

Configuration de la connexion PPP

Olivier Hoarau (olivier.hoarau@funix.org)

V1.3 du 15.07.02

1	Historique	3
2	Préambule	3
3	Prérequis général	3
4	Configuration PPP à la main	3
4.1	Présentation	3
4.2	Configuration	4
4.2.1	Configuration générale	4
4.2.2	Configuration PPP avec CHAP (Free)	6
4.2.3	Configuration des serveurs DNS	7
4.3	Comment ça marche	7
4.4	Configuration avancée	9
4.4.1	Utilisateurs PPP	9
4.4.2	Estimation du coût des communications	10
4.4.3	Connaître la vitesse de connexion	11
5	Configuration PPP façon clique-bouton	12
5.1	Présentation	12
5.2	Configuration avec drakconf	12
5.3	kppp	14
6	Configuration PPP (cas de plusieurs providers)	16
6.1	Présentation	16
6.2	Mise en place des scripts	17
6.3	Estimer le coût de ses communications	23
7	Lancer la connexion d'un poste client	27
7.1	Présentation	27
7.2	Masqodialer	27
7.2.1	Présentation	27
7.2.2	Installation de Masqodialer	27
7.2.3	Configuration de Masqodialer	30
7.2.4	Lancement de Masqodialer	35
7.3	Diald	35
7.3.1	Présentation	35
7.3.2	Installation	36
7.3.3	Configuration	36
7.3.4	Premiers essais	39

7.3.5	Configuration avancée.....	41
7.3.6	Lancement automatique de diald.....	41
7.4	pppd.....	42
7.4.1	Présentation	42
7.4.2	Configuration	43
7.4.3	Lancement automatique de pppd.....	44
7.5	Configuration des postes clients.....	44
7.5.1	Configuration générale	44
7.5.2	Dans le cas de masqdiaper	45
7.5.3	Dans le cas de pppd ou diald.....	49

1 Historique

V1.3 15.07.02 Grosses mises à jour, attribution automatique des serveurs DNS, modifications dans les scripts de connexion et de configuration, passage à diald 1.0, présentation de nouveaux clients pour Masqdiabler, suppression de netcfg et usernet, présentation de drakconf

V1.2 3.12.00 Rajout de la partie « connexion ppp quand on est abonné à plusieurs FAI »

2 Préambule

Ce document a pour but de présenter la configuration d'une connexion PPP sous linux.

La dernière version de ce document est téléchargeable à l'URL <http://www.funix.org>. Ce document peut être reproduit et distribué librement dès lors qu'il n'est pas modifié et qu'il soit toujours fait mention de son origine et de son auteur, si vous avez l'intention de le modifier ou d'y apporter des rajouts, contactez l'auteur pour en faire profiter tout le monde. Ce document ne peut pas être utilisé dans un but commercial sans le consentement de son auteur. Ce document vous est fourni "dans l'état" sans aucune garantie de toute sorte, l'auteur ne saurait être tenu responsable des quelconques misères qui pourraient vous arriver lors des manipulations décrites dans ce document.

3 Prérequis général

Vous avez d'abord besoin des éléments suivants:

- Le numéro de téléphone de votre fournisseur d'accès
- Votre nom de compte chez votre fournisseur et votre mot de passe
- L'adresse IP du serveur DNS primaire de votre fournisseur (éventuellement)
- L'adresse IP du serveur DNS secondaire de votre fournisseur (éventuellement)

4 Configuration PPP à la main

4.1 Présentation

Vous avez la possibilité de configurer la connexion au travers d'outils de configuration du style clique-bouton, cependant si vous voulez vous connecter "à la main" ou du moins comprendre le mécanisme de connexion, ce paragraphe est fait pour vous.

Cette configuration est valable dans le cas où vous avez un provider unique, dans le cas où vous avez plusieurs providers (essais de FAI gratuits), reportez vous au [configuration PPP avec plusieurs FAI](#). Je trouve la configuration de la connexion PPP sur la Mandrake particulièrement confuse, il y a des fichiers sous **/usr/bin** et une tonne sous **/etc/sysconfig/network-scripts**, je n'ai jamais vraiment réussi à comprendre le mécanisme d'une connexion PPP. C'est pourquoi j'ai essayé de construire une solution qui se veut la plus portable possible et donc doit marcher sur toutes les distribs de linux. Cette page comprend l'identification CHAP telle que l'utilise Free.

Dans la suite des opérations **kervao.fr** est le nom du domaine local.

ATTENTION:

- Pour info pour des versions de package de **ppp** inférieures à **2.4.0**, on m'a fait remarquer que mes scripts pouvaient ne pas marcher, je vous conseille donc d'upgrader **ppp**, ce qui a pour effet de résoudre le problème.

- Attention si vous avez une carte réseau, veillez à ce que dans le fichier `/etc/sysconfig/network` vous n'avez pas **GATEWAYDEV=eth0**, voici le contenu du mien:

```
NETWORKING=yes
FORWARD_IPV4=true
HOSTNAME=asterix.kervao.fr
DOMAINNAME=kervao.fr
GATEWAYDEV=
```

4.2 Configuration

4.2.1 Configuration générale

Quand on installe le package **ppp**, celui-ci crée un répertoire `/etc/ppp`, on y trouve les fichiers suivants:

```
-rw----- 1 root daemon 78 jui 19 2001 chap-secrets
-rwxr-xr-x 1 root root 513 sep 17 2001 ip-down*
-rwxr-xr-x 1 root root 2839 jui 24 2001 ip-down.ipv6to4*
-rwxr-xr-x 1 root root 956 sep 17 2001 ip-up*
-rwxr-xr-x 1 root root 4845 jui 24 2001 ip-up.ipv6to4*
-rw-r--r-- 1 root daemon 36 jui 19 2001 options
-rw----- 1 root daemon 102 jun 13 18:43 pap-secrets
drwxr-xr-x 2 root daemon 1024 jui 19 2001 peers/
```

On va créer des scripts personnalisés de connexion, le premier est **ppp-on**, cet exécutable permet de lancer une connexion, voici son contenu:

```
#!/bin/bash

# définition du nom du fichier contenant les paramètres de connexion
DIALER_SCRIPT=/etc/ppp/ppp-on-dialer

#définition du port sur lequel est branché le modem
MODEM=/dev/ttyS0

#définition de la vitesse max du modem (on peut mettre 115200 même si le modem
plafonne à 56400)
SPEED=115200

#options diverses PPP, pour plus d'info "man pppd"
PPPOPTIONS="lock noauth modem crtsets asyncmap 00000000 defaultroute debug"

exec /usr/sbin/pppd -detach $PPPOPTIONS $MODEM $SPEED \
    remotename $DEVICE ipparam $DEVICE \
    connect "/usr/sbin/chat -v -f $DIALER_SCRIPT"
```

Les options que je passe à **pppd** sont celles qui sont définies par défaut sur une distrib Mandrake, je n'ai pas cherché à les comprendre et encore moins à les modifier, retenez quand même que **debug** sert à avoir plus d'informations et que sans **defaultroute** le routage ne marchera pas.

Donnez les droits d'exécution à ce script

```
chmod 755 /etc/ppp/ppp-on
```

Le script exécutable **ppp-off** va permettre de stopper la connexion:

```
#!/bin/bash
if [ "$1" = "" ]; then
    DEVICE=ppp0
else
    DEVICE=$1
fi

if [ -r /var/run/$DEVICE.pid ]; then
    kill -INT `cat /var/run/$DEVICE.pid`

    if [ ! "$?" = "0" ]; then
        rm -f /var/run/$DEVICE.pid
        echo "Argh erreur en détruisant le fichier pid"
        exit 1
    fi

    echo "Liaison PPP $DEVICE terminé"
    exit 0
fi

echo "Erreur liaison PPP non active sur $DEVICE"
exit 1
```

Donnez les droits d'exécution à ce script

```
chmod 755 /etc/ppp/ppp-off
```

Le fichier **ppp-on-dialer** contient les paramètres de connexion

```
'ABORT' 'BUSY'
'ABORT' 'ERROR'
'ABORT' 'NO CARRIER'
'ABORT' 'NO DIALTONE'
'ABORT' 'Invalid Login'
'ABORT' 'Login incorrect'
" 'ATZ'
'OK' 'ATDT0491294697'
'CONNECT' "
'ogin:' 'le-login-chez-le-fai'
'ord:' 'le-mot-de-passe'
'TIMEOUT' '5'
'~_' "
```

Vous remplacez évidemment le numéro de téléphone, le login et le mot de passe par les vôtres. Donnez des droits de lecture uniquement à root.

chmod 600 /etc/ppp/ppp-on-dialer

Dans le fichier **ip-up** vous pouvez mettre ce que voulez comme par exemple l'envoi du courrier en attente de même que la réception

```
#!/bin/bash
/usr/sbin/sendmail -q -v
/usr/bin/fetchmail
```

ATTENTION:

- mettre le chemin complet des commandes
- - c'est avec les droits root que vous allez lancer le script **ip-up**, par conséquent veiller à ce que root ait un **.fetchmailrc** pour relever le mail de tous vos utilisateurs.

De même dans le fichier **ip-down**, vous mettrez les commandes que vous souhaitez lancer à l'arrêt d'une connexion.

4.2.2 Configuration PPP avec CHAP (Free)

La configuration est identique à celle décrite plus [haut](#), les fichiers diffèrent un peu à commencer par **ppp-on**:

```
#!/bin/bash

# définition du nom du fichier contenant les paramètres de connexion
DIALER_SCRIPT=/etc/ppp/ppp-on-dialer

#définition du port sur lequel est branché le modem
MODEM=/dev/ttyS0

#définition de la vitesse max du modem (on peut mettre 115200 même si le modem
plafonne à 56400)
SPEED=115200

#options diverses PPP, pour plus d'info "man pppd"
PPPOPTIONS=" noauth defaultroute modem debug ipcp-accept-remote ipcp-accept-
local idle 180"

exec /usr/sbin/pppd $MODEM $SPEED $PPPOPTIONS \
    connect "/usr/sbin/chat -v -f $DIALER_SCRIPT" \
    remotename proxad user login-chez-free
```

Le fichier **/etc/ppp/ppp-on-dialer** est plus court:

```
'ABORT' 'BUSY'
'ABORT' 'ERROR'
'ABORT' 'NO CARRIER'
'ABORT' 'NO DIALTONE'
'' 'ATZ'
```

```
'OK' 'ATDT0491294697'  
'CONNECT' ''
```

Un fichier fait son apparition `/etc/ppp/chap-secrets`, il contient

```
login-chez-free * mot-de-passe-chez-free
```

Ce fichier doit avoir les droits 600 pour des raisons évidentes de sécurité.

4.2.3 Configuration des serveurs DNS

Vous éditez le fichier `/etc/resolv.conf` et vous indiquez le nom du serveur primaire et secondaire après le mot clé `nameserver`:

```
search kervao.fr  
nameserver 194.149.160.9  
nameserver 194.149.160.1
```

A noter qu'il n'est absolument pas nécessaire de spécifier le même nom de domaine que celui de votre provider, vous pouvez garder votre nom farfelu.

En cas d'attribution automatique des serveurs de noms, le fichier `/etc/resolv.conf` sera automatiquement modifié en conséquence sur une Mandrake par un jeu de fichiers qui m'échappe. Si c'est votre cas, pas de problème, sinon voilà un petit script qui permettra de renseigner automatiquement le fichier `/etc/resolv.conf` il suffit de le rajouter au début de `/etc/ppp/ip-up`

```
rm -f /etc/resolv.conf  
echo "search kervao.fr" /etc/resolv.conf  
# en cas de serveur DNS local, décommentez la ligne suivante  
# echo "nameserver 127.0.0.1"  
rm -f /etc/resolv.conf  
echo "nameserver " $DNS1 /etc/resolv.conf  
echo "nameserver " $DNS2 /etc/resolv.conf
```

Par ailleurs vous devez rajouter l'option `usepeerdns` à `pppd` dans le fichier `ppp-on` comme ceci

```
PPPOPTIONS="usepeerdns noauth defaultroute modem debug ipcp-accept-remote  
ipcp-accept-local idle 180"
```

Pour info c'est `pppd` qui passe les adresses IP à `ip-up` à travers les variables `DNS1` et `DNS2` grâce à l'option `usepeerdns` (`man pppd` pour plus de détails).

