

7PLUS - un encodeur avec capacités de correction étendues.

Introduction

Développé par Axel Bauda, DG1BBQ, et disponible pour divers systèmes d'exploitation (DOS, OS/2, Linux,...) 7PLUS a rapidement supplanté sur le réseau packet tous les autres encodeurs, c-à-d. les utilitaires permettant de transmettre des fichiers binaires (programmes, images,...) sur un réseau dont certains éléments ne sont prévus que pour la transmission de fichiers texte.

Le point fort de 7PLUS est sa gestion des erreurs. Avec la plupart des outils du même genre, il est nécessaire de retransmettre tout le fichier si celui-ci parvient endommagé à son destinataire, ce qui arrive d'ailleurs bien plus fréquemment que le caractère théoriquement sans erreur du protocole AX-25 pourrait le laisser supposer !

En fait, il ne faut pas oublier que l'AX-25 ne contrôle souvent que l'échange de données entre les TNC; la transmission sur le câble série entre le TNC et le PC ne faisant quant à elle l'objet d'aucun contrôle, du moins si on ne travaille pas en hostmode...

En cas d'erreur(s) donc, 7PLUS permet une retransmission limitée aux lignes endommagées et est même capable de corriger de lui-même toute erreur ne concernant qu'un caractère par ligne, qu'il soit manquant, modifié ou en trop.

Utilisation de base

Pour tout ce qui suit, nous supposons que le fichier à transmettre s'appelle **MONPROG.EXE**. Le programme 7PLUS se trouvera quant à lui de préférence dans un répertoire figurant dans le PATH du système afin d'être appelable de partout.

Voyons donc les différentes étapes de l'encodage, du décodage et de la correction des éventuelles erreurs de transmission.

a) Encodage et transmission d'un fichier.

L'encodage est on ne peut plus simple. Il suffit de taper: **7PLUS MONPROG.EXE**

Si le fichier de départ est petit (moins de 8,5 kB environ), un seul fichier nommé MONPROG.7PL sera généré, avec une augmentation de taille due à l'encodage d'environ 13 à 15 %. Un fichier de plus grande taille sera scindé en plusieurs parties de 10 kB portant toutes le nom MONPROG, suivi des extensions P01, P02, etc... correspondant au numéro de la partie en hexadécimal (donc après P09 vient P0A et ce n'est qu'après P0F que vient P10, si la longueur du programme le nécessite bien sûr).

Ce sont ces parties qu'il faudra transmettre, en créant un message pour chacune. Certains programmes packet peuvent gérer cela tous seuls, mais la plupart vous laisseront le soin de gérer cela manuellement.

Donc, pour chaque partie:

- envoyer la commande de création de message, p.ex:
SPHB9XYZ@HB9ZYX
- envoyer le titre, normalement le nom complet du programme de départ suivi du numéro de la partie et du nombre total de parties, par exemple: **MONPROG.EXE P01/05**.
- envoyer la bonne (!) partie.

Il est possible d'agir sur la taille des parties au moyen des options **-SB n** (n = taille des parties en octets) et **-SP m** (m = nombre de parties).

Par exemple:

7PLUS MONPROG.EXE -SB 5000 créera des parties de 5000 octets au lieu des 10000 par défaut,

7PLUS MONPROG.EXE -SP 12 encodera le programme en 12 parties de taille égale (sauf la dernière qui est en général un peu plus petite).

Il convient d'être prudent en utilisant ces options s'il s'agit d'augmenter la taille des parties. 7PLUS admet en effet jusqu'à 36 kB mais il est vivement déconseillé d'utiliser plus que les 10 kB habituels car cela rend le forward des messages beaucoup plus difficile, voire impossible sur certaines portions de réseau encore à 1200 bits/s. Pour le forward par ondes courtes, donc à 300 bits/s, il ne faut en aucun cas dépasser 5 kB par partie, 2 à 3 kB étant préférable, et limiter ce type de diffusion au strict minimum vu le faible débit.

b) Réception et décodage d'un fichier.

Ici aussi, la complexité dépend largement du programme packet utilisé. Les plus performants sauvegardent automatiquement les parties en 7PLUS sous le bon nom au fur et à mesure de leur lecture et il ne reste plus qu'à cliquer sur une icône pour décoder le tout.

