

Fli4L

C'est quoi Fli4L ?

Fli4L est un routeur Ethernet, ISDN (RNIS), ADSL fondé sur Linux (Debian).

Une seule disquette est nécessaire. Et il suffit d'un PC 486 avec 16 Mo de RAM pour créer un routeur.

La disquette peut être créée sous Unix, Linux ainsi que sous Windows.

Aucune connaissance spécifique à Linux n'est nécessaire mais il est néanmoins utile de maîtriser les commandes de base du système. Et une connaissance de base des réseaux telle que TCP/IP, DNS et routage est également conseillée.

Par Eric M.C.DECLERCK
eric.m.c.declerck@free.fr

Caractéristiques :

- Création de la disquette du routeur sous Unix, Linux et Windows.
- Configuration à l'aide d'un seul fichier texte (ASCII).
- Support pour IP-Masquerading et Port-Forwarding (NAT et PAT).
- Least-Cost-Routing (LCR) : sélection automatique du fournisseur d'Accès Internet en fonction d'un horaire prédéfini et du coût des communications.
- Affichage du protocole de la connexion et calcul du coût des communications.
- Client Windows/Unix/Linux imonc inclus (interface vers imond et telmond).
- Upload de nouveaux fichiers via le client Windows, imonc (afin de mettre à jour le routeur).
- Disquette de démarrage en vfat.
- Utilisation possible de disquettes de 1680 ko.
- Filtrage de paquets (Packet filter) : contrôle des accès vers les ports du routeur (firewall).
- Utilisation d'interfaces virtuelles WAN appelées circuits.
- Utilisation possible en parallèle des circuits ADSL et RNIS.

Routeur :

- Noyau Linux 2.2.19.
- IP-Masquerading + PacketFilter pour la sécurité.
- Serveur DNS afin d'éviter l'accès des PC Windows vers le WAN pour la plupart des requêtes.
- Affichage des informations de connexion (monitoring) et LCR pour le serveur imond.
- Gestion des appels téléphoniques entrants pour le serveur telmond.

Support Ethernet :

- Gestionnaires pour la plupart des cartes réseau : plus de 40 familles de cartes reconnues.

Support ADSL :

- Driver Roaring Penguin PPPoE avec Dial-on-Demand (numérotation à la demande facultative).
- PPTP pour les connexions ADSL en Autriche et Pays-Bas (EXPERIMENTAL).

Support ISDN :

- Drivers HiSaX actuels : support pour 37 adaptateurs ISDN.
- Différentes variantes de connexion : in/out/callback, raw-ip/ppp.
- Agrégation de canaux (Multilink PPP) : sélection automatique ou manuelle en fonction de la bande passante.
- Sélection manuelle du second canal via le client Windows/Unix.
- En option : routage IPX.

Applications en option (packages) :

- Serveur DNS.
- Serveur DHCP.
- Login SSH.
- Service Telnet et/ou FTP.
- Affichage online/offline simple par LED.
- Programme d'affichage en LCD avec format changeable.
- Console série en option (voir HOWTO : http://www.fli4l.de/german/extern/docu/stable/doc/english/html/akap_en.html).
- Mini-serveur pour ADSL monitoring.
- Module IPSEC, ipsec et pptp.
- Restriction d'accès pour des réseaux distants.
- Support PCMCIA (EXPERIMENTAL).
- Protocole des messages système: syslogd et klogd.
- Configuration des adaptateurs ISA-PNP: iasnp tools.
- Utilitaires supplémentaires pour le débogage.
- Configuration du port série.
- Système de secours pour le contrôle distant via ISDN.
- Ecran LCD: affichage des connexions et du flux de données.
- Serveur/routeur PPP via le port série.
- Emulateur ISDN via le port série.
- Serveur d'impression.
- Accès au serveur de temps pour la synchronisation du réseau.
- Exécution de commandes/procédures quand un appel téléphonique se présente (ex: Internet dial-in).
- Support pour l'IP aliasing (permet d'avoir plus d'une adresse IP par carte réseau).