4.3 Comment ça marche

Pour lancer la connexion, il suffit de taper `ppp-on` (en tâche de fond pour pouvoir voir les commentaires) et `ppp-off` pour arrêter. Voici le genre de commentaires que vous pouvez voir:

```
[root@tavel /]# ./ppp-on  
Serial connection established.  
using channel 3  
Using interface ppp0
```

```

Connect: ppp0 <-- /dev/ttyS0
sent [LCP ConfReq id=0x1 <asynctest 0x0 <magic 0x81624e37 <pcomp <accomp]
rcvd [LCP ConfNak id=0x1 <asynctest 0xffffffff]
sent [LCP ConfReq id=0x2 <magic 0x81624e37 <pcomp <accomp]
rcvd [LCP ConfAck id=0x2 <magic 0x81624e37 <pcomp <accomp]
rcvd [LCP ConfReq id=0x2 <mru 1500 <asynctest 0xffffffff <magic 0xc2d873ec
<pcomp <accomp <mrru 1500 <endpoint [null]]
sent [LCP ConfRej id=0x2 <mrru 1500]
rcvd [LCP ConfReq id=0x3 <mru 1500 <asynctest 0xffffffff <magic 0xc2d873ec
<pcomp <accomp]
sent [LCP ConfAck id=0x3 <mru 1500 <asynctest 0xffffffff <magic 0xc2d873ec <pcomp
<accomp]
sent [IPCP ConfReq id=0x1 <addr 0.0.0.0 <compress VJ 0f 01 <ms-dns1 0.0.0.0 <ms-
dns3 0.0.0.0]
sent [CCP ConfReq id=0x1 <deflate 15 <deflate(old#) 15 <bsd v1 15]
rcvd [IPCP ConfReq id=0x4 <compress VJ 0f 01 <addr 202.3.224.35]
sent [IPCP ConfAck id=0x4 <compress VJ 0f 01 <addr 202.3.224.35]
rcvd [IPCP ConfNak id=0x1 <addr 202.3.231.33 <ms-dns1 202.3.225.10 <ms-dns3
202.3.225.20]
sent [IPCP ConfReq id=0x2 <addr 202.3.231.33 <compress VJ 0f 01 <ms-dns1
202.3.225.10 <ms-dns3 202.3.225.20]
rcvd [LCP ProtRej id=0x5 80 fd 01 01 00 0f 1a 04 78 00 18 04 78 00 15 03 2f]
rcvd [IPCP ConfAck id=0x2 <addr 202.3.231.33 <compress VJ 0f 01 <ms-dns1
202.3.225.10 <ms-dns3 202.3.225.20]
local IP address 202.3.231.33
remote IP address 202.3.224.35
primary DNS address 202.3.225.10
secondary DNS address 202.3.225.20
Script /etc/ppp/ip-up started (pid 2438)
Script /etc/ppp/ip-up finished (pid 2438), status = 0x0

```

Pour avoir un max d'info, avant de lancer **ppp-on**, ouvrir un shell en tant que root et taper **tail -f /var/log/messages**, vous devriez y voir ça:

```

Jul 9 08:28:19 asterix pppd[2377]: pppd 2.4.1 started by root, uid 0
Jul 9 08:28:20 asterix chat[2391]: report (CONNECT)
Jul 9 08:28:20 asterix chat[2391]: abort on (BUSY)
Jul 9 08:28:20 asterix chat[2391]: abort on (ERROR)
Jul 9 08:28:20 asterix chat[2391]: abort on (NO CARRIER)
Jul 9 08:28:20 asterix chat[2391]: abort on (NO DIALTONE)
Jul 9 08:28:20 asterix chat[2391]: abort on (Invalid Login)
Jul 9 08:28:20 asterix chat[2391]: abort on (Login incorrect)
Jul 9 08:28:20 asterix chat[2391]: send (ATZ^M)
Jul 9 08:28:20 asterix chat[2391]: expect (OK)
Jul 9 08:28:21 asterix chat[2391]: ATZ^M^M
Jul 9 08:28:21 asterix chat[2391]: OK
Jul 9 08:28:21 asterix chat[2391]: -- got it
Jul 9 08:28:21 asterix chat[2391]: send (ATDT3640^M)
Jul 9 08:28:21 asterix chat[2391]: expect (CONNECT)
Jul 9 08:28:21 asterix chat[2391]: ^M

```

```

Jul 9 08:28:40 asterix chat[2391]: ATDT3640^M^M
Jul 9 08:28:40 asterix chat[2391]: CONNECT
Jul 9 08:28:40 asterix chat[2391]: -- got it
Jul 9 08:28:40 asterix chat[2391]: send (^M)
Jul 9 08:28:40 asterix chat[2391]: expect (ogin:)
Jul 9 08:28:40 asterix chat[2391]: 36000/ARQ/V90/LAPM/V42BIS^M
Jul 9 08:28:42 asterix chat[2391]: ^M
Jul 9 08:28:42 asterix chat[2391]: Welcome to 3Com Total Control HiPer ARC
(TM)^M
Jul 9 08:28:42 asterix chat[2391]: Networks That Go The Distance (TM)^M
Jul 9 08:28:42 asterix chat[2391]: ^M
Jul 9 08:28:42 asterix chat[2391]: login:
Jul 9 08:28:42 asterix chat[2391]: -- got it
Jul 9 08:28:42 asterix chat[2391]: send (login-fai^M)
Jul 9 08:28:42 asterix chat[2391]: expect (ord:)
Jul 9 08:28:42 asterix chat[2391]: login-fai^M
Jul 9 08:28:42 asterix chat[2391]: Password:
Jul 9 08:28:42 asterix chat[2391]: -- got it
Jul 9 08:28:42 asterix chat[2391]: send (mot-de-passe^M)
Jul 9 08:28:42 asterix chat[2391]: timeout set to 5 seconds
Jul 9 08:28:42 asterix chat[2391]: expect (~)
Jul 9 08:28:42 asterix chat[2391]: ^M
Jul 9 08:28:42 asterix chat[2391]: ~
Jul 9 08:28:42 asterix chat[2391]: -- got it
Jul 9 08:28:42 asterix chat[2391]: send (^M)
Jul 9 08:28:43 asterix pppd[2377]: Serial connection established.
Jul 9 08:28:43 asterix pppd[2377]: Using interface ppp0
Jul 9 08:28:43 asterix pppd[2377]: Connect: ppp0 <-- /dev/ttyS0
Jul 9 08:28:46 asterix pppd[2377]: local IP address 202.3.231.33
Jul 9 08:28:46 asterix pppd[2377]: remote IP address 202.3.224.35
Jul 9 08:28:46 asterix pppd[2377]: primary DNS address 202.3.225.10
Jul 9 08:28:46 asterix pppd[2377]: secondary DNS address 202.3.225.20

```

Cette sortie est un peu différente dans le cas d'une connexion avec identification CHAP (Free).

A noter que le mot de passe de connexion PPP ainsi que le login apparaît en clair dans le fichier de log.

4.4 Configuration avancée

4.4.1 Utilisateurs PPP

Maintenant il faut faire en sorte que la connexion puisse être lancée par d'autres utilisateurs ne jouissant pas forcément de droits super utilisateur. Pour cela on va se servir de **sudo**, voici le fichier de configuration de **sudo /etc/sudoers**:

```

# sudoers file.
#
# This file MUST be edited with the 'visudo' command as root.

```

```
#
# See the man page for the details on how to write a sudoers file.
#
```

```
# Host alias specification
```

```
# User alias specification
```

```
# Cmnd alias specification
```

```
Cmnd_Alias PPPON=/etc/ppp/ppp-on
Cmnd_Alias PPPOFF=/etc/ppp/ppp-off
```

```
# User privilege specification
```

```
root ALL=(ALL) ALL
```

```
olivier ALL=NOPASSWD:PPPON,NOPASSWD:PPPOFF
```

```
veronique ALL=NOPASSWD:PPPON,NOPASSWD:PPPOFF
```

Mes deux utilisateurs **olivier** et **veronique** ont le droit d'utiliser les commandes **ppp-on** et **ppp-off** et donc de déclencher et d'arrêter une connexion PPP. Pour déclencher une connexion, il suffira donc de taper:

```
sudo /etc/ppp/ppp-on
```

Et pour l'arrêter:

```
sudo /etc/ppp/ppp-off
```

NOTE: Pour simplifier rajouter dans le **.bashrc** de vos utilisateurs habilités **sudo**

```
alias connecte='sudo /etc/ppp/ppp-on'
```

```
alias deconnecte='sudo /etc/ppp/ppp-off'
```

Pour prendre en compte les modifs

```
source ~/.bashrc
```

Il suffit alors de taper dans un shell

```
connecte
```

ou

```
deconnecte
```

4.4.2 Estimation du coût des communications

Si vous voulez connaître la vitesse de connexion, on va légèrement modifier le script **ppp-on** au niveau de :

```
#####
##### Initialisation du fichier /etc/ppp/ppp-on-dialer
#####
cat <<END_OF-DATA /etc/ppp/ppp-on-dialer
'ABORT' 'BUSY'
```

On va rajouter

```
cat <<END_OF-DATA /etc/ppp/ppp-on-dialer
'REPORT' 'CONNECT'
```

'ABORT' 'BUSY'

Maintenant au niveau de la commande **chat**, on va rajouter au lieu de :

```
"/usr/sbin/chat -v -f $DIALER_SCRIPT"
```

Ceci

```
"/usr/sbin/chat -v -f $DIALER_SCRIPT -r /tmp/report-file"
```

Attention, ces commandes apparaissent deux fois dans le fichier. Maintenant lors d'une connexion, l'inscription suivante va s'inscrire dans le fichier **/tmp/report-file** :

```
Opening "/tmp/report-file"...
```

```
chat: Jul 14 09:17:56 CONNECT 45333/ARQ/V90/LAPM/V42BIS
```

```
Closing "/tmp/report-file".
```

Ce fichier n'est pas écrasé mais incrémenté à chaque connexion. Il vous suffit à chaque début de connexion de jeter un coup d'oeil dedans pour voir à quelle vitesse vous êtes connectés.

Voici un petit script pour en extraire la vitesse:

```
#!/bin/bash
LINE=`fgrep CONNECT /tmp/report-file`
set $LINE none
echo $6 | awk -F "/" '{ print $1 }'
```

On peut faire en sorte que ce soit inscrit dans le fichier **historique-connexion**, pour cela en début de fichier **ppp-on** on va d'abord supprimer tout fichier **/tmp/report-file** en rajoutant:

```
rm -f /var/log/ChatReport
```

Copier le script d'extraction de la vitesse dans **/etc/ppp** et appelez le **speed**, donnez lui les droits 755. Maintenant apportez ces modifications dans le fichier **ip-down**, au lieu de :

```
date /var/log/historique-connexion
```

On va écrire

```
echo -n "`date`|" /var/log/historique-connexion
echo "`/etc/ppp/speed`" /var/log/historique-connexion
```

Voilà un extrait maintenant du fichier **historique-connexion** :

```
voonoo|Fri Jul 14 10:23:52 CEST 2000|Fri Jul 14 10:23:59 CEST 2000|45533
liberty|Fri Jul 14 10:26:18 CEST 2000|Fri Jul 14 10:26:42 CEST 2000|40000
```

4.4.3 Connaître la vitesse de connexion

Vous devez modifier légèrement le fichier **ppp-on-dialer** la première ligne du fichier doit être:

```
'REPORT' 'CONNECT'
```

Maintenant au niveau de la commande **chat** du fichier **ppp-on**, on va rajouter au lieu de :

```
"/usr/sbin/chat -v -f $DIALER_SCRIPT"
```

Ceci

```
"/usr/sbin/chat -v -f $DIALER_SCRIPT -r /tmp/report-file"
```

Maintenant lors d'une connexion, l'inscription suivante va s'inscrire dans le fichier `/tmp/report-file`:

Opening `/tmp/report-file`...

```
chat: Jul 09 08:28:40 CONNECT 36000/ARQ/V90/LAPM/V42BIS
```

Closing `/tmp/report-file`.

Ce fichier n'est pas écrasé mais incrémenté à chaque connexion. Il vous suffit à chaque début de connexion de jeter un coup d'oeil dedans pour voir à quelle vitesse vous êtes connecté.

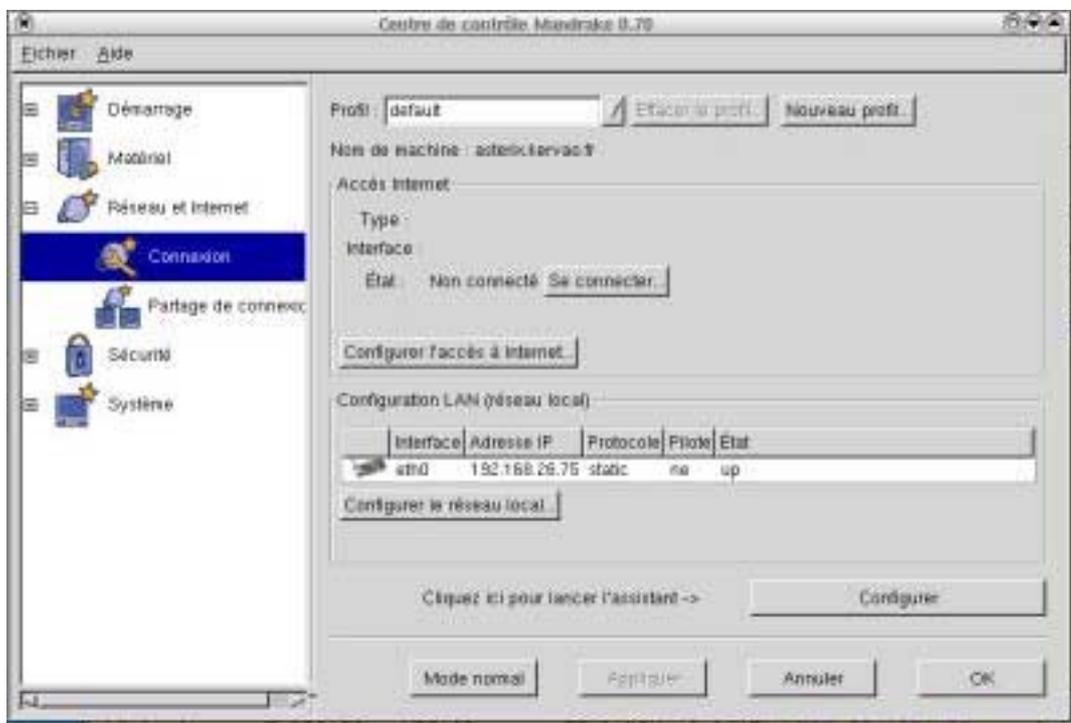
5 Configuration PPP façon clique-bouton

5.1 Présentation

Pour configurer la connexion Internet sur Linux, il n'est pas forcément nécessaire de trifouiller les fichiers, il existe des utilitaires graphiques qui vous simplifient la vie.