De nombreux programmes moins évolués ne disposant toutefois pas de ces fonctions, voici une marche à suivre applicable avec presque n'importe quel programme:

- ouvrir la capture du texte reçu sur disque (le programme doit au moins être capable de sauvegarder sur disque ce qui arrive !), dans un fichier de nom quelconque, disons RECU.TXT.
- lire toutes les parties 7PLUS
- fermer la capture. Une variante consiste à sauvegarder à ce moment seulement le buffer de réception du programme packet mais c'est risqué car s'il est trop petit, le début du programme en 7PLUS ne s'y trouvera déjà plus.
- aller dans le répertoire où se trouve le fichier reçu et taper la commande: **7PLUS -X RECU.TXT**. Ceci extraira d'un coup toutes les parties contenues dans ce fichier, créant donc des fichiers portant le nom choisi par l'expéditeur, MONPROG dans notre exemple, et les extensions correspondant aux numéros des parties.
- décoder ces fichiers 7PLUS par la commande: **7PLUS MONPROG**. Attention: il ne faut pas ajouter d'extension (.7PL ou .P01) au nom de fichier.

Si tout s'est bien passé, vous serez finalement l'heureux propriétaire d'un fichier MONPROG.EXE prêt à être utilisé. Dans le cas contraire, il sera nécessaire de lire aussi le chapitre suivant !

c) Procédure de correction des erreurs.

Si des erreurs non récupérables sont rencontrées, 7PLUS signale que le décodage est infructueux et génère un fichier nommé **MONPROG.ERR**. Ce fichier contient la liste des lignes endommagées dans chaque partie. Tout ce que vous avez à faire est d'envoyer ce fichier à l'expéditeur du programme et d'attendre qu'il vous fasse parvenir en retour un fichier nommé MONPROG.COR contenant les fameuses lignes corrigées. Si vous avez lu un envoi en 7PLUS diffusé sous forme de bulletin, il est fort possible que la correction arrive sous forme de fichier de correction collectif (voir plus bas).

Une fois ce fichier reçu, enregistrez-le sous le nom **MONPROG.COR**, c'est important: si vous changez le nom, 7PLUS ne verra plus qu'il s'agit d'une correction de MONPROG et si vous changez l'extension il ne saura plus que c'est un fichier de correction.

Pour effectuer la correction, retournez dans le répertoire où se trouvent toutes les parties plus la correction et tapez une nouvelle fois: **7PLUS MONPROG**.

Bien qu'il s'agisse de la même commande que lors de la première tentative de décodage, vous remarquerez que 7PLUS n'indique plus le décodage de toutes les parties mais seulement le traitement de MONPROG.COR. Ceci est possible du fait que, en cas de décodage infructueux, 7PLUS conserve le résultat imparfait de ce travail dans un fichier **MONPROG.7MF** qui ressemble déjà beaucoup au programme à décoder, mais a un contenu provisoire aux endroits concernés par les erreurs. Lors du traitement d'une correction, ce fichier est automatiquement utilisé et mis à jour pour donner le produit fini correct: MONPROG.EXE.

d) Génération d'un fichier de corrections.

Voyons maintenant ce qui se passe chez l'expéditeur du programme lorsqu'il reçoit votre fichier d'erreurs. La commande suivante lui permet de générer un fichier de corrections à votre attention:

7PLUS MONPROG.ERR C:\REPERT\.

où C:\REPERT\ («\» final obligatoire !) désigne le répertoire dans lequel se trouvent les parties encodées d'origine, qu'il est donc nécessaire de conserver jusqu'à être sûr que le décodage a réussi. Cette opération génère un fichier MONPROG.COR prêt pour l'envoi.

Et si la correction arrive elle-même endommagée ? Pas de problème: le processus peut être répété à volonté jusqu'à ce que toutes les erreurs aient été corrigées, mais c'est vraiment un cas peu fréquent. Le plus simple est même souvent d'envoyer une seconde fois le même fichier de correction, vu que sa taille est en général très petite.

e) Paramétrage du TNC (mode terminal).

Il est important de réaliser que, contrairement à des utilitaires similaires, par exemple UUENCODE très utilisé sur internet, 7PLUS utilise des codes ASCII supérieurs à 127. Son utilisation ne dispense donc pas de paramétrer votre station packet en 8 bits !

Cela ne pose en général pas de problèmes en hostmode, mais plus souvent en mode terminal (avec YAPP, TPK ou programmes similaires).

Avec une EPROM TAPR, équipant de nombreux TNC-2, les paramètres cruciaux sont: **AWLEN 8** et **8BITCONV ON**.