Hardware :

- ISDN : minimal : CPU 386 à 25MHz, recommandé : CPU 486 à 33 MHz.
- ADSL : minimal : 486 CPU - DX2/66, recommandé : CPU 486 DX4/100 à 100 MHz ou CPU Pentium à 75 MHz.
- Mémoire 8MB, recommandé : 16 MB.
- Adaptateur Ethernet (support pour plus de 40 familles d'adaptateurs).
- ISDN : adaptateur ISDN supporté par HiSaX (Type 1-37), AVM-B1 ISA/PCI ou ICN-2B.
- Pas de disque dur, juste un lecteur de disquette.
- Une disquette de démarrage contenant tous les fichiers nécessaires.

Pour l'affichage des informations de contrôle du routeur FLI4L, une autre application est

Fli4L

disponible : le client *imonc*. Ce programme est disponible pour Windows et pour Linux (unix/gtk-imonc).

Pour Windows : <http://www.wallmeier-online.de/imonc/>.

Pour Linux et gtk : <http://userpage.fu-berlin.de/~Ezeank/gtk-imonc/>.

Vous trouverez ici d'autres clients pour KDE, Windows, etc. :

<http://www.fli4l.de/german/addons.htm>.

Site Web : <http://www.fli4l.de>.

Auteur du programme : Frank Meyer.

Vous trouverez ces informations en anglais et bien d'autres à l'adresse :

http://www.fli4l.de/german/extern/docu/stable/doc/english/html/index_en.html.

Installation

Il est conseillé de commencer par une installation sur disquette. Mais, il est également possible d'installer FLI4L sur un disque dur ou une cartouche Zip.

Cette solution (utilisation d'une disquette) est idéale pour tout ceux qui possèdent un second ordinateur déjà relié en réseau via un hub (concentrateur) ou un switch (commutateur).

Je vais ici décrire l'installation que j'ai faite pour un essai de connexion en ISDN.

Afin de rendre les choses un peu difficiles j'ai choisi d'utiliser la carte Gazel ISA.

L'équipement de mon réseau :

1. Cartes réseau D-LINK DFE-530TX
2. HUB SMC 3616TC
3. Câbles Cat5 non croisées
4. Adaptateurs ISDN:
 - Gazel ISA-PNP
 - ELSA microlink PCI

Ordinateur faisant fonction temporaire de routeur : Pentium 200 MHz muni d'un lecteur de disquette.

Procédure :

Il faut d'abord s'assurer que les utilitaires isapnp sont installés sur l'ordinateur qui fera fonction de routeur.

- Faire un **pnpdump > isapnp.conf-tmp**.
- Editer le fichier *isapnp.conf-tmp*. Enlever toutes les parties qui ne concernent pas la carte Gazel. Ne pas oublier de préciser l'IRQ et l'IO exact de la carte. Pour connaître ces deux paramètres, vous pouvez vous reporter aux informations fournies par le Gestionnaire de périphériques) de Windows (si Windows est installé sur le PC), ou bien lancer un utilitaire de configuration sous MS-Dos livré avec la carte ISDN.
- Copier *isapnp.conf-tmp* sur une disquette (formatée en vfat) en le renommant *isapnp.conf*.
- Attention, dans la doc on préconise de le faire après le démarrage du routeur. A éviter, cela ne marche pas !
- Il est inutile de faire quoi que ce soit sous le compte root pour FLI4L. Donc, restez sous votre

Fli4L

compte utilisateur.

- Télécharger *fli4l-2.0.4.tar.gz* depuis (http://www.fli4l.de/english/e_download.htm ou <http://www.fli4l.de/german/download.php>).
- Télécharger *isdn.tar.gz*.
- Télécharger *gtk-imonc-0.1.6.tar.gz* (trop fainéant pour taper les commandes au clavier :)).
- Décompacter *fli4l-2.0.4.tar.gz* dans un sous-répertoire de mon dossier utilisateur. (ex : `~/isdn-routeur`).
- Décompacter *isdn.tar.gz* dans le dossier *fli4l-2.0.4/* (ex : `~/isdn-routeur/fli4l-2.0.4/`).
- Décompacter *gtk-imonc-0.1.6.tar.gz* dans un sous dossier de mon dossier utilisateur.
- Copier *isapnp.conf* qui se trouve sur la disquette dans `~/isdn-routeur/fli4l-2.0.4/opt/etc/`.
- Aller dans `~/isdn-routeur/fli4l-2.0.4/`.