5.2 Configuration avec drakconf

Si vous ne l'avez pas déjà fait, branchez votre modem sur un port série libre. A partir d'un shell, lancez l'utilitaire **drakconf**, saisissez le mot de passe root. Pour info cet utilitaire se trouve dans le package du même nom.



Au niveau de **Réseau et Internet**, on choisit **Connexion**, puis **Mode expert** dans les boutons en bas. Si la connexion internet n'a pas déjà été configuré lors de l'installation de la Mandrake,

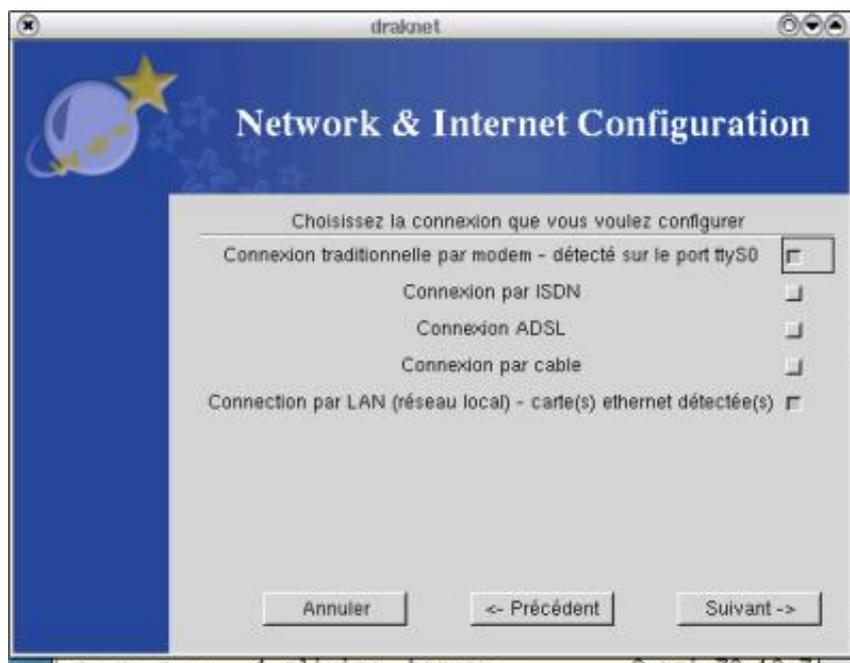
cliquez sur **Configurer l'accès à internet**.

Une fenêtre s'affiche avec la mention

Vous n'avez aucune connexion à Internet

Vous pouvez en créer une en cliquant sur "Configurer"

On passe sur la faute classique sur connexion et on clique sur **OK** pour fermer la fenêtre. On doit alors cliquer sur le bouton **Configurer** pour lancer l'assistant. Celui propose la détection automatique des périphériques. Cochez la case **Utilisez la détection automatique**, puis **Suivant**.



Il a détecté automatiquement le modem sur le port série 1 (désigné par **ttyS0**) et accessoirement ma carte réseau. On clique maintenant sur **Suivant**. Un message nous indique **Nous allons configurer la connexion modem** (sans faute cette fois-ci). Cliquez sur **Suivant**.

Options d'appel	
Nom de la connexion	fai
Numéro de téléphone	36804645
Identifiant de connexion	login-fai
Mot de passe	*****
Authentification	PAP
Nom de domaine	kervao.fr
Premier serveur de noms (DNS) (optionnel)	
Deuxième serveur de noms (DNS) (optionnel)	

On doit maintenant saisir les informations fournies par le fournisseur d'accès internet pour la connexion, à savoir:

- identifiant de la connexion (mettez ce que vous voulez)
- numéro de téléphone
- votre login
- votre mot de passe
- type d'authentification (CHAP pour free par exemple)
- nom de domaine, vous n'êtes pas obligé de mettre celui du FAI
- éventuellement les adresses DNS du FAI

Puis **Suivant** pour terminer la configuration. **Drakconf** passe ensuite à la configuration des autres connexions détectées, pour ma part cela concerne uniquement la connexion réseau, je clique sur **Annuler** si je ne veux pas aller plus loin.

5.3 kppp

kppp est l'interface graphique KDE pour permettre la connexion PPP vers un FAI.

Connexion à :	fnac
Nom d'utilisateur :	login-chez-fnac
Mot de passe :	*****
<input checked="" type="checkbox"/> Afficher la fenêtre de log	
<input type="button" value="Quitter"/> <input type="button" value="Configuration"/> <input type="button" value="Aide"/> <input type="button" value="Connecter"/>	

Il vous est possible avec **kppp** de définir plusieurs connexions, dans le cas par exemple où vous êtes abonnés à plusieurs FAI pour voir. Le nom de l'utilisateur et le mot de passe vous ont été attribués par le FAI. Pour configurer la connexion, il suffit d'appuyer sur la bouton **Configuration**.

A noter qu'en cochant **Afficher la fenêtre de log**, vous aurez des info sur le déroulement de la connexion, ce qui peut être intéressant pour déboguer.

Vous avez plusieurs onglets, dans l'onglet **Comptes**, vous allez définir les paramètres de connexion chez le FAI. Dans l'onglet **Périphérique**, vous allez définir les paramètres de communication avec votre modem. Les autres onglets sont d'importantes moindres.



Pour créer une connexion, il suffit d'appuyer sur la touche **Nouveau**, on nous donne le choix entre lancer l'assistant ou la boîte de dialogue classique. Je vous déconseille l'assistant car il faut que votre FAI soit dans une base de données des FAI de kppp. On choisit donc boîte de dialogue standard.

une autre fenêtre avec onglets apparaît:

- Onglet **Numérotation** avec les champs suivant:

* **Nom de la connexion** : Saisissez le nom du FAI

* **Numéro du téléphone**: Saisissez le numéro de téléphone pour joindre le FAI (cliquez sur Ajouter auparavant)

* **Authentification**: laissez PAP/CHAP ça marche dans la plupart des cas

* laissez la case **Enregistrer le mot de passe** cochée

* vous pouvez donner des options supplémentaires à **pppd** l'utilitaire qui sert à établir la connexion en cliquant sur Personnaliser les paramètres de pppd

- Onglet **IP**

* Dans la plupart des cas il faudra laisser cocher **Adresse IP dynamique**

* Dans la plupart des cas, il faudra laisser décocher **Configurer automatiquement le nom d'hôte à partir de cette adresse IP**

- Onglet **Passerelle**

* Dans la plupart des cas, il faudra laisser **Passerelle par défaut**

* Dans la plupart des cas, il faudra cocher **Définir cette passerelle comme itinéraire par défaut**

* **Identification**: Vous pouvez laisser celle par défaut, ça n'a pas d'incidence

* **Programme à lancer la connexion**: Vous pouvez indiquer un nom de programme à lancer à la connexion, fetchmail par exemple pour récupérer le courrier, ou un script quelconque

* **Programme à lancer avant la déconnexion**: Comme son nom l'indique

* **Programme à lancer à la déconnexion**: idem

* Vous pouvez éventuellement éditer les paramètres de **pppd**, je vous conseille de ne pas y toucher.

- Onglet **IP**

- * Cochez **adresse IP dynamique** qui est valable dans la majorité des cas
- * Vous pouvez laisser décoché la case **configurer automatiquement le nom d'hôte à partir de cette adresse IP**

- Onglet **DNS**

- * Indiquez un nom de domaine, qui n'est pas forcément celui du FAI
- * Dans la plupart des cas on mettra Configuration automatique mais si vous connaissez les adresses IP des serveurs primaires et secondaires du FAI, choisissez Configuration Manuelle puis rajoutez les adresses IP (appuyez sur **Ajouter** pour la prise en compte)
- * Vous pouvez laisser décocher la case **désactiver les serveurs DNS existants pendant la connexion**

- Onglet **Script de connexion**, sans intérêt la plupart du temps

- Onglet **Exécution**, qui permet de lancer des programmes après l'établissement de la connexion ou juste avant la fin, intéressant pour récupérer et envoyer automatiquement le mail et les posts de news.

- Onglet **Facturation**, pour activer une estimation du coût des communications.

Ca y est votre connexion est configurée. Maintenant on retourne à la première fenêtre de configuration:

- Onglet **Périphérique**

- * **Périphérique modem**: mon modem est identifié par **/dev/modem**
- * **Contrôle de flux**: CRTSCTS
- * **Terminaison de la ligne**: CR
- * **Vitesse de la connexion**: 115200 (mettre le max)
- * Cochez la case **utiliser un fichier de verrouillage**, j'imagine que c'est pour éviter de lancer plusieurs connexions PPP en même temps.
- * **temps limite du modem**: 60 secondes, je sais pas trop à quoi ça correspond, sûrement un timeout quelconque de réponse max du modem

- Onglet **Modem**

- * case **attendre la tonalité avant de numéroté** cochée
 - * **Attente active**: 0 seconde, un autre timeout
 - * **Volume du modem**: à la moitié chez moi
- Vous disposez de trois boutons pour tester votre modem.

- Onglet **Graphe et Divers**

Configurations diverses du logiciel **kppp**

Voilà c'est terminé, maintenant pour activer une connexion, il suffit d'appuyer sur le bouton **Connecter** de **kppp**. C'est à ce moment là qu'on va saisir le login et le mot de passe attribués par le FAI qu'on pourra sauvegarder.

6 Configuration PPP (cas de plusieurs providers)

6.1 Présentation

Ce paragraphe a pour objet de vous expliquer comment créer des scripts de connexion PPP dans le cas où vous disposez de plusieurs FAI, par exemple dans le cas où vous vous testez

plusieurs FAI gratuits.

6.2 Mise en place des scripts

On suppose que vous avez les fichiers de connexion PPP tel que décrits dans le paragraphe [connexion PPP à la main](#) . Je présente une config avec libertysurf et fnac qui se contente d'un login et d'un mot de passe, et free et waika9 qui nécessitent une identification par CHAP. Ce script ne gère pas les identifications PAP pour le moment.

Au cas où vous avez un serveur DNS qui tourne sur votre machine, créer le fichier `/etc/named` contenant :

```
};
};
zone "." {
    type hint;
    file "named.ca";
};
zone "kervao.fr"{
    type master;
    file "kervao.fr";
    notify no;
};
zone "0.0.127.in-addr.arpa"{
    type master;
    file "named.local";
};
```

On va modifier le fichier `/etc/ppp/ppp-on` comme suit:

```
#!/bin/bash
#
# définition du nom de domaine de la machine qui peut être totalement différent de celui
# du FAI
MON_DOMAINE=kervao.fr
# définition de l'interface ppp0, si vous en avez plusieurs,
# vous pouvez faire en sorte de la rentrer en argument
DEVICE=ppp0
# définition du fichier contenant les paramètres de connexion
DIALER_SCRIPT=/etc/ppp/ppp-on-dialer
# définition des options à passer à pppd
OPTIONS="modem defaultroute debug"
OPTIONS_DEF="-detach noauth lock crtscts asyncmap 00000000"
OPTIONS_CHAP="noauth ipcp-accept-remote ipcp-accept-local idle 180"
if [ "$1" = "" ] ; then
    #####
    ##### Pas de paramètre indiqué : on utilise la connexion par défaut
```

```

#####
CONNEXION=fnac
else
  CONNEXION=$1
fi

# à rajouter si vous voulez que sendmail soit reconfiguré à chaque connexion en fonction
# du FAI
# voir document sendmail
export $CONNEXION

#####
#
##### Initialisations diverses en fonction de la connexion désirée
#####
#
#
case $CONNEXION in
  fnac)
    #####
    # fnac ou mageos #
    #####
    echo Tentative de connexion sur Fnac
    TYPE=
    REMOTE_NAME=$DEVICE
    UTIL=login-chez-fnac
    MOTP=password-en-clair
    TELE=0298304698
    # en cas d'attribution automatique d'adresses IP des serveurs DNS du fai
    # commentez les lignes DNS et DNS2
    DNS="194.149.160.9 194.149.160.1"
    # si vous avez un serveur DNS décommenter la ligne suivante
    # DNS2=""194.149.160.9; 194.149.160.1;"
    ;;

  liberty)
    #####
    # libertysurf #
    #####
    echo Tentative de connexion sur LibertySurf
    TYPE=
    REMOTE_NAME=$DEVICE
    UTIL=login-chez-liberty
    MOTP=password-en-clair
    TELE=0298304691
    # en cas d'attribution automatique d'adresses IP des serveurs DNS du fai
    # commentez les lignes DNS et DNS2
    DNS="194.149.160.9 194.149.160.1"
    # si vous avez un serveur DNS décommenter la ligne suivante
    # DNS2="194.149.160.9; 194.149.160.1;"