Il faudra aussi souvent changer le caractère de commutation de voie (paramètre **STREAMSW**) étant donné que le caractère «\» (code ASCII 124) utilisé par défaut dans ces EPROMS est utilisé par 7PLUS pour l'encodage. Une solution est par exemple d'utiliser le code ASCII 1 (Ctrl-A) qui n'a pas d'effets dommageables.

Voici d'ailleurs la liste complète des codes ASCII (en décimal) non utilisés par 7PLUS:

0 à 32, 42, 127, 145 et 147.

Utilisation avancée

f) Génération d'un fichier de correction collectif.

Cette procédure est très utile lorsque le document encodé en 7PLUS est envoyé non pas à un seul destinataire en message personnel, mais à tous sous forme de bulletin. Au cours de la diffusion des messages, les erreurs vont survenir aléatoirement de sorte que votre bulletin sera reçu avec des erreurs différentes selon les régions. Vous allez donc voir arriver des fichiers d'erreurs de contenu varié.

Une méthode serait bien sûr de procéder comme décrit ci-dessus pour chaque fichier d'erreurs. Imaginez toutefois le travail s'il en arrive des dizaines ! Pour cette raison, il est déconseillé de répondre individuellement à chacun. La méthode la plus efficace est l'établissement d'un fichier de correction collectif, prenant en compte toutes les erreurs annoncées. Voici comment procéder:

Au départ, les différents fichiers d'erreurs doivent porter les extensions .ERR pour le premier reçu, puis .E01, .E02, etc...(en hexadécimal). Les programmes modernes enregistrent automatiquement chaque fichier d'erreurs sous le nom approprié au fur et à mesure de son arrivée. Si votre programme ne dispose pas de cette fonction, la procédure devient terriblement fastidieuse car il n'est pas possible de sauvegarder tous les messages d'erreurs dans un seul fichier et de les extraire, comme cela peut se faire pour les parties 7PLUS proprement dites (cf. chapitre b). Il faudrait donc enregistrer manuellement chaque rapport d'erreurs sous le bon nom mais franchement, si vous utilisez régulièrement 7PLUS, mieux vaut adopter un programme disposant des fonctions de traitement automatisé du 7PLUS !

Une fois tous les fichiers d'erreurs enregistrés sous le nom approprié, il ne reste plus qu'à fusionner dans le fichier .ERR le contenu des tous les autres. Ceci se fait avec la commande:

7PLUS -J MONPROG.ERR.

Le fichier de correction MONPROG.COR peut dès lors être généré comme décrit au chapitre précédent. Ce fichier sera ensuite envoyé en bulletin dans la **même rubrique** que les parties d'origine. Les lecteurs ayant envoyé un fichier d'erreurs sont responsables de contrôler régulièrement leur BBS pour voir si une correction est arrivée.

Si l'on veut être (presque) sûr que chacun reçoive la correction, il est possible d'envoyer une copie du fichier de correction collectif aux intéressés, pour autant que leur nombre soit raisonnable. Même dans ce cas, il est nécessaire d'envoyer la correction sous forme de bulletin, pour au moins deux raisons:

- les OM's qui liront votre envoi un peu plus tard pourront souvent utiliser tout de suite votre correction sans avoir besoin de vous envoyer un message d'erreurs et d'attendre votre réponse.
- les systèmes de décodage automatiques installés sur certaines BBS pourront aussi en faire bon usage.

g) Pour en savoir plus.

Le présent article ne couvre pas toutes les possibilités et options de 7PLUS, mais seulement les plus utilisées. Pour tirer un parti maximal de cet utilitaire, il est vivement conseillé de parcourir le fichier de documentation fourni avec chaque diffusion de 7PLUS. L'appel de 7PLUS sans paramètres affiche aussi quelques options supplémentaires qui peuvent être utiles.

Conclusion

7PLUS représente un moyen très pratique de diffuser des programmes, images ou autres données, mais encore faut-il éviter qu'une prolifération incontrôlée de tels bulletins ne sature le réseau packet. Il est donc bon de respecter un certain nombre de règles dans le but de contribuer à conserver un réseau packet non saturé, donc efficace pour tous:

