Il y avait un bogue dans les bêta-versions Daytona, qui peut encore se trouver dans les versions définitives, qui fait que cette séquence n'est pas supportée. Il insiste pour que le système Linux supporte cette authentification.

Par conséquent, allez dans la configuration des paramètres TCP/IP pour Daytona et supprimez la séquence d'authentification. Du coup, tout marchera avec des implémentations standard de PPP, comme celle de Linux.

Le protocole d'authentification de Microsoft est du type PAP, avec leur propre algorithme d'encryptage des mots de passe. Le PAP normal envoie les mots de passe en clair, ce qui violerait la sécurité C2.

3.52 Avez-vous un lecteur de courrier compatible PPP ? et pour les News ?

Heu... ?

Vous vous trompez de groupe si vous cherchez des bricoles MSDOS. PPP n'a rien à voir du tout avec l'agent de transport de courrier ! Ils sont tous "compatibles" avec PPP :-)

4 Courts exemples à titre de pense-bête

Cette section ne constitue pas un mode d'emploi, elle contient quelques trucs épars qui peuvent rendre service, sans aller chercher dans la documentation générale.

4.1 Vérifier que le noyau contient bien le pilote PPP

On utilise le système de fichiers `/proc`, très pratique pour ce genre de choses:

```
bash$ cat /proc/net/dev
Inter-|   Receive                  |  Transmit
 face |packets errs drop fifo frame|packets errs drop fifo colls carrier
    lo:      0    0    0    0    0        0    0    0    0     0    0
  ppp0:      0    0    0    0    0        0    0    0    0     0    0
  ppp1:      0    0    0    0    0        0    0    0    0     0    0
  ppp2:      0    0    0    0    0        0    0    0    0     0    0
  ppp3:      0    0    0    0    0        0    0    0    0     0    0
   sl0:      0    0    0    0    0        0    0    0    0     0    0
   sl1:      0    0    0    0    0        0    0    0    0     0    0
   sl2:      0    0    0    0    0        0    0    0    0     0    0
   sl3:      0    0    0    0    0        0    0    0    0     0    0
  eth0:      0    0    0    0    0        0    0    0    0     0    0
```

Si vous avez les quatre lignes `ppp0`, le pilote PPP est bien là. Les lignes `sl0` concernent SLIP et `eth0` l'Ethernet, vous pouvez ne pas en disposer bien sûr.

4.2 Connexion à un serveur PPP via modem.

Pour utiliser PPP, vous invoquez le programme `pppd` avec les options appropriées. Tout ce que vous avez besoin de savoir sur ce programme est contenu dans la page de manuel de `pppd(8)`. Voici un exemple assez général.

La commande ci-après peut être directement tapée sous shell, ou de façon plus pratique faire l'objet d'un petit shell-script.

```
pppd connect 'chat -v "" ATDT5551212 CONNECT "" ogin: ppp word: sesame' \
    /dev/cua1 38400 -detach debug crtscts modem defaultroute 192.1.1.17:
```

Adaptez ces valeurs aux votres bien entendu.

4.3 Vérifier la connexion

Tout d'abord, tapez `/sbin/ifconfig`, vous devez obtenir quelque chose comme ce qui suit, ce qui compte est la partie `ppp0`:

```
lo        Link encap Local Loopback
          inet addr 127.0.0.1  Bcast 127.255.255.255  Mask 255.0.0.0
          UP LOOPBACK RUNNING  MTU 2000  Metric 1
          RX packets 0 errors 0 dropped 0 overrun 0
          TX packets 0 errors 0 dropped 0 overrun 0
```

```
ppp0      Link encap Point-Point Protocol
          inet addr 192.76.32.2  P-t-P 129.67.1.165  Mask 255.255.255.0
          UP POINTOPOINT RUNNING  MTU 1500  Metric 1
          RX packets 33 errors 0 dropped 0 overrun 0
          TX packets 42 errors 0 dropped 0 overrun 0
```

Vous aurez l'état du lien PPP, et les adresses IP établies lors de la négociation. Vérifiez les routes par la commande `route` ou `netstat -r`, et essayez trois ou quatre `ping`.

5 Créer un compte pour recevoir des connexions PPP

Créez un compte nommé par exemple, `ppp`, comme ceci:

```
ppp:(mot de passe crypte):102:50:PPP LOGIN:/tmp:/etc/ppp/ppplogin
```

Le fichier `/etc/ppp/ppplogin` sera un shell-script exécutable contenant par exemple:

```
#!/bin/sh
exec /usr/etc/pppd :192.1.2.23
```

Dans cet exemple, pppd insistera pour que la machine qui appelle se voit attribuer l'adresse IP 192.1.2.23, l'adresse locale pour le serveur sera celle trouvée dans /etc/hosts puisqu'elle n'est pas précisée.

6 Rajouter des canaux PPP supplémentaires

Par défaut, vous disposez de 4 canaux, ce qui permet 4 connexions simultanées (sur 4 modems par exemple). Si vous en désirez plus, vous pouvez très facilement reconfigurer le noyau pour les obtenir. Cela se fait en deux étapes.

Tout d'abord vous devez éditer le fichier `/usr/src/linux/drivers/net/Space.c`. Il contient une section qui ressemble à ceci:

```
#if defined(CONFIG_PPP)
extern int ppp_init(struct device *);
static struct device ppp3_dev = {
    "ppp3", 0x0, 0x0, 0x0, 0x0, 3, 0, 0, 0, 0, NEXT_DEV,  ppp_init, };
static struct device ppp2_dev = {
    "ppp2", 0x0, 0x0, 0x0, 0x0, 2, 0, 0, 0, 0, \&ppp3_dev, ppp_init, };
static struct device ppp1_dev = {
    "ppp1", 0x0, 0x0, 0x0, 0x0, 1, 0, 0, 0, 0, \&ppp2_dev, ppp_init, };
static struct device ppp0_dev = {
    "ppp0", 0x0, 0x0, 0x0, 0x0, 0, 0, 0, 0, 0, \&ppp1_dev, ppp_init, };
#undef NEXT_DEV
#define NEXT_DEV (\&ppp0_dev)
#endif   /* PPP */
```

Cela doit vous sembler évident. Rajoutez les entrées dont vous avez besoin, n'oubliez pas de terminer par `NEXT_DEV` sur la première ligne et ne vous trompez pas dans la numérotation.

Ensuite, vous devez modifier une ligne dans le fichier `include/linux/ppp.h`:

```
#define PPP_NRUNIT     4
```

Mettez la nouvelle valeur. Enfin, recompilez le noyau et réamorcez la machine. Les messages lors du démarrage, ainsi que l'affichage de `/proc/net/dev` devraient refléter vos modifications.