```

```

;;

free)
#####
# free      #
#####
echo Tentative de connexion sur Free
TYPE=CHAP
REMOTE_NAME=proxad
UTIL=login-chez-free
MOTP=password-en-clair
TELE=0860912006
# en cas d'attribution automatique d'adresses IP des serveurs DNS du fai
# commentez les lignes DNS et DNS2
DNS="217.27.32.5 217.27.32.6"
# si vous avez un serveur DNS décommenter la ligne suivante
# DNS2="217.27.32.5; 217.27.32.6;"
;;

waika9)
#####
# waika9    #
#####
echo Tentative de connexion sur Waika9
TYPE=CHAP
REMOTE_NAME=proxad
UTIL=login-chez-waika9
MOTP=password-en-clair
TELE=0860912006
# en cas d'attribution automatique d'adresses IP des serveurs DNS du fai
# commentez les lignes DNS et DNS2
DNS="217.27.32.5 217.27.32.6"
# si vous avez un serveur DNS décommenter la ligne suivante
# DNS2="217.27.32.5; 217.27.32.6;"
;;

*)
echo "Nom de connexion inconnu $CONNEXION ??"
echo "Vous avez le choix entre free, liberty, waika9 et fnac"
exit 0
;;

esac

#####
#### Création du fichier /etc/resolv.conf
#### Commenter cette partie en cas d'attribution automatique des adresses
#### IP des serveurs DNS du fai
#####
echo "search $MON_DOMAINE" /etc/resolv.conf
# si vous avez un serveur DNS local décommenter la ligne suivante

```

```

# echo "nameserver 127.0.0.1" /etc/resolv.conf
for dns in $DNS ; do
    echo "nameserver $dns" /etc/resolv.conf
done

#####
#### Création du fichier /etc/named.conf
#### A décommenter en cas de serveur DNS local et
#### de NON attribution des adresses IP des serveurs DNS du FAI
#####
# echo "options {" /etc/named.conf
# echo "    directory \"/var/named\";" /etc/named.conf
# echo "    forwarders{" /etc/named.conf
# for dns in $DNS2 ; do
#     echo "    $dns" /etc/named.conf
# done
# cat /etc/named /etc/named.conf

#####
##### Initialisation du fichier /etc/ppp/ppp-on-dialer
#####
cat <<END_OF-DATA /etc/ppp/ppp-on-dialer
'ABORT' 'BUSY'
'ABORT' 'ERROR'
'ABORT' 'NO CARRIER'
'ABORT' 'NO DIALTONE'
'ABORT' 'Invalid Login'
'ABORT' 'Login incorrect'
" 'ATZ'
'OK' 'ATDT$TELE'
'CONNECT' "
END_OF-DATA

#####
##### Lancement de la connexion
#####

if [ $TYPE = "CHAP" ] ; then
echo "$UTIL * $MOTP" /etc/ppp/chap-secrets
    rm -f /etc/ppp/pap-secrets
    touch /etc/ppp/pap-secrets
    chmod 600 /etc/ppp/pap-secrets
    chmod 600 /etc/ppp/chap-secrets
    PPPOPTIONS="$OPTIONS $OPTIONS_CHAP remotename $REMOTE_NAME
user $UTIL"
exec /usr/sbin/pppd noauth /dev/ttyS0 115200 $PPPOPTIONS ipparam $CONNEXION
\
    connect "/usr/sbin/chat -v -f $DIALER_SCRIPT"
else
    rm -f /etc/ppp/chap-secrets

```

```

touch /etc/ppp/chap-secrets
chmod 600 /etc/ppp/pap-secrets
chmod 600 /etc/ppp/chap-secrets
PPOPTIONS="$OPTIONS $OPTIONS_DEF ipparam $CONNEXION"
#####
##### Complément du fichier /etc/ppp/ppp-on-dialer
#####
cat <<END_OF-DATA /etc/ppp/ppp-on-dialer
'ogin:' '$UTIL'
'ord:' '$MOTP'
'TIMEOUT' '5'
'~_' ''
END_OF-DATA

exec /usr/sbin/pppd $PPOPTIONS /dev/ttyS0 115200 \
    connect "/usr/sbin/chat -v -f $DIALER_SCRIPT"
fi

```

ATTENTION: Ne pas mettre d'espace ou de tabulation au niveau du '**ogin:**', jusqu'à **END_OF-DATA**.

NOTES:

- **ipparam** sert à passer un argument à **/etc/ppp/ip-up** le script qui sera appelé à l'établissement de la connexion, dans le cas présent on lui passe le nom du FAI utilisé, il servira après pour lancer certaines actions dans **ip-up** suivant le type de FAI. Pour avoir une idée de l'utilité de ce paramètre, voir le document **sendmail** ou la page disponible sur <http://www.funix.org>

- En cas d'attribution automatique des serveurs **DNS**, il n'est pas nécessaire de définir la variable **DNS**, le fichier **/etc/resolv.conf** sera automatiquement renseigné.

Voici maintenant à quoi ressemble le fichier **/etc/ppp/ip-up**

```

#!/bin/bash

# This file should not be modified -- make local changes to
# /etc/ppp/ip-up.local instead
# l'argument 4 correspond à l'adresse IP attribuée par le FAI
# l'argument 6 correspond est passé en argument de la commande pppd juste
# après ipparam, voir /etc/ppp/ppp-on
ADRESSE=$4
CONNEXION=$6

# écriture des paramètres de connexion dans un fichier de log
echo " " /var/log/connex
echo $1 /var/log/connex
echo "Connecté à $CONNEXION adresse $4 le " /var/log/connex
date /var/log/connex
nslookup $4|grep Name:|sed 's/Name:*/' /var/log/connex

# si serveur DNS local

```

```

# en cas d'attribution automatique d'adresses IP des serveurs DNS
# on décommente les lignes suivantes

#rm -f /etc/named.conf
#echo "options {"/etc/named.conf
#echo "    directory \"/var/named\";"/etc/named.conf
#echo "    forwarders{"/etc/named.conf
#cat /etc/ppp/resolv.conf| grep nameserver | awk -F" " '{if ($2 != "127.0.0.1") {print
$2";"}}' /etc/named.conf
#cat /etc/named /etc/named.conf

# jusqu'à ici

# Pour serveur DNS local, attribution auto ou pas
# on relance le serveur en décommentant la ligne suivante
# /etc/rc.d/init/named restart

# on envoie le courrier
# attention vous ne pourrez envoyer le courrier
# sur un serveur SMTP d'un fai que si vous êtes
# connecté avec ce fai
# /usr/sbin/sendmail -q -v /var/log/connex

# on récupère le courrier
# root doit posséder un .fetchmailrc
# relevant les boîtes aux lettres de tout le monde
/usr/bin/fetchmail -a

exit 0

```

A tout honneur, tout seigneur, les grandes lignes du script **ppp-on** m'ont été communiquées par [Bernard Delechamp](#) j'ai adapté ensuite. Pour appeler ce script, c'est très simple, il suffit de taper:

```
/etc/ppp/ppp-on liberty
```

A la place de liberty, dans mon exemple, on peut aussi bien mettre fnac, waika9 et free.

Servez vous ensuite de **sudo**, pour spécifier les utilisateurs qui pourront lancer une connexion PPP, ils taperont ensuite:

```
sudo /etc/ppp-on liberty
```

Maintenant si vous voulez que **sendmail** se reconfigure suivant le fai utilisé (pour pouvoir envoyer des emails), allez faire un tour sur la page ou consulter le document téléchargeable **sendmail** disponible sur <http://www.funix.org>.

Si pour une raison x ou y votre fichier **/etc/resolv.conf** n'est pas renseigné automatiquement en cas d'attribution automatique des adresses IP de votre serveur DNS une fois la connexion établie, rajoutez les lignes suivantes dans votre fichier **/etc/ppp/ip-up** en début de fichier (après le **#!/bin/bash** tout de même)

```
rm -f /etc/resolv.conf
```

```

echo "search kervao.fr" /etc/resolv.conf
# en cas de serveur DNS local, décommentez la ligne suivante
# echo "nameserver 127.0.0.1"
tail --lines 25 /var/log/messages | grep DNS | awk -F" " '{print "nameserver "$9}'
/etc/resolv.conf

```

6.3 Estimer le coût de ses communications

Il est intéressant de pouvoir faire des stats sur ses communications passées durant le mois, pour cela lors de chaque connexion nous allons noter le FAI utilisé, l'heure de début et de fin de connexion dans un fichier particulier.

Dans le fichier `/etc/ppp/ip-up` présenté plus haut, vous allez rajouter la ligne suivante:

```
echo -n "$CONNEXION|`date`" /var/log/historique-connexion
```

A présent dans le fichier `/etc/ppp/ip-down` rajouter la ligne suivante:

```
date /var/log/historique-connexion
```

Cela va donner un fichier qui aura cette tête là:

```
fnac|Sat Jun 24 18:45:35 CEST 2000|Sat Jun 24 18:54:46 CEST 2000
liberty|Sat Jun 24 19:14:02 CEST 2000|Sat Jun 24 19:34:11 CEST 2000
```

Maintenant libre à vous de créer un petit programme pour mettre en forme tout ça. Je vous propose un programme en PHP3 `connex.php3` qui est le suivant :

<?

```

$mfnc=array(0,0,0,0,0,0,0,0,0,0,0);
$mfrc=array(0,0,0,0,0,0,0,0,0,0,0);
$mlliberty=array(0,0,0,0,0,0,0,0,0,0,0);
$mfreesbee=array(0,0,0,0,0,0,0,0,0,0,0);
function heurefai($debut,$fin,&$fai)
{
$stabdebut=explode(" ",$debut);
$stabfin=explode(" ",$fin);
$heuredeb=explode(":",$stabdebut[3]);
$heurefin=explode(":",$stabfin[3]);
$secondedebut=($heuredeb[0]*3600)+($heuredeb[1]*60)+$heuredeb[2];
$secondefin=($heurefin[0]*3600)+($heurefin[1]*60)+$heurefin[2];
if ($stabdebut[1]==$stabfin[1])
{
// même jour même mois, pas de passage à minuit
if ($stabdebut[2]==$stabfin[2])
{
$seconde=$secondefin-$secondedebut;
}
// passage à minuit, même mois
if ($stabfin[2]>$stabdebut[2])
{

```

```

$secondedebut=86400-$secondedebut;
$seconde=$secondedebut+$secondefin;
}

$minute=$seconde/60;
switch ($stabdebut[1])
{
    case "Jan":
        $fai[0]=$fai[0]+$minute;
        break;
    case "Feb":
        $fai[1]=$fai[1]+$minute;
        break;
    case "Mar":
        $fai[2]=$fai[2]+$minute;
        break;
    case "Apr":
        $fai[3]=$fai[3]+$minute;
        break;
    case "May":
        $fai[4]=$fai[4]+$minute;
        break;
    case "Jun":
        $fai[5]=$fai[5]+$minute;
        break;
    case "Jul":
        $fai[6]=$fai[6]+$minute;
        break;
    case "Aug":
        $fai[7]=$fai[7]+$minute;
        break;
    case "Sep":
        $fai[8]=$fai[8]+$minute;
        break;
    case "Oct":
        $fai[9]=$fai[9]+$minute;
        break;
    case "Nov":
        $fai[10]=$fai[10]+$minute;
        break;
    case "Dec":
        $fai[11]=$fai[11]+$minute;
        break;
}
}
}
$nomfile="/var/log/historique-connexion";
$fp=fopen($nomfile,"r");
while(!feof($fp))
{

```

```

$ligne=fgets($fp,255);
$stab=explode("|",$ligne);
switch($stab[0])
{
    case fnac:
        heurfai($stab[1],$stab[2],$mfnc);
        break;
    case liberty:
        heurfai($stab[1],$stab[2],$mliberty);
        break;
    case freesbee:
        heurfai($stab[1],$stab[2],$mfreesbee);
        break;
    case free:
        heurfai($stab[1],$stab[2],$mfree);
        break;
    default:
        break;
}
}
for ($mois=0;$mois<=11;$mois++)
{
$mois[$mois]=$mfnc[$mois]+$mliberty[$mois]+$mfreesbee[$mois]+$mfree[$mois];
    $cmois[$mois]=$mois[$mois]*0.14;
}
echo "<table border=1>";
echo "<tr><td>FAI</td><td>Janvier</td><td>Février</td><td>Mars</td>";
echo "<td>Avril</td><td>Mai</td><td>Juin</td><td>Juillet</td>";
echo
"<td>Août</td><td>Septembre</td><td>Octobre</td><td>Novembre</td>";
echo "<td>Décembre</td></tr>";
echo "<tr></tr>";
echo
"<tr><td>Fnac/Mageos</td><td>$mfnc[0]</td><td>$mfnc[1]</td><td>$mfnc[2]</t
d>";
echo
"<td>$mfnc[3]</td><td>$mfnc[4]</td><td>$mfnc[5]</td><td>$mfnc[6]</td><td>$
mfnc[7]</td><td>$mfnc[8]</td>";
echo "<td>$mfnc[9]</td><td>$mfnc[10]</td><td>$mfnc[11]</td></tr>";
echo "<tr><td>LibertySurf</td><td>$mliberty[0]</td><td>$mliberty[1]</td>";
echo "<td>$mliberty[2]</td><td>$mliberty[3]</td><td>$mliberty[4]</td>";
echo "<td>$mliberty[5]</td><td>$mliberty[6]</td><td>$mliberty[7]</td>";
echo "<td>$mliberty[8]</td><td>$mliberty[9]</td><td>$mliberty[10]</td>";
echo "<td>$mliberty[11]</td></tr>";
echo "<tr><td>Free</td><td>$mfree[0]</td><td>$mfree[1]</td>";
echo "<td>$mfree[2]</td><td>$mfree[3]</td><td>$mfree[4]</td>";
echo "<td>$mfree[5]</td><td>$mfree[6]</td><td>$mfree[7]</td>";
echo "<td>$mfree[8]</td><td>$mfree[9]</td><td>$mfree[10]</td>";