- attention surtout à la taille des programmes: plus de 500 kB environ semble excessif dans à peu près tous les cas. Il convient de ne dépasser les 100 à 200 kB qu'exceptionnellement, si le document semble vraiment de nature à intéresser beaucoup de monde.
- ne pas coder en parties de plus de 10 kB (qui est la valeur par défaut). Les parties de 36 kB (taille maximale autorisée par 7PLUS) posent de gros problèmes de forward dans les régions à links faibles et y sont d'ailleurs souvent purement et simplement rejetées.
- fuir les contenus non radioamateurs, genre utilitaires Windows, drivers d'imprimantes ou de cartes vidéo, jeux, images (sauf schémas en rapport avec notre hobby) et sons par exemple. Ce serait très bien de pouvoir envoyer tout cela, mais d'une part cela est incompatible avec nos conditions de concession et d'autre part notre réseau n'a pas un débit suffisant pour prétendre être un mini-internet !
- **avant** de diffuser les parties codées, envoyez toujours **en clair** et dans la même rubrique un fichier d'explications qui doit permettre à chacun de déterminer s'il vaut la peine pour lui de lire votre envoi. Dans le cas d'un programme, y inclure au moins: l'équipement nécessaire pour l'utiliser (matériel + logiciel), la taille du programme, sa version et une brève description de son utilité.
- ne diffuser en aucun cas de programmes commerciaux, il s'agit d'une violation du copyright ! Les freewares et sharewares peuvent être diffusés pour autant que les conditions de diffusion qui pourraient être fixées par leurs auteurs (ne pas retirer de fichiers par exemple) soient respectées.
- choisir intelligemment la rubrique (éviter TOUS, ALL, etc...) et la zone de diffusion. Cela n'a par exemple aucun sens de diffuser @WW un programme en français.
- s'assurer que ce que vous envisagez de diffuser ne l'a pas été récemment. On voit régulièrement le même programme arriver deux ou même trois fois en moins d'un mois !

Robert CHALMAS - HB9BZA
(mars 1997)

La page LINUX

Voilà une grande nouveauté dans notre journal: je vais vous parler de Linux et comment l'utiliser en packet. L'avantage quand on s'adresse à un club comme le nôtre est de savoir déjà que chacun dispose d'un PC. Donc pour ceux qui ne connaissent pas le système, il faut un PC. Mais ce n'est pas tout, pour installer et utiliser Linux, vous devrez d'abord vous séparer de certaines idées qui font l'informatique d'aujourd'hui, comme Plug and Play (qui souvent, sous Windows, se transforme en Plug and Pray), le mot «INSTALL» et l'installation sans lire la doc.

Donc on installe Linux une fois qu'on a pris connaissance de ce que c'est et qu'on a lu la documentation. Alors voici un petit guide qui vous donnera une idée pour utiliser Linux en packet.

Qu'est-ce que Linux ?

C'est un système d'exploitation Unix parfaitement multi-tâches et multi-utilisateurs, un système Unix qui a été porté sur PC au début des années 90. Ce fut un pari d'étudiants: un noyau Unix sur processeur Intel. A cette époque seul le noyau fonctionnait, sans aucune application. Aujourd'hui Linux se trouve parmi les Unix les plus stables et les plus performants, il a en plus de ses extraordinaires performances l'avantage d'être gratuit.

Pourquoi Linux ?

Linux est une plateforme orientée réseau, comme tout Unix qui se respecte. En fait c'est sa spécialité. Son noyau multi-tâches permet de faire tourner plusieurs applications en même temps, sans blocage. Il est rare qu'un PC sous Linux soit relancé quand une application plante, le noyau la place dans une zone «morte» et libère la quantité de RAM que cette application moribonde utilisait. Ainsi il est envisageable d'utiliser Linux sur un PC enclenché en permanence. Déjà beaucoup de machines sur notre réseau packet fournissent des services comme BBS, nodes ou DX-Cluster sous Linux. Pour l'utilisateur, Linux apporte beaucoup d'avantages également, particulièrement pour celui qui aime personnaliser.

Comment se procurer Linux et comment l'installer ?

Il existe 3 méthodes d'installation, toutes sur la même distribution sous forme de CD-ROM. La distribution la plus répandue est Infomagic sur 6 CD-ROM, dont le prix avoisine les 40 francs suisses. Les 3 méthodes d'installation ont toutes leurs avantages, ainsi la RED HAT propose une installation rapide et très conviviale, très appréciée par les débutants; la SLACKWARE est elle un peu plus compliquée à installer mais offre un meilleur confort d'utilisation, les anciennes versions avaient souvent beaucoup de bugs, corrigés maintenant; enfin reste la DEBIAN qui permet une installation complète, mais reste à la portée des plus chevronnés.

Les CD-ROM peuvent être trouvés en Suisse dans les bonnes librairies techniques.

Comment se documenter ?

Linux offre une importante documentation dans ses CD. Il existe aussi d'autres sources d'information comme des livres et les médias électroniques, tels le packet radio.