POUR FLI4L :

- Editer `~/isdn-router/fli4l-2.0.4/config/base.txt`.
- L'adapter en fonction de sa configuration et des services que l'on souhaite mettre en place et sauver ensuite le fichier *base.txt*.

Voici mon fichier de configuration *base.txt* :

```
#-----
# General settings:
#-----
HOSTNAME='xxxxxx'          # name of fli4l router (ex. fli4l)
PASSWORD='xxxxxx'         # password for telnetd, ftpd and sshd (nécessaire)
MOUNT_BOOT='rw'           # mount boot device (floppy): ro, rw, no

RAMSIZE='2048'             # size of ramdisk for unzipped opt.tgz
# the variables MOUNT_OPT, PART_OPT and UPDATE_MODE will be ignored if
# RAMSIZE is not empty. see docu
MOUNT_OPT='ro'            # mount opt device: ro, rw
PART_OPT='ram1'           # location of opt-files, ram1 or disk-partition
UPDATE_MODE='full'        # add, cfg, full, none, see documentation

#-----
# Ethernet card drivers:
# uncomment your ethernet card
#-----
ETH_DRV_N='1'             # number of ethernet drivers to load, usually 1
#ETH_DRV_1='3c505'        # ISA: 3COM Etherlink Plus (3c505)
#ETH_DRV_1='3c507'        # ISA: 3COM Etherlink 16 (3c507)
#ETH_DRV_1='3c509'        # ISA: 3COM EtherLinkIII (3c509)
#ETH_DRV_1='3c515'        # ISA: 3COM EtherLink XL ISA (3c515)
#ETH_DRV_1='3c59x'        # PCI: 3COM Vortex/Boomerang 3c59x,3c900,3c905
#ETH_DRV_1='82596'        # Apricot Xen-II on board Ethernet
#ETH_DRV_1='3c503'        # ISA: 3COM EtherLinkII (3c503)
#ETH_DRV_1='e2100'        # ISA: Cabletron E21xx ISA
#ETH_DRV_1='hp'           # ISA: HP PCLAN (27245, 27xxx) ISA
#ETH_DRV_1='hp-plus'      # ISA: HP PCLAN+ (27247B and 27252A) ISA
#ETH_DRV_1='ne'           # ISA: NE2000 ISA clone (eg. Realtek 8019,
                          # Accton 16xx, NatSemi 8390, UMC 9003/9008)
#ETH_DRV_1='ne2k-pci'     # PCI: NE2000 PCI clone (eg. Realtek 8029,
                          # Winbond 89c940)
#ETH_DRV_1='smc-ultra'    # ISA: SMC ULTRA
```