```

```

echo "<td>$mfree[11]</td></tr>";
    echo "<tr><td>Freesbee</td><td>$mfreesbee[0]</td><td>$mfreesbee[1]</td>";
echo "<td>$mfreesbee[2]</td><td>$mfreesbee[3]</td><td>$mfreesbee[4]</td>";
echo "<td>$mfreesbee[5]</td><td>$mfreesbee[6]</td><td>$mfreesbee[7]</td>";
echo "<td>$mfreesbee[8]</td><td>$mfreesbee[9]</td><td>$mfreesbee[10]</td>";
echo "<td>$mfreesbee[11]</td></tr>";
    echo "<tr></tr>";
    echo "<tr><td>Total min</td><td>$mois[0]</td><td>$mois[1]</td>";
echo "<td>$mois[2]</td><td>$mois[3]</td><td>$mois[4]</td>";
echo "<td>$mois[5]</td><td>$mois[6]</td><td>$mois[7]</td>";
echo "<td>$mois[8]</td><td>$mois[9]</td><td>$mois[10]</td>";
echo "<td>$mois[11]</td></tr>";
echo "<tr><td>Coût estimé F</td><td>$cmois[0]</td><td>$cmois[1]</td>";
echo "<td>$cmois[2]</td><td>$cmois[3]</td><td>$cmois[4]</td>";
echo "<td>$cmois[5]</td><td>$cmois[6]</td><td>$cmois[7]</td>";
echo "<td>$cmois[8]</td><td>$cmois[9]</td><td>$cmois[10]</td>";
echo "<td>$cmois[11]</td></tr>";
    echo "</table>";

fclose($fp);
?>

```

qui présentera les résultats sous forme d'une page HTML, comme ceci (les chiffres sont en minute):

| FAI | Janvier | Février | Mars | Avril | Mai | Juin | Juillet | Août | Septembre | Octobre | Novembre | Décembre |
|----------------------|---------|---------|------|-------|-----|---------------------|---------|------|-----------|---------|----------|----------|
| Fnac/Mageos | 0 | 0 | 0 | 0 | 0 | 9.183333333
3333 | 0 | 0 | 0 | 0 | 0 | 0 |
| LibertySurf | 0 | 0 | 0 | 0 | 0 | 6.15 | 0 | 0 | 0 | 0 | 0 | 0 |
| Free | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Freesbee | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Total (min) | 0 | 0 | 0 | 0 | 0 | 15.33333333
3333 | 0 | 0 | 0 | 0 | 0 | 0 |
| Coût estimé
(FHT) | 0 | 0 | 0 | 0 | 0 | 2.146666666
6667 | 0 | 0 | 0 | 0 | 0 | 0 |

Ce script PHP3 présente un défaut il ne gère pas le passage d'un mois à un autre, si vous vous connectez à cheval d'un mois à un autre (avec passage à minuit), la communication n'est pas prise en compte. De même pour le coût comme j'appelle uniquement en heure creuse (0.14FHT la min), il n'y a pas de distinction de ce côté là (heure creuse, heure pleine), par ailleurs l'établissement de chaque appel est facturé 0.615FHT, ce n'est pas pris en compte.

C'est la première version du script, par ailleurs je ne suis pas une bête de programmation, je compte néanmoins l'améliorer, pour corriger notamment les défauts cités plus haut. Si vous trouvez un bug ou si vous avez des remarques au sujet de ce programme n'hésitez pas à m'en faire part.

Note Désolé j'ai pas encore converti mon script en euros, je vous laisse ce soin.

7 Lancer la connexion d'un poste client

7.1 Présentation

Cette page a pour objet de présenter les outils qui présente de lancer une connexion d'un poste client (sous windows, linux ou encore autres choses) du réseau privé sans avoir à intervenir sur le serveur linux sur lequel est branché le modem.

Vous avez la possibilité de déclencher la connexion manuellement avec **masqdiabler**, ou alors automatiquement avec **pppd** ou **diald**, c'est à dire que du poste client (ou du poste linux) dès que vous essayez d'aller sur le net (un **ping** suffit), la connexion va se déclencher. Avec la méthode manuelle avec **masqdiabler**, il est nécessaire d'installer un programme sous windows qui présente l'intérêt de donner le débit de connexion et une indication sur le trafic entrant et sortant, avec la méthode automatique nul besoin d'installer un logiciel supplémentaire sous windows, il suffit de vouloir aller sur le net. La déconnexion dans le premier cas est déclenchée du poste client au travers du programme de connexion, dans le deuxième cas, c'est au bout d'un certain temps d'inactivité que la connexion est automatiquement coupée.

Personnellement je préfère **masqdiabler** car on maîtrise la connexion, en plus on a le choix du FAI à utiliser, par contre son développement semble à l'abandon ce qui est dommage.

On considèrera l'adresse IP du réseau local comme étant 192.168.13.*, celle du serveur 192.168.13.1, il a pour nom **asterix**.

7.2 Masqdiabler

7.2.1 Présentation

Masqdiabler est un daemon à installer sur votre poste linux connecté à internet (par PPP, ADSL, RNIS, ...) de manière ponctuelle. Il permet à des utilisateurs choisis de lancer une connexion à partir d'un poste distant de votre réseau privé qui se trouve sous Windows ou éventuellement sous un autre OS (MacOS, ...), il autorise aussi à ces mêmes utilisateurs de stopper la connexion.

L'intérêt de **masqdiabler** est qu'il faut une intervention manuelle du poste Windows pour pouvoir lancer la connexion, a contrario **diald** ou **pppd** qui permettent la même chose sont des daemons qui scrutent l'activité de votre réseau, ils déclencheront une connexion automatiquement dès qu'on veut aller voir à l'extérieur, ce qui peut être gênant.

7.2.2 Installation de Masqdiabler

7.2.2.1 Récupération de l'archive

On peut récupérer **masqdiabler** à l'adresse w3.cpwright.com/mserver, on y trouve un package

RPM de binaires prêt à l'emploi **c-mserver-0_5_5-4_i386.rpm**, ou le package des sources à compiler **c-mserver-0_5_5-4_src.rpm**.

7.2.2.2 Installation avec RPM

Pour le package binaire il suffit pour l'installer de taper en tant que root:

```
rpm -ivh c-mserver-0_5_5-4_i386.rpm
```

Ca va vous installer un exécutable **mserver** dans **/usr/sbin**, un répertoire de doc **c-mserver-0.5.5** dans **/usr/doc** et un fichier de config **mserver.conf** dans **/etc**. Il va de plus configurer le démarrage pour que le serveur se lance au boot. Vous allez donc retrouver un fichier **mserver** sous **/etc/rc.d/init.d** avec cette tête là (avec des droits d'exécution) :

```
#!/bin/sh
# Copyright (c) 1995-1998 S.u.S.E. GmbH Fuerth, Germany.

# Original Author: Mike Klinkert
# New Author: Willi Eigenmann
# Modified to suit redhat distribution 22/06/1999

# chkconfig: 345 93 36
# description: Starts and stops the mserver \
#          used to provide Internet access.

#    /etc/rc.d/init.d/mserver

# Source function library.
./etc/rc.d/init.d/functions

# Source networking configuration.
./etc/sysconfig/network

# Check that networking is up.
[ ${NETWORKING} = "no" ] && exit 0

# Check that mserver.conf exists.
[ -f /etc/mserver.conf ] || exit 0

case "$1" in
  start)
    echo "Starting mserver."
    /usr/sbin/mserver
    ;;
  stop)
    echo -n "Shutting down mserver: "
    killproc /usr/sbin/mserver
    echo
    ;;
  restart|reload)
    $0 stop
    $0 start
    ;;
```

```

check|status)
    echo -n "Checking for mserver, pid is: "
    pidof /usr/sbin/mserver
    echo
    ;;
*)
    echo "Usage: $0 {start|stop|restart|check}"
    exit 1
    ;;
esac

exit 0

```

Par ailleurs les liens symboliques suivants sur le fichier `/etc/rc.d/init.d/mserver` sont créés automatiquement:

```

/etc/rc.d/rc0.d/K36mserver
/etc/rc.d/rc1.d/K36mserver
/etc/rc.d/rc2.d/K36mserver
/etc/rc.d/rc3.d/S93mserver
/etc/rc.d/rc5.d/S93mserver
/etc/rc.d/rc6.d/K36mserver
/etc/rc.d/rc4.d/S93mserver

```

7.2.2.3 Installation avec tarball

Pour les inconditionnels du `tar.gz`, on peut récupérer les sources `c-mserver-0.5.5.tar.gz` qu'on décompresse en tapant :

```
tar xvfz c-mserver-0.5.5.tar.gz
```

Cela créé un répertoire `mserver-0.5.5` dans lequel vous taperez :

```
./configure --sysconfdir=/etc
```

puis

```
make
```

Et enfin en tant que `root`

```
make install
```

Par défaut le serveur `mserver` se trouve sous `/usr/local/bin`. Pour un lancement automatique vous avez un fichier `mserver` se trouvant sous `./mserver-0.5.5/initscripts` à copier sous `/etc/rc.d/init.d`, mais ce fichier est pas terrible je vous propose de créer plutôt le fichier `mserver` proposé plus haut par l'installation RPM et d'y adapter les chemins qui vont bien. Donnez lui les droits en exécution.

```
chmod 755 /etc/rc.d/init.d/mserver
```

Maintenant pour un lancement automatique à l'état de marche 3,4 et 5 :

```
chkconfig --level 345 mserver on
```

Et un arrêt aux autres états de marche

```
chkconfig --level 0126 mserver off
```

7.2.3 Configuration de Masqodialer

7.2.3.1 Le fichier `/etc/mserver.conf`

Pour configurer **masqodialer**, il suffit d'éditer et de modifier `/etc/mserver.conf`. Le fichier est créé automatiquement avec l'installation RPM, avec l'installation avec tarball, copier le fichier **mserver.conf** se trouvant `./mserver-0.5.5/mserver/` sous `/etc`. Voici les lignes importantes du fichier

```
# c-mserver config file  
# Charles P. Wright  
# cpwright@villagenet.com
```

```
# en clair ici, on indique ce quelles interfaces le daemon mserver doit écouter en attente de demande de connexion
```

```
# vous allez mettre l'adresse IP de votre serveur linux, dans mon exemple 192.168.13.1,  
# le port indiqué 224 doit être cohérent avec celui déclaré dans /etc/services (voir plus bas)
```

```
# 127.0.0.1 ne doit pas être changé, ça correspond à la machine linux elle même  
listen_on = "192.168.13.1:224,127.0.0.1:224"
```

```
(...)
```

```
# How many clients can we have waiting on the incoming queue before they get  
# locked out
```

```
# limitation du nombre de clients  
slimit = 10
```

```
(...)
```

```
# If there is more than one client is one of them allowed to hangup the  
# connection?
```

```
# plusieurs clients peuvent-ils couper la connexion  
multiclientkill = true
```

```
(...)
```

```
# What version of getstat should we use? pppdlock, netdev, lockonly and  
# exitcode are your choices if you are using PPP then you should really use  
# pppdlock. If you have a weird method of connecting use netdev. If you want  
# to check if a file simply exists then use lockonly. Exitcode will execute a  
# program and if it returns the value in goodcode then the link will count as  
# being up. pppdpid will record the pid of pppd when it connects and ensures  
# that that process is still a pppd.  
# pppdlock correspond au fichier de lock de ppp
```

```
# statype = pppdlock
# chez moi, en choisissant pppdlock je ne peux stopper la connexion d'un client, j'ai
choisi
statype = lockonly
```

(...)

```
# What is the name of your interface that exists when and only when you are
# connected? This can used for the stat check and it is used for the netload
# feature. You can set this on a device by device basis.
# nom de l'interface ppp
netdev = ppp0
```

(...)

```
# What command should we use to check the link status? If the link is good
# it should return 0, otherwise it should return a non-zero value. This is used
# as the format string to sprintf, the %s will be replaced with the checkhost
# which is defined below.
# commande à taper pour vérifier qu'on est bien connecté
linkcheck = "/bin/ping -c 1 %s"
```

(...)

```
# What host should we make sure is active when performing a linkcheck? I
# suggest using an IP number instead of a hostname to expedite the operation,
# this gets performed a lot and the longer it takes for it to happen the more
# frustrated a user will get.
# machine de test à pinguer pour voir si on est connecté
checkhost = "194.149.160.9"
```

(...)

```
# What is the lockfile for your modem, dial out as normal and then look in
# /var/lock
# En choisissant statype=lockonly, on doit préciser le fichier lock du modem
# pour ma part c'est celui-là (et non pas LCK..modem)
lockfile = "/var/lock/LCK..ttyS0"
```