Pour les livres je vous recommande de vous procurer «Le Système Linux» de Matt Welsh et Lar Kaufman aux éditions O'Reilly, ISBN 2-84177-002-8 et «Administration Réseau sous Linux» de Olaf Kirch aux mêmes éditions, ISBN 2-84177-007-9. Ces deux ouvrages ont été traduits par René Cougnenc.

Les médias électroniques comme internet ou les BBS packet donnent également toute l'information nécessaire pour débiter. Il existe un guide du ROOTard, disponible en FTP sur internet sur le site ftp.ibp.fr et en français. Ce guide est aussi disponible sur la BBS HB9IAP dans le FBBDOS \LINUX\DOCS.

Et si on parlait de packet ?

Il existe plusieurs softs packet sous Linux. Beaucoup sont faits pour des systèmes comme BBS, DX-Cluster, nodes routeurs TCP/IP, etc. Parmi ceux dédiés à l'utilisateur, je vais m'attarder sur un soft packet qui semble tout simple, mais sacrément efficace et très agréable à utiliser. Il s'agit de TNT.

Tout d'abord, son interface. Il ressemble à SP et ses commandes sont assez semblables. Il tourne sur une console SVGA ou bien sur X-window. On regrette que sur l'interface X-window, le graphisme ne soit pas meilleur. Côté caractéristiques il porte bien son nom, c'est une vraie bombe...

- Utilise un TNC avec eprom TF
- Avec TFKISS on peut utiliser un TNC en mode KISS ou un BAYCOM
- Supporte autobin et yapp
- Gestion des mots de passe pour les sysops en reconnaissant les ssid
- Fonctions serveur donnant accès au shell de Linux ou au packet depuis le shell par telnet donc vous packetez depuis une autre machine si vous êtes relié en réseau local
- Couplage possible avec la BBS DPBox

- Peut servir de frontal pour d'autres programmes comme WAMPES
- Broadcast pacsat et FBB
- Capture et traitement automatique des fichiers 7plus
- Auto-routing, comme SP
- ...etc...

Au moment où j'écris ces lignes, la version actuelle est TNT1.0 et vous pouvez vous la procurer sur internet sur <http://www.snafu.de/~wahlm> ou sur quelques BBS allemandes.

L'installation est assez simple et une documentation en français est disponible dans l'archive. Cette doc est pour une version plus ancienne, mais permet de faire l'installation sans problème.

Installation de TNT:

Récupérer l'archive sur <http://www.snafu.de/~wahlm>. Le fichier à prendre est `tntsrc10.tgz` (version actuelle au moment où j'écris ces lignes). Si vous avez un disque dur IDE, récupérez aussi `hdparm` sur le même site.

Désarchiver le fichier dans le répertoire de votre choix (normalement `/usr/src/tnt/`) et compiler l'application. La procédure est très simplement expliquée dans `README`.

Créer où vous voulez un répertoire TNT (`/work/tnt/` est le répertoire préconfiguré dans les fichiers `ini`). Y créer les répertoires suivants:

```
drwxr-xr-x 2 root root 1024 Nov 9 22:40 abin/
drwxr-xr-x 4 root root 1024 Apr 6 1996 bcast/
drwxr-xr-x 2 root wheel 1024 Dec 23 16:43 bin/
drwxr-xr-x 2 root root 1024 Apr 6 1996 cookie/
drwxr-xr-x 2 root root 1024 Feb 6 01:38 ctext/
drwxr-xr-x 2 root root 1024 Apr 21 1996 doc/
drwxr-xr-x 2 root root 1024 Oct 22 07:57 exemples/
drwxr-xr-x 2 root root 1024 Apr 6 1996 macro/
drwxr-xr-x 2 root root 1024 Feb 6 00:05 newmail/
drwxr-xr-x 2 guest ham 1024 Feb 7 17:02 rem/
drwxr-xr-x 2 root root 1024 Aug 17 05:00 samples/
srwxrwxrwx 1 root root 0 Feb 8 08:11 socket=
drwxr-xr-x 6 root root 2048 Apr 6 1996 sources/
drwxr-xr-x 5 guest ham 1024 Jun 18 1996 users/
drwxr-xr-x 2 root wheel 1024 Feb 6 01:40 yapp/
```

Copier dans le répertoire de TNT tous les exécutables et les fichiers se terminant par `.sample`, les éditer en mettant vos paramètres personnels, vérifier si les chemins d'accès sont corrects et renommer sans le `.sample`.

Copier la doc des sources dans `/doc`.

A partir de là, vous devriez pouvoir faire du packet et tenter la première connexion.

Si vous utilisez un TNC sans eeprom TF, vous devrez le passer en KISS et utiliser `tfkiss` que vous pouvez vous procurer sur le même site que TNT.

En cas de problème ?

En principe TNT s'installe sans trop de problèmes. Si ce n'était pas le cas, relisez la documentation. Si ce n'est pas suffisant, vous pouvez alors expliquer votre problème dans la rubrique `LINUX@FRANCA`. Il y a aussi une mail-list consacrée à TNT sur `DB0GR`. Pour s'inscrire, envoyer un message à `LSTSRV@DB0GR`, dans le sujet mettez `SUBSCRIBE DP8TNT`. Vous serez averti de votre inscription. Au cas où cela ne devrait pas fonctionner, envoyer un message à `DL4YBG@DB0GR`. La plupart des messages sont en allemand ou en anglais. TNT dispose aussi d'une documentation en français, complète à 95 % car elle a été écrite pour une version plus ancienne. Les notes de mise à jour en anglais informent des changements lors d'une nouvelle version.

T'as le look coco... ça ressemble un peu à SP, non ?

rm

Ch:01 Stat: HB9IAP-8 My: HB9HLI AxSt:00|00|00

Il y a 1 message(s) nouveau(x). RN pour le(s) lire.

```
1:HB9IAP >
rn

Expediteur   : DG5OJ
Destinataire: HB9HLI
Type/status  : PN
Date/heure   : 28-Fev 14:01
Bid          : E6BAWC_DB0GR
Message #    : 165060
Titre        : (DP8TNT 207) tnt-interface?
Path: !HB9PD!HB9AJ!DB0EAM!DB0NHM!DK0MNL!DB0GR!
From: DG5OJ @ DB0GR.#BLN.DEU.EU (Michael)
To: HB9HLI @ HB9IAP.SROM.CHE.EU
Reply-To: DG5OJ @ DB0UHI.#NDS.DEU.EU

** CONNECT *** CHANNEL 01 *****
***RMNC/FlexNet V3.3f
```

Il existe aussi la possibilité de lancer TNT dans une fenêtre X-window, pour cela il faut utiliser xtnt.
Les possesseurs de modems baycom peuvent aussi utiliser TNT, mais doivent configurer le noyau et différents fichiers.
Comme il y a peu de place, on en reparlera la prochaine fois. En attendant, si vous avez accès à internet, allez voir sur <http://www.ife.ee.ethz.ch/~sailer>.

A suivre...

Powered by Applixware/Linux.
Linux 2.0.29 (Posix) 99.99% working on linux,no more dollars for Bill!
Email : hb9hli@bluewin.ch or muller@eicn.ch
AX25mail : HB9HLI@HB9IAP.SROM.CHE.EU
Snail-mail: PO Box 133 ch-2400 Le Locle - Switzerland
Web : <http://www.eicn.etna.ch/~muller>

Packet et linux 2ème partie - mars 1998

Plutôt que vous donner la recette miracle pour installer des logiciels packet sous linux, je vous propose cette année un survol des possibilités actuelles pour votre activité packet sous linux. En effet, l'année passée je vous promettais de parler de l'installation pour un modem baycom. Seulement voilà, linux évolue pour plus de fiabilité ce qui fait que, pour être actuel, il n'est plus possible de préparer des articles à l'avance, sous peine de dépasser le «Migros-data» et vous proposer une solution obsolète. Je vous propose cette année un survol de l'actualité du packet- radio sous linux.

1. TNT et DPBOX n'ont plus évolué depuis quelque temps. Les dernières versions sont parfaitement stables et n'ont plus de bugs. Les archives de ces programmes sont disponibles sur <http://www.snafu.de/~wahlm> ou en packet sur DB0LJ-8 (fileserver).

2. LINUXNET, le célèbre node du Swiss-artg n'est pas forcément que pour monter un digipeater, c'est une solution simple pour diriger plusieurs programmes packet sur le même TNC. La version actuelle est la 1.12 et l'archive se trouve sur <http://www.swiss-artg.ch/xnet> ou en packet sur DB0LJ-8.