(...)

```
# What machines are allowed to capture the modem?
# 127.0.0.1 correspond à votre machine linux (ne pas toucher cette adresse),
# 192.168.13.* au préfixe des adresses IP de votre réseau privé
capture_ipallow = "127.0.0.1:192.168.13.*"
```

```
# What machines are allowed to administer the server. I recommend you make
# this a machine that only you have access to. The firewall (localhost) is
# usually a good choice. You can disable this by setting it to 0.0.0.0 (I am
# assuming no host will have that IP address).
# la machine qui peut administrer le serveur (par défaut localhost)
```

```

admin_ipallow = "127.0.0.1"

# What users are allowed to administer the server. I would choose your account
# but not root. Having the root password being transmitted over the network
# can be a *really* bad thing. If you are using it on only loopback it isn't
# as bad.
# comme son nom l'indique, franchement je vois pas trop l'intérêt d'une telle commande
# pour moi l'admin de masqdiaper se réduit à un vi sur le fichier de config en tant que
root
admin_userallow = "olivier"

# What machines are allowed to lock and unlock connections?
# qui a le droit de créer un fichier de lock et à le virer
lock_ipallow="192.168.13.*:127.0.0.1"

# What users are allowed to lock and unlock connections?
# mettez les utilisateurs de votre réseau (éventuellement *)
lock_userallow = "olivier,veronique"

(...)

# What file contains the username/password combinations. It is in the format of
# /etc/passwd. You might want to choose something else as your file since
# transmitting unix passwords over the network in the clear is a bad thing.
# You can use the provided authgen utility to setup an authorization file.
# si vous mettez en place une authentification par login/mot de passe
# il faut indiquer ici le fichier qui va bien
authfile = "/etc/passwd"

# Should we use shadow passwords? This requires that you run the daemon as
# root and that you use the account names in /etc/passwd. This is probably
# not as good as a separate authorization file because the passwords are being
# sent over the network, but it is convenient. If you use this option the
# authfile will be ignored and the server will use the functions in shadow.h!
# si vous avez mis en place les shadows passwd mettre true
shadow = true

(...)

# What script should we run just after a client connects?
# The argument is the client hostname
# ici c'est le script qui va être lancé quand le client va se connecter
# sur le serveur linux (et non pas chez le FAI)
# j'ai laissé la valeur par défaut, à tout hasard j'ai créé un fichier du
# même nom mais vide, cette option est très intéressante à mon avis
clientup = "/usr/local/share/msserver/clientup"

# What script should we run just after a client disconnects?
# The argument is the client hostname
# script à appeler lors d'une déconnexion du poste client du poste linux
# et non pas du FAI

```

```

# j'ai laissé la valeur par défaut, à tout hasard j'ai créé un fichier du
# même nom mais vide
clientdown = "/usr/local/share/mserver/clientdown"

# What script should we run just after a user authenticates?
# The argument is the client hostname and username
#script à lancer lors de l'authentification sur le poste linux
# j'ai laissé la valeur par défaut, à tout hasard j'ai créé un fichier du
# même nom mais vide
authup = "/usr/local/share/mserver/authup"

# What script should we run just after a client unauthenticates?
# The argument is the client hostname and username
# j'ai laissé la valeur par défaut, à tout hasard j'ai créé un fichier du
# même nom mais vide
authdown = "/usr/local/share/mserver/authdown"

# What script should we run after the link has been brought up?
# This can be specified on a per connection basis.
# The argument is the connection's name
# script à lancer quand une connexion PPP est établie, le script
# standard est ip-up
linkup = "/etc/ppp/ip-up"

# What script should we run right before the link has been intentionally brought
# down?
# script à appeler juste avant la déconnexion,
prekill = "/etc/ppp/goingdown"

# What script should we run after the link has been intentionally brought down?
# script à appeler juste après la déconnexion du FAI
# traditionnellement ip-down
linkdown = "/etc/ppp/ip-down"

(...)

# Which connection do you want mserver to auto-dial?
# comment this out if you don't want it to auto-dial.
# décommentez si on veut une connexion automatique
#autodial "VillageNet(Balder)"

# What are the names of our connections? The order that they are listed
# is the order that LIST will use, the first one is the default connection
# for some clients.
# on indique ici les connexion possibles
cname = "fnac,liberty"

# Actually Define our connections

# What is the command you use to bring up the connection, this can be a script
# or you can call pppd directly.
# script à lancer pour lancer une connexion sur la fnac (voir paragraphe connexion PPP

```

```

avec plusieurs FAI)
fnac_script = "/etc/ppp/ppp-on fnac"

# What IP addresses are allowed to manipulate this connection. You are allowed
# up to 10 different masks separated by colons. You must either use the real
# numbers or a '*' to match any in each quad. You must specify all of the quads# that is
# '*' wouldn't work but '*.*.*.*' would.
# adresses IP des postes réseau autorisés à se connecter avec la fnac
fnac_ipallow = "192.168.13.*:127.0.0.1"

# What users are allowed to dial out this connection? "*" or "" will allow any
# user to dial out, otherwise provide a comma separated list of users.

# This is a string that is displayed when CINFO is called, it isn't actually
# used for any dialing, but just for cosmetics. The clients may eventually
# depend on this so I suggest setting it.
# ch'tit commentaire
fnac_info = "Connexion à la fnac"

# The internal kill method is much more reliable unless you have something wierd# you
# should really use it!
# j'ai laissé ça en commentaire, visiblement c'est pas utile
# fnac_kill = "/etc/ppp/ppp-off"

# The per connection netdev, this is only needed for the netdev stat method
# it defaults to the value of netdev.
# nom du device ppp0 utilisé
fnac_netdev = "ppp0"

# The per connection checkhost, this is only needed for the exitcode stat method
# it defaults to the value of checkhost.
# adresse IP d'une machine sur internet à pinguer pour voir si on est connecté
fnac_checkhost = "194.149.160.9"

# Which method of parsing the logfile should we use for this connection?
# j'ai laissé ça par défaut
fnac_parselog = "pap"

# Another connection
# définition de la connexion liberty
liberty_script = "/etc/ppp/ppp-on liberty"
liberty_info = "Connexion à Liberty"
liberty_checkhost = "194.149.160.9"
liberty_ipallow = "127.0.0.1:192.168.13.*"

(...)

# What users are allowed to do what?
# Quels utilisateurs sont autorisés à faire quoi
userallow = "olivier"      # olivier peut tout faire
userallow_connect = "*"    # n'importe qui peut lancer une connexion
userallow_disconnect = "*" # n'importe qui peut stopper une connexion

```

7.2.3.2 Le fichier `/etc/services`

Voilà on a fini avec la config de **mserver**, maintenant il faut déclarer le port que le daemon va "écouter" pour attendre les demandes de connexion/deconnexion. Voilà les lignes à rajouter (c'est fait automatiquement à l'install du RPM, à faire avec le tarball) dans le fichier `/etc/services`.

```
masqdialer 224/tcp      masqdialer # added by c-mserver
masqdialer 224/udp      masqdialer # added by c-mserver
```

7.2.4 Lancement de Masqdialer

Pour lancer tout ça, vous pouvez taper la commande:

```
/usr/sbin/mserver -v -c /etc/mserver.conf
```

Le chemin est `/usr/local/bin/mserver` dans les cas d'une install par tarball.

Ou alors

```
/etc/rc.d/init.d/mserver start
```

Voilà la trace au lancement dans `/var/log/messages`

```
Jul 15 07:57:58 asterix mserver[2496]: Listening on address: 192.168.13.1
Jul 15 07:57:58 asterix mserver[2496]: Listening on address: 127.0.0.1
```

Lors d'une connexion on aura les messages suivants

```
Jul 15 08:13:36 asterix mserver[2497]: Forked new client 2549
Jul 15 08:13:36 asterix mserver[2497]: Signon: client.kervao.fr
Jul 15 08:13:36 asterix mserver[2549]: Set Authorized: 0
Jul 15 08:13:36 asterix mserver[2549]: Set Authorized: 0
Jul 15 08:13:36 asterix mserver[2549]: Set Authorized: 1
Jul 15 08:13:36 asterix mserver[2549]: Get Authorized: 1
Jul 15 08:13:36 asterix mserver[2549]: connection name: mana
Jul 15 08:13:36 asterix mserver[2549]: allowed ips: 192.168.13.*:127.0.0.1
Jul 15 08:13:36 asterix mserver[2549]: Get Authorized: 1
Jul 15 08:13:36 asterix mserver[2549]: connection name: mana
Jul 15 08:13:36 asterix mserver[2549]: allowed users: olivier,*
Jul 15 08:13:36 asterix mserver[2549]: Token = "olivier"
Jul 15 08:13:36 asterix mserver[2549]: Get Authorized: 1
Jul 15 08:13:36 asterix mserver[2549]: Dialing connection: mana by user olivier Jul 15
08:13:36 asterix mserver[2549]: Executing: /etc/ppp/ppp-on
Jul 15 08:13:36 asterix pppd[2552]: pppd 2.4.1 started by root, uid 0
Jul 15 08:13:37 asterix mserver[2549]: PPPD pid is 2552
(...)
```

7.3 *Diald*

7.3.1 Présentation

Diald permet de lancer une connexion à internet de manière automatique à partir de votre

poste connecté à internet mais aussi des postes clients de votre réseau. Est présentée ici une connexion de type **ppp** pour un accès internet en utilisant LibertySurf et Free.

7.3.2 Installation

On récupérera l'archive **diald-1.0.tar.gz** à l'URL diald.sourceforge.net, on la décompresse en tapant :

```
tar xvzf diald-1.0.tar.gz
```

Cela va créer un répertoire **diald-1.0**. Dans ce répertoire, on doit rajouter la ligne

```
#include <time.h
```

Dans les fichiers suivants :

```
diald.c  
firewall.c  
timer.c
```

(en première place avant **#include <config.h**)

on va taper ensuite successivement

```
./configure --sysconfdir=/etc  
make
```

Puis en tant que root

```
make install
```

Ce qui va installer un certain nombre de fichiers de conf sous **/usr/local/lib/diald**

Et les exe suivants

```
/usr/local/sbin/diald  
/usr/local/bin/dctrl
```

Par ailleurs on trouvera le fichier **/etc/pam.d/diald** pour permettre une authentification des clients en utilisant **PAM**

Pour modifier éventuellement l'emplacement des exécutables et des fichiers de configuration, tapez configure avec certaines options que vous obtiendrez en tapant

```
./configure -help
```

7.3.3 Configuration

On crée maintenant le fichier **/etc/ppp/ppp-on-dialer** si ce n'est déjà fait ([connexion PPP à la main](#)) contenant les informations de connexion, je prends pour exemple une connexion chez LibertySurf

```
ABORT BUSY  
ABORT "NO CARRIER"
```

```
ABORT VOICE
ABORT "NO DIALTONE"
"" ATDT0298304691
ogin login-chez liberty
word \qmot-de-passe
```

A noter le \q devant le mot de passe Ce format de fichier passe également

```
'REPORT' 'CONNECT'
'ABORT' 'BUSY'
'ABORT' 'ERROR'
'ABORT' 'NO CARRIER'
'ABORT' 'NO DIALTONE'
'ABORT' 'Invalid Login'
'ABORT' 'Login incorrect'
" 'ATZ'
'OK' 'ATDT0298304691'
'CONNECT' "
'ogin:' 'login-chez-liberty'
'ord:' 'mot-de-passe'
'TIMEOUT' '5'
'~~~' "
```

Pour une connexion avec **Free** on tapera :

```
'REPORT' 'CONNECT'
'ABORT' 'BUSY'
'ABORT' 'ERROR'
'ABORT' 'NO CARRIER'
'ABORT' 'NO DIALTONE'
'ABORT' 'Invalid Login'
'ABORT' 'Login incorrect'
" 'ATZ'
'OK' 'ATDT0860912006'
'CONNECT' "
```

N'oubliez pas le fichier `/etc/ppp/chap-secrets`

login-free * mot-de-passe-de-connexion

Voici maintenant le fichier de config à créer `/etc/diald.conf`, on considère que le modem est branché sur le port série 1 (`/dev/ttyS0`) et que l'adresse IP locale est 192.168.13.1

```
fifo /var/run/diald.fifo
mode ppp
connect "/usr/sbin/chat -v -f /etc/ppp/ppp-on-dialer"
device /dev/ttyS0
speed 115200
modem
lock
rtscts
local 192.168.13.1
```

```
dynamic
defaultroute
pppd-options noauth asyncmap 0
ip-up /etc/ppp/ip-up
ip-goingdown /etc/ppp/goingdown
ip-down /etc/ppp/ip-down
include /usr/local/lib/diald/standard.filter
```

Le fichier `/usr/local/lib/diald/standard.filter` fixe les requêtes (suivant le protocole) qui nécessitent une connexion internet et le temps de déconnexion minimum par type de requête. Exemple avec une requête HTTP, le temps d'attente est fixé à 2min (2*60s)

```
accept tcp 120 tcp.dest=tcp.www
accept tcp 120 tcp.source=tcp.www
```

Faire **man diald** pour avoir le détail de la syntaxe. Vous trouverez également davantage d'options pour le fichier **diald.conf**

On suppose que votre fichier `/etc/resolv.conf` est correctement renseigné

```
# définition de votre domaine privé
# qui peut être différent de celui du FAI
search kervao.fr
# en cas de serveur DNS local
# nameserver 127.0.0.1
# définition des serveurs de noms de votre FAI
nameserver adresse-IP-premier-serveur-DNS-FAI
nameserver adresse-IP-deuxième-serveur-DNS-FAI
```

Dans le cas où ce n'est pas le cas (attribution automatique des adresses IP des serveurs DNS), on modifiera le fichier désigné par la variable **ip-up** qui est lancé juste après la connexion, il va nous servir à renseigner automatiquement le fichier `/etc/resolv.conf`, il doit contenir au moins les lignes suivantes:

```
#!/bin/bash

rm -f /etc/resolv.conf
echo "search kervao.fr" /etc/resolv.conf
# en cas de serveur DNS local, décommentez la ligne suivante
# echo "nameserver 127.0.0.1"
tail --lines 25 /var/log/messages | grep DNS | awk -F" " '{print "nameserver "$9}'
/etc/resolv.conf
```

J'extrais les adresses IP des serveurs DNS à partir du fichier `/var/log/messages`

```
Jul 9 19:04:19 asterix diald[4538]: Running pppd (pid = 4615).
Jul 9 19:04:19 asterix pppd[4615]: pppd 2.4.1 started by root, uid 0
Jul 9 19:04:19 asterix pppd[4615]: Using interface ppp0
Jul 9 19:04:19 asterix pppd[4615]: Connect: ppp0 <-- /dev/tts/0
Jul 9 19:04:22 asterix pppd[4615]: local IP address 202.3.236.230
Jul 9 19:04:22 asterix pppd[4615]: remote IP address 202.3.224.37
Jul 9 19:04:22 asterix pppd[4615]: primary DNS address 202.3.225.10
```

Jul 9 19:04:22 asterix pppd[4615]: secondary DNS address 202.3.225.20

A la suite de ces lignes de **ip-up**, vous pouvez lancer d'autres exe à la connexion comme **sendmail** pour envoyer le courrier et **fetchmail** pour en recevoir (attention dans ce cas mettre le chemin en absolu des exécutable).

ip-goingdown désigne le fichier à lancer juste avant de se déconnecter, il pourrait contenir ceci

```
#!/bin/bash
/usr/sbin/sendmail -q
```

ip-down désigne le fichier à lancer juste après s'être déconnecté. Ces trois derniers fichiers sont des exécutable (droit à 755).

7.3.4 Premiers essais

Pour lancer **diald** il suffit de taper :

diald

Pour éventuellement bénéficier de avantages de détails niveau debug, lancer **diald** avec l'option suivante :

diald debug 8

Les infos de déboguage vont apparaître dans le fichier **/var/log/messages**.

Jul 9 10:50:06 asterix diald[4426]: ignore parsing error. Got token 'udp.route'. Not a known udp service port.

Jul 9 10:50:06 asterix diald[4426]: parse string: 'udp udp.dest=udp.route'

Jul 9 10:50:06 asterix diald[4426]: ignore parsing error. Got token 'udp.route'. Not a known udp service port.

Jul 9 10:50:06 asterix diald[4426]: parse string: 'udp udp.source=udp.route'

Jul 9 10:50:06 asterix diald[4426]: ignore parsing error. Got token 'udp.route'. Not a known udp service port.

Jul 9 10:50:06 asterix diald[4426]: parse string: 'udp tcp.dest=udp.route'

Jul 9 10:50:06 asterix diald[4426]: ignore parsing error. Got token 'udp.route'. Not a known udp service port.

Jul 9 10:50:06 asterix diald[4426]: parse string: 'udp tcp.source=udp.route'

Jul 9 10:50:06 asterix diald[4427]: Creating FIFO

Jul 9 10:50:06 asterix diald[4427]: Using fifo /var/run/diald.fifo

Jul 9 10:50:07 asterix diald[4427]: Proxy device established on interface sl0

Jul 9 10:50:07 asterix diald[4427]: start sl0: SIOCSIFMETRIC: Opération non supportée

Jul 9 10:50:07 asterix diald[4427]: Diald initial setup completed.

L'erreur sur **udp.route** vient de mon fichier **/etc/services**, au niveau de

router 520/udp local routing process (on site)

J'ai rajouté la ligne suivante

route 520/udp local routing process (on site);

Pour lancer une connexion de votre serveur, il faut que votre fichier `/etc/resolv.conf` contienne les adresses IP des serveurs DNS de votre FAI, si vous les connaissez pas, c'est pas grave vous pouvez mettre n'importe quoi car le script `ip-up` comme on l'a vu plus haut va le renseigner correctement une fois la connexion établie.

Voilà ce que ça donne après une tentative de surf.

```
Jul 9 10:58:40 asterix diald[4539]: Trigger: udp 192.168.13.1/32820 202.3.225.10/53
Jul 9 10:58:40 asterix diald[4539]: Calling site (null)
Jul 9 10:58:41 asterix chat[4610]: report (CONNECT)
Jul 9 10:58:41 asterix chat[4610]: abort on (BUSY)
Jul 9 10:58:41 asterix chat[4610]: abort on (ERROR)
Jul 9 10:58:41 asterix chat[4610]: abort on (NO CARRIER)
Jul 9 10:58:41 asterix chat[4610]: abort on (NO DIALTONE)
Jul 9 10:58:41 asterix chat[4610]: abort on (Invalid Login)
Jul 9 10:58:41 asterix chat[4610]: abort on (Login incorrect)
Jul 9 10:58:41 asterix chat[4610]: send (ATZ^M)
Jul 9 10:58:41 asterix chat[4610]: expect (OK)
Jul 9 10:58:41 asterix chat[4610]: ATZ^M^M
Jul 9 10:58:41 asterix chat[4610]: OK
Jul 9 10:58:41 asterix chat[4610]: -- got it
Jul 9 10:58:41 asterix chat[4610]: send (ATDT3640^M)
Jul 9 10:58:42 asterix chat[4610]: expect (CONNECT)
Jul 9 10:58:42 asterix chat[4610]: ^M
```

(...)

```
Jul 9 10:59:01 asterix diald[4539]: Connected to site (null)
Jul 9 10:59:01 asterix diald[4539]: Running pppd (pid = 4611).
Jul 9 10:59:01 asterix pppd[4611]: pppd 2.4.1 started by root, uid 0
Jul 9 10:59:01 asterix pppd[4611]: Using interface ppp0
Jul 9 10:59:01 asterix pppd[4611]: Connect: ppp0 <-- /dev/tts/0
Jul 9 10:59:05 asterix pppd[4611]: local IP address 202.3.240.173
Jul 9 10:59:05 asterix pppd[4611]: remote IP address 202.3.224.36
Jul 9 10:59:05 asterix pppd[4611]: primary DNS address 202.3.225.10
Jul 9 10:59:05 asterix pppd[4611]: secondary DNS address 202.3.225.20
Jul 9 10:59:05 asterix diald[4539]: New addresses: local 202.3.240.173, remote
202.3.224.36, broadcast 0.0.0.0, netmask 255.255.255.255
Jul 9 10:59:05 asterix diald[4539]: start ppp0: SIOCSIFMETRIC: Opération non
supportée
```

(...)

```
Jul 9 10:59:41 asterix diald[4539]: Closing down idle link.
Jul 9 10:59:41 asterix diald[4539]: start sl0: SIOCSIFMETRIC: Opération non
supportée
Jul 9 10:59:42 asterix pppd[4611]: Terminating on signal 2.
Jul 9 10:59:42 asterix pppd[4611]: Connection terminated.
Jul 9 10:59:42 asterix pppd[4611]: Connect time 0.7 minutes.
```

```
Jul 9 10:59:42 asterix pppd[4611]: Sent 740 bytes, received 513 bytes.
Jul 9 10:59:42 asterix pppd[4611]: Exit.
Jul 9 10:59:42 asterix diald[4539]: Closing /dev/ttyS0
Jul 9 10:59:43 asterix diald[4539]: Delaying 30 seconds before clear to dial.
```

7.3.5 Configuration avancée

A noter que vous pouvez rajouter des restrictions sur les heures d'utilisation de **diald**, par exemple ici tous les jours de la semaine (lundi=1, ..., vendredi=5) la connexion ne peut se faire de minuit à 19h, elle peut se faire le reste du temps.

```
restrict 00:00:00 19:00:00 1 * *
or-restrict 00:00:00 19:00:00 2 * *
or-restrict 00:00:00 19:00:00 3 * *
or-restrict 00:00:00 19:00:00 4 * *
or-restrict 00:00:00 19:00:00 5 * *
or-restrict 00:00:00 20:00:00 6 * *
down
restrict * * * * *
up
```

Vous devez rajouter ce fichier à **diald.conf**. Pour plus d'info sur la syntaxe

man diald

7.3.6 Lancement automatique de diald

Pour un lancement automatique de **diald**, il existe un fichier **diald-1.0/setup/etc/init.d/diald** théoriquement à placer sous **/etc/rc.d/init.d** comme il marche pas trop chez moi, je vous propose celui-ci **/etc/rc.d/init.d/diald**

```
#!/bin/bash

# chkconfig: - 60 20
# description: diald permet de lancer une connexion internet \
#             de maniere automatique du serveur et d'un poste client
# processname: diald

# Get config.
./etc/sysconfig/network

# Get functions
./etc/rc.d/init.d/functions

# Check that networking is up.
if [ ${NETWORKING} = "no" ] ; then
    exit 0
fi

# See how we were called.
case "$1" in
    start)
```

```

    echo -n "Starting diald: "
    daemon diald
    echo
    touch /var/lock/subsys/diald
    ;;
stop)
    echo -n "Stopping diald: "
    killproc diald
    echo
    rm -f /var/lock/subsys/diald
    ;;
status)
    status diald
    ;;
restart)
    $0 stop
    $0 start
    ;;
*)
    echo "Usage: $0 {start|stop|status|restart}"
    exit 1
    ;;
esac

exit 0

```

Donnez lui des droits en exécution

```
chmod 755 /etc/rc.d/init/diald
```

Pour un lancement automatique à l'état de marche 3, 4 et 5 on tapera :

```
chkconfig --level 345 diald on
```

Pour un arrêt automatique à l'état de marche 0, 1, 2 et 6 on tapera :

```
chkconfig --level 0126 diald off
```

Maintenant pour lancer **diald** il suffit de taper

```
/etc/rc.d/init.d/diald start
```

7.4 pppd

7.4.1 Présentation

pppd est le programme qui permet de lancer une connexion PPP, il possède une option qui permet d'en lancer une automatiquement si nécessaire et de couper la connexion au bout d'un certain temps. J'ai pris comme exemple une connexion à fnac.net, pour une connexion avec authentification CHAP (Free, Waika9).

7.4.2 Configuration

La première chose à faire est de lancer une connexion puis de taper **route**:

```
[root@asterix init.d]# route -nv
Table de routage IP du noyau
Destination  Passerelle  Genmask      Indic Metric Ref  Use Iface
202.3.224.34 0.0.0.0     255.255.255.255 UH  0   0   0 ppp0
192.168.13.0 0.0.0.0     255.255.255.0  U   0   0   0 eth0
127.0.0.0     0.0.0.0     255.0.0.0     U   0   0   0 lo
0.0.0.0       202.3.224.34 0.0.0.0       UG  0   0   0 ppp0
```

192.168.13.0 est l'adresse de votre sous domaine IP privé, **202.3.224.34** qu'on retrouve au niveau de la colonne passerelle est l'adresse IP de routage de votre fai, toutes vos requêtes vont passées par cette adresse, notez bien cette adresse dans un coin.

On crée un script **connect-demand**, qu'on place dans **/usr/bin**, voici son contenu:

```
#!/bin/bash
# script contenant les paramètres de connexion
DIALER_SCRIPT=/etc/ppp/ppp-on-dialer
# nom du device ppp
DEVICE=ppp0
# port de branchement du modem, ttyS0 port série 1
MODEM=/dev/ttyS0
# options pppd
PPPOPTIONS="usepeerdns lock modem crtscts asyncmap 00000000 defaultroute
debug"

exec /usr/sbin/pppd -detach $PPPOPTIONS $MODEM 115200 \
  remotename $DEVICE ipparam $DEVICE idle 60 demand 0.0.0.0:202.3.224.34 \
  connect "/usr/sbin/chat -v -f $DIALER_SCRIPT"
```

Pour info **115200** est la vitesse de connexion, **idle 60** (chiffre en seconde) permet de stopper la connexion au bout de 60 secondes d'inactivité. Le fichier **/etc/ppp/ppp-on-dialer** est identique à celui décrit dans le paragraphe [connexion PPP à la main](#). Rendez ce fichier exécutable (**chmod 755 /usr/bin/connect-demand**).

On suppose que votre fichier **/etc/resolv.conf** est correctement renseigné

```
# définition de votre domaine privé
# qui peut être différent de celui du FAI
search kervao.fr
# en cas de serveur DNS local
# nameserver 127.0.0.1
# définition des serveurs de noms de votre FAI
nameserver adresse-IP-premier-serveur-DNS-FAI
nameserver adresse-IP-deuxième-serveur-DNS-FAI
```

Dans le cas où ce n'est pas le cas (attribution automatique des adresses IP des serveurs DNS), on modifiera les options passées à **pppd** comme ceci:

```
PPPOPTIONS="usepeerdns lock modem crtscts asyncmap 00000000 defaultroute
```

debug"

On modifiera le fichier **/etc/ppp/ip-up** pour qu'il mette à jour automatiquement **/etc/resolv.conf** comme ceci

```
#!/bin/bash
rm -f /etc/resolv.conf
echo "search kervao.fr" /etc/resolv.conf
# en cas de serveur DNS local, décommentez la ligne suivante
# echo "nameserver 127.0.0.1" /etc/resolv.conf
echo "nameserver " $DNS1 /etc/resolv.conf
echo "nameserver " $DNS2 /etc/resolv.conf
```

Pour info c'est **pppd** qui passe les adresses IP à **ip-up** à travers les variables **DNS1** et **DNS2** grâce à l'option **usepeerdns** (**man pppd** pour plus de détails).

7.4.3 Lancement automatique de pppd

Pour lancer votre script automatiquement au boot, vous rajouterez à la fin de **/etc/rc.d/rc.local**

```
/usr/bin/connect-demand &
```

ATTENTION N'oubliez surtout pas le **&** (lancement en mode background) sinon ça bloque le boot !!

Pour lancer une connexion de votre serveur, il faut que votre fichier **/etc/resolv.conf** contienne les adresses IP des serveurs DNS de votre FAI, si vous les connaissez pas, c'est pas grave vous pouvez mettre n'importe quoi car le script **ip-up** comme on l'a vu plus haut va le renseigner correctement une fois la connexion établie.