3. NOYAU AX25. Il suit de très près le développement du noyau de linux, votre système packet devra être paramétré lors de la recompilation du noyau. Cette mouture est disponible avec les noyaux en développement et le sera bien évidemment dès l'arrivée des noyaux stables 2.2.x. Les interfaces sont nombreuses, en voici quelques unes:

KISS, 6PACK pour TNC2, BAYCOM 1K2 et 9K6 sur port série et parallèle, SB16 (packet avec la carte son) pour 1K2 et 9K6, ... et encore beaucoup d'autres.

Le grand atout du noyau ax25, c'est le tcp/ip et son utilisation permet les mêmes fonctions que sur internet. Une fois le noyau ax25 configuré, la couche tcp/ip est la plus simple à obtenir. L'interface WWW est donc naturellement possible. Les archives du noyau ax25 se trouvent sur <http://zone.pspt.fi/~jsn/> ou en packet sur DB0LJ-8.

4. EXEMPLE DE CONFIGURATION:

Voici une description de ma station packet sous linux qui se trouve dans un seul et unique PC, un 486 DX 4 100 MHz:

```
XNET 1.12 5 ports:      port 1: KISS TNC2
                        port 2: KISS TNT et DPBOX
                        port 3: Accès TCP/IP (noyau ax25)
                        port 3: DXNET (version linux via le noyau ax25)
                        port 4: AXIP (pour link sur un autre pc-réseau)
                        port 5: KISS PC/FlexNet sur un 286 pour test.
```

Ainsi nous avons sur la même machine un terminal, une BBS, un DX-Cluster et un accès TCP/IP avec serveur web apache et toutes les autres fonctions d'internet.

Comment se tenir au courant ?

1. Sur la BBS HB9IAP-8, lire les bulletins LINUX et UNIX pour les questions générales, XNET pour le node XNET, DPTNT pour TNT et DPBOX.

2. Sur internet, il y a quelques mail-lists, pour la plupart traitant un domaine général, comme frpacket (listproc@ham.ireste.fr, mettre HELP dans le corps du message, sans titre). Linux-ham comme son nom l'indique traite toutes sortes de logiciels sous linux pour le monde radioamateur (Majordomo@vger.rutgers.edu, mettre subscribe linux-ham dans le corps du message sans titre).

Je vous souhaite un bon packet sous linux.

Dominique HB9HLI
Christophe F1RPC

La page Linux

Ces quelques mots s'adressent essentiellement aux utilisateurs du packet radio sous linux. Le problème en ce moment c'est que linux se trouve dans une période charnière entre 2 librairies système, ce qui rend les choses un peu problématiques. Pour faire du packet, je vous présente une nouvelle version de TNT. Cette version a plusieurs avantages, celui de fonctionner tant en libc5 qu'en libc6 et d'être très simple à installer.

Quelques caractéristiques de tnt:

- Support transfert de fichier en yapp et autobin.
- Passerelle réseau/ax25.
- Front-end pour la DpBox.
- Stockage des passwords pour sysop, différenciés selon les ssid.
- Utilisation simple car ressemble à d'autres terminaux packet comme SP et TOP.
- Support broadcast pacsat.
- Compression Huffman
- Configurable à souhait, etc.

J'ai testé l'installation de la version 1.6, et cela s'est parfaitement déroulé sur 3 machines différentes. La version 1.7 est la dernière version au moment où j'écris ces lignes, mais le script d'installation ne fonctionne pas comme il faut. Il se peut que la version 1.8 sorte et soit meilleure, sinon rabattez-vous sur la 1.6.

Pour vous procurer TNT, il suffit d'aller chercher l'archive sur internet:

<http://excelsior.kullen.rwth-aachen.de/wsp/packet.php3> ou

<http://www.bfl.at/mayer/ham/>

Si vous n'avez pas internet, j'ai mis tnt1.6 dans le fbbdos de la BBS HB9IAP:

`\LINUX\TNT\TNT1_6.TGZ`

Choisissez un moment calme pour le récupérer, car l'archive fait plus de 400 Ko.

L'installation est ultra simple, la marche à suivre est clairement indiquée dans la documentation. Vous avez le choix de l'installer à un endroit particulier de l'arborescence, sinon laissez les chemins par défaut. Lors de l'installation, vous devrez entrer quelques paramètres personnels, comme votre indicatif ce qui, après installation devrait fonctionner directement si vous utilisez un TNC2 avec eprom hostmode.

Pour le fonctionnement avec un modem baycom, vous devrez utiliser le noyau ax25 et les ax25-utilities. Je vous propose une solution qui fonctionne parfaitement avec une SUSE 6.0 sans devoir recompiler le noyau, car dans la SUSE par défaut, tout est compilé en modules. Vous devrez cependant récupérer les ax25-utils-2.0.42a.rpm que vous installerez avec Yast. Récupérez également tkiss sur <http://www.snafu.de/~wahlm> et placez le binaire tkiss et les fichiers de config dans le répertoire de tnt.