Ca y est c'est fini, dès qu'on voudra aller sur le net (du poste linux ou du poste client) la connexion va se lancer automatiquement.

NOTE En accédant à un site local avec une adresse locale (du style **www.funix.kervao.fr**), une connexion automatique est déclenchée ! Pourtant dans le fichier **/etc/nsswitch.conf** j'ai bien l'ordre de recherche des hosts avec d'abord **/etc/hosts** puis les serveurs DNS du fai (**hosts: files dns**). C'est ennuyeux ...

7.5 Configuration des postes clients

7.5.1 Configuration générale

Vos clients doivent être configurés de telle sorte que votre serveur **masqdiabler**, **pppd** ou **diald** soit la passerelle, pour les clients linux, le fichier **/etc/sysconfig/network** doit ressembler à ca

```
NETWORKING=yes
FORWARD_IPV4=false
HOSTNAME=client.kervao.fr
DOMAINNAME=kervao.fr
GATEWAY=serveur
GATEWAYDEV=eth0
```

La commande route doit donner les lignes

Table de routage IP du noyau

| Destination | Passerelle | Genmask | Indic | Metric | Ref | Use | Iface |
|--------------|-------------------|---------------|-------|--------|-----|-----|-------|
| 192.168.13.0 | * | 255.255.255.0 | U | 0 | 0 | 0 | eth0 |
| 127.0.0.1 | * | 255.0.0.0 | U | 0 | 0 | 0 | lo |
| default | serveur.kervao.fr | 0.0.0.0 | U | 1 | 0 | 0 | eth0 |

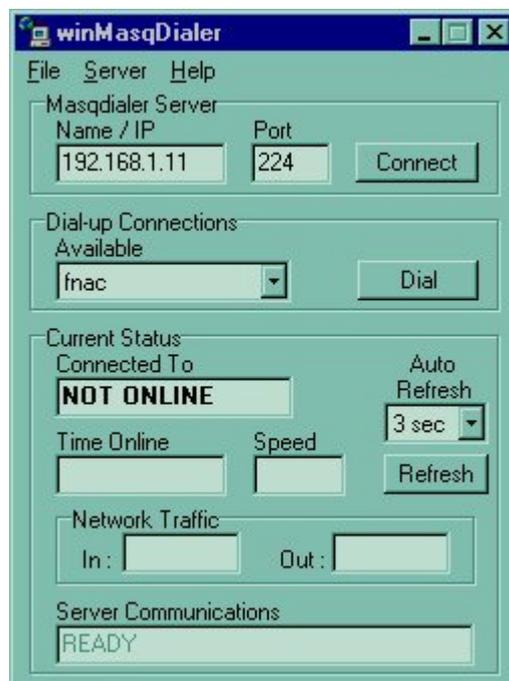
L'IP Masquerade doit évidemment être configuré sur le serveur **masqdiabler**, **pppd** ou **diald**. Concernant la configuration DNS, soit vous avez un serveur DNS local et dans ce cas vos clients en sont clients., soit vous n'en avez pas et dans ce cas, vous devez indiquer dans le fichier `/etc/resolv.conf`, les adresses IP des serveurs DNS de votre FAI. Si vous les connaissez pas, le plus simple est d'installer un serveur DNS local et de voir du côté du paragraphe [connexion PPP à la main](#) comment récupérer ces adresses IP.

Pour les clients windows voir au niveau de la configuration réseau.

7.5.2 Dans le cas de masqdiabler

7.5.2.1 Clients Windows

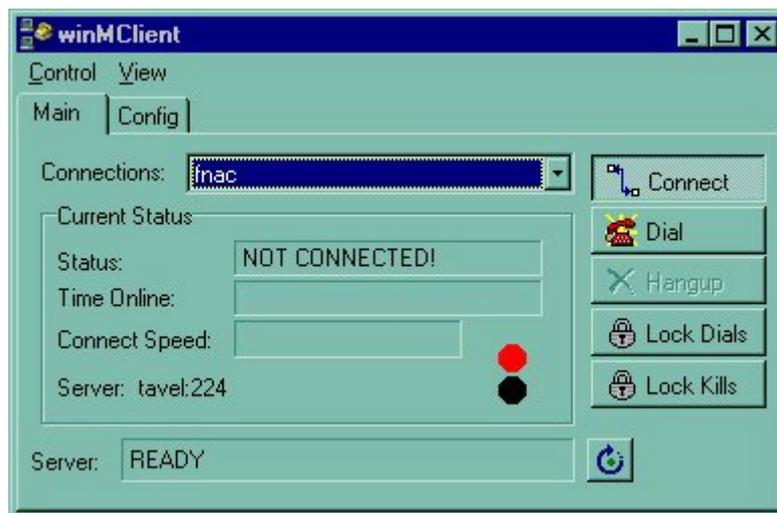
Sur le site de **masqdiabler**, on indique des adresses de programmes clients tournant sous Windows ou sous d'autres d'OS, permettant de lancer une connexion PPP à distance. Certains sites proposaient des clients mais ils ne sont malheureusement plus accessibles, à défaut vous pourrez télécharger ces exécutables sur le site <http://www.funix.org>.



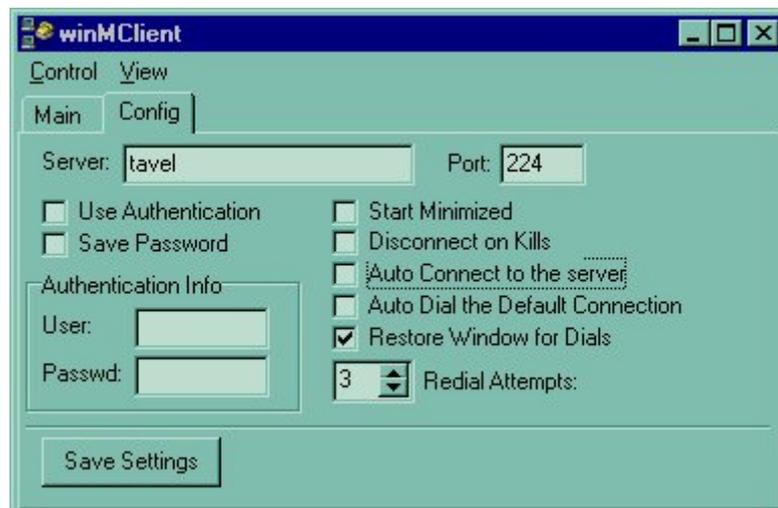
[WinMasqDialer](#) se récupère à l'adresse suckfree.com/masq, c'est un zip qui contient un exe unique et quelques fichiers de ressources (icônes, mais pas de dll ou autres).

Voici à droite un screenshot, il suffit d'indiquer l'adresse IP (dans le champ **Name/IP**) ou le nom de votre serveur linux, le numéro du port (224 par défaut). Quand on se connecte au serveur linux, on a alors la liste des connexions possibles (dans mon cas fnac et liberty). Pour lancer une connexion, il suffit d'en choisir une, puis de cliquer sur **Dial**, on a alors le débit de connexion et une idée du trafic. Pour stopper la connexion, le bouton **Dial** se transforme en bouton **Hangup**, il suffit de cliquer dessus.

A noter que lors d'une connexion, on a des petites icônes dans la barre de tâche windows en bas à droite.

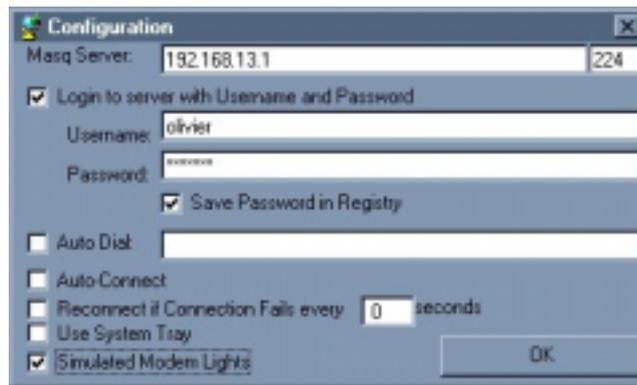


[WinMClient](http://www.buffnet.net/~millard/winmclient.html) se trouvait à l'adresse www.buffnet.net/~millard/winmclient.html, c'est un fichier zip qui contient un exécutable unique. A gauche un screenshot, le bouton **Connect** permet de se connecter au serveur linux et d'avoir la liste des connexions possibles. **Dial** permet de se connecter après avoir choisi son FAI et **Hangup** permet de raccrocher.

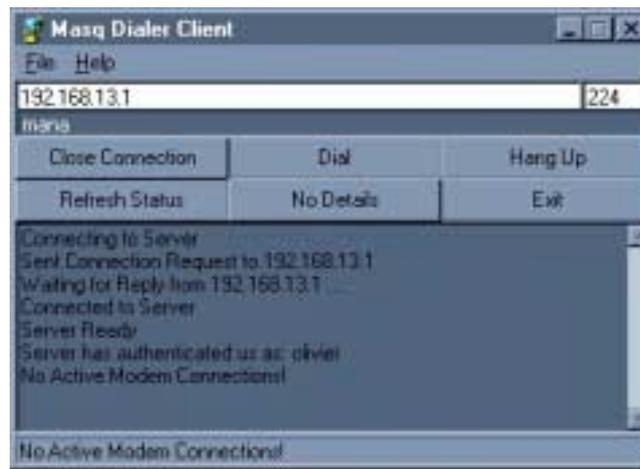


Voilà la configuration de **WinMClient** quand on clique sur l'onglet **Config**, on doit indiquer le nom du serveur linux (ou son adresse IP) ainsi que le port, vous pouvez choisir l'authentification et éventuellement d'autres paramètres, la config peut être sauvegardée.

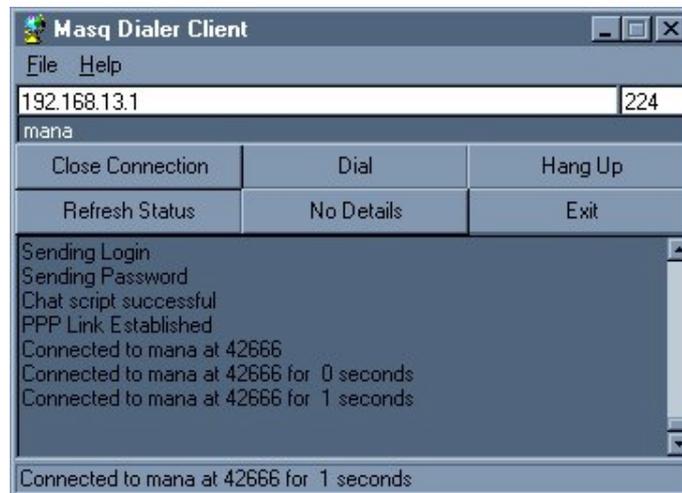
Le seul client windows disponible sur le site de masqdiabler se trouve à l'URL <http://w3.cpwright.com/win95nt-mclient/>. Il se trouve sous la forme d'une archive zip **win95nt-mclient-1.2.2-i386.zip**. Voilà quelques screenshots.



Configuration du client, on précise le nom du serveur, le login sur le serveur.



Connexion au serveur avec affichage des détails en cliquant sur **Connect to Server**



Lancement de la connexion en cliquant sur **Dial**. On stoppe la connexion en cliquant sur **Hang Up** (Raccrocher)

7.5.2.2 Clients Linux

Sur le même site que **MasqDialer**, précisément ici <http://w3.cpwright.com/cli-mclient/> on trouve un client linux en ligne de commande. On décompresse l'archive en tapant

tar xvzf cli-mclient-2.8.tar.gz

Cela donne un répertoire **mclient-2.8** dans lequel on tape

make

Libre à vous ensuite de l'installer sous **/usr/bin**

Pour avoir la liste des connexions PPP disponibles sur le serveur on tapera

mclient -h asterix -U olivier -P mot-de-passe-login-sur-serveur -v -l
Unix Command Line Masqdiabler Client v2.7
Copyright (c) 1998, 1999 Kevin Lindsay and Charles P. Wright

Connecting to server: asterix

Connection Established...

Authentication Successful!

Dialup Connections Available

mana
free

Pour se connecter sur la connexion **mana**, on tape

mclient -h asterix -c mana -U olivier -P mot-de-passe-login-sur-serveur -v -d
Unix Command Line Masqdiabler Client v2.7
Copyright (c) 1998, 1999 Kevin Lindsay and Charles P. Wright

Connecting to server: asterix

Connection Established...

Authentication Successful!

Dialing ...

Dialing connection: mana by user olivier

Executing: /etc/ppp/ppp-on

PPPD Process Started

Modem connected

Initializing Modem

Dialing Modem

Modem connected

Modem Connected at 42666

Sending Login

Sending Password

Chat script successful

PPP Link Established

Connected to mana at 42666

Pour avoir un état de la connexion

mclient -h asterix -c mana -U olivier -P mot-de-passe-login-sur-serveur -v -s

Pour stopper la connexion

mclient -h asterix -c mana -U olivier -P mot-de-passe-login-sur-serveur -v -k

7.5.3 Dans le cas de pppd ou diald

Après les légères modifications indiquées plus haut, il suffit d'inscrire une URL quelconque dans un navigateur, de taper enter et c'est parti la connexion se lance, au bout d'un certain temps (défini dans le script de connexion) d'inactivité la connexion est automatiquement coupée.

A noter que les accès sur les sites locaux peuvent déclencher une connexion.