Editez `tnt.ini` et modifiez-le ainsi:

```
# defines if select() shall be used, normally 1,
# put to 0, if you are using old kernels
# (select() produces there a higher CPU-load).
use_select 1

# -----

# 0 for real TNC at serial port, 1 for TFKISS on a
# UNIX-socket, 2 for TFKISS on other socket
soft_tnc 1

# serial port to which TNC is connected, UNIX-socket
# of TFKISS or other socket for TFKISS
device /ham/tnt/socket

# lockfile for serial port or TFKISS
tnt_lockfile /var/lock/LCK..tnt
```

```
# baudrate used, only used if TNC at serial port
# speed 9600
-----cut-----
```

Editez `tfkiss.ini` et modifiez-le ainsi:

```
# enable (1) or disable (0) kiss port
kiss_active 1
# enable (1) or disable (0) axip port
axip_active 0
# device for kiss port
device /dev/ttys1
# lockfile for device used for kiss port
tfkiss_lockfile /usr/spool/uucp/LCK..ttys1
# speed on device for kiss port
speed 9600
-----cut-----
```

Voici le script qui va réveiller votre bon vieux baycom (serial 1k2 dans ce cas):

```
/usr/local/bin/tnt-baycom:
#! /bin/bash
#
echo Désactiver l'uart du COM 1
setserial /dev/ttyS0 uart none
echo load ax25 et baycom modules...
sleep 1
insmod ax25
sleep 1
insmod mkiss
sleep 1
insmod hdlcdrv
sleep 1
insmod baycom mode="ser12*" iobase=0x3f8 irq=4
sleep 1
sethdlc -p -i bc0 mode "ser12*" io 0x3f8 irq 4
echo configuration reseau
sleep 1
/sbin/ifconfig bc0 hw ax25 HB9HLI 44.142.39.100 up
route add -net 44.0.0.0 bc0
echo configuration du transcodeur reseau - kiss
/usr/sbin/net2kiss -i bc0 /dev/ptys1 &
echo monitoring des trames ax25
sleep 7
listen -cart8 </dev/tty8 >/dev/tty8 &
echo OK.
sleep 3
echo start DPTNT...
# tnt
cd /ham/tnt
./tfkiss
./tnt -d
echo OK.
sleep 2
./tntc
```

Voilà, normalement vous devriez y arriver. Ces scripts ne sont que des indications, car je les ai pris en partie, ayant une configuration très différente, mais sont basés sur ce principe.

Voici ce que donne tnt dans un xterm.

```

-- TNTC
Ok
:xc hb9hli-11
Ok
c hb9hli
Ch:01 Stat: IXF wid Dom:HB9HLI           My: HB9HLI-15 AxSt:00|00|00
-----
                AX25 et TCP/IP routeur
                -----
                Connecte a tux.hb9hli.ch
                Pour connecter mon terminal, taper [ope].

=>c hb9hli

link setup (1)...
*** connected to HB9HLI
Linu(X)net version 1.16

                AX25 Packet Radio et TCP/IP routeur
                -----
                Connecte a hb9hli.ampr.org (44.142.39.1)

                ----> Pour parler avec moi (chat) taper [ope] ou [tnt]
                ----> Pour m'envoyer un message taper [m]
                ----> Information aux visiteurs de ce noeud [i]
                ----> Services utilisables, taper [services]
                Vous etes sur un systeme unix, respecter les minuscules.

                Ma page web a ete completement revue: http://hb9hli.cjb.net

a,c,conv,d,h,i,l,linux,m,map,n,ope,q,services,u,v
=>
*C*I01**** 0:=== hb9hli 2:=== 3:=== 4:=== 5:=== 6:=== 7:=== 8:=== 9:===*
fm HB9HLI-15 to HB9HLI-11 ct1 RR6v - 27.03.99 19:44:48

```

Bon packet sous linux.

Dominique HB9HLI

Email :hb9hli@bluewin.ch or muller@eicn.ch ICQ#: 16107255
 AX25mail :HB9HLI@HB9IAP.SROM.CHE.EU Node: HB9IAC-11
 Snail-mail: PO Box 133 ch-2400 Le Locle - Switzerland
 Web :http://hb9hli.cjb.net (hamradio and linux)
 Powered by LinuX 2.0