Fli4L

```
#ETH_DRV_1='smc-ultra32'    # EISA: SMC ULTRA32 (NEW)
#ETH_DRV_1='wd'            # ISA: SMC WD80*3
#ETH_DRV_1='at1700'        # ISA: AT1700 (Fujitsu 86965) ISA
#ETH_DRV_1='cs89x0'        # ISA: IBM Etherjet, cs89x0 based Cards (Option io=0xnnn
                            necessary!)
#ETH_DRV_1='de4x5'         # PCI/EISA: Digital DE425, DE434, DE435, DE450, DE500
#ETH_DRV_1='depca'         # ISA: DEPCA, DE10x, DE200, DE201, DE202, DE422
#ETH_DRV_1='dgrs'          # PCI: Digi International RightSwitch PCI/EISA
#ETH_DRV_1='dmfe'          # PCI: DM9102 compatible PCI cards from Davicom
#ETH_DRV_1='elp486'        # ISA: Intel Professional Workstation/panther 82596
#ETH_DRV_1='eepro'         # ISA: Intel EtherExpress Pro/10
#ETH_DRV_1='eepro100'      # PCI: Intel EtherExpressPro PCI 10+/100B/100+
#ETH_DRV_1='eexpress'      # ISA: EtherExpress16 ISA
#ETH_DRV_1='epic100'       # PCI: SMC EPIC/100 (EtherPower II) PCI
#ETH_DRV_1='eth16i'        # ISA/EISA: ICL EtherTeam 16i/32
#ETH_DRV_1='ewrk3'         # ISA: EtherWORKS 3 ISA (DE203, DE204, DE205)
#ETH_DRV_1='fealnx'        # PCI: ASOUND LAN 8139 card - not RTL8139 (NEW)
#ETH_DRV_1='hp100'         # ISA/EISA/PCI: HP 10/100VG PCLAN (ISA, EISA, PCI)
#ETH_DRV_1='lance'         # ISA: AMD LANCE and PCnet (AT1500, NE2100) ISA
#ETH_DRV_1='old_tulip'     # PCI: Old DECchip Tulip (dc21x4x) PCI
#ETH_DRV_1='pcnet32'       # PCI: AMD PCI PCnet32
#ETH_DRV_1='rtl8139-orig'  # PCI: RealTek 8129/8139 (not 8019/8029!)
#ETH_DRV_1='rtl8139'      # PCI: RealTek 8129/8139 (not 8019/8029!) (NEW)
#ETH_DRV_1='8139too'       # PCI: RealTek 8139 10/100 MB (NEW)
#ETH_DRV_1='sis900'        # PCI: SiS 900/7016
#ETH_DRV_1='sundance'     # PCI: DFE-550FX or DFE-530TXS (NEW)
#ETH_DRV_1='tlan'         # PCI: TI ThunderLAN (Compaq Netelligent ...)
#ETH_DRV_1='tulip'        # PCI: DECchip Tulip (dc21x4x) PCI
#ETH_DRV_1='natsemi'      # PCI: Nat Semi
#ETH_DRV_1='starfire'     # PCI: Starfire
ETH_DRV_1='via-rhine'     # PCI: VIA Rhine PCI (3043, VT86c100A, dfe-530tx)
#ETH_DRV_1='winbond-840'  # PCI: Winbond 840
#ETH_DRV_1='lanstreamer'  # Token Ring: IBM Auto LANStreamer PCI Adapter
#ETH_DRV_1='olympic'      # Token Ring: IBM cards (Pit/Pit-Phy/Olympic)
#ETH_DRV_1='ibmtr'        # Token Ring: IBM 16/4
#ETH_DRV_1='pcnet_cs'     # PCMCIA: NS8390-based cards (NE2000, DLINK etc)
#ETH_DRV_1='3c574_cs'     # PCMCIA: 3Com 574
#ETH_DRV_1='3c575_cb'     # PCMCIA: 3Com 575
#ETH_DRV_1='3c589_cs'     # PCMCIA: 3Com 589
#ETH_DRV_1='airo'         # PCMCIA: Airo 4500 & 4800 series cards
#ETH_DRV_1='airo_cs'      # PCMCIA: Airo 4500 & 4800 series cards
#ETH_DRV_1='eepro100_cb'  # PCMCIA: EtherExpress Pro 100
#ETH_DRV_1='epic_cb'      # PCMCIA: SMC 83c170 EPIC/100
#ETH_DRV_1='ibmtr_cs'     # PCMCIA: IBM Token Ring
#ETH_DRV_1='netwave_cs'   # PCMCIA: Netwave AirSurfer Wireless LAN
#ETH_DRV_1='nmclan_cs'    # PCMCIA: New Media Ethernet LAN
#ETH_DRV_1='ray_cs'       # PCMCIA: Raylink wireless cards
#ETH_DRV_1='smc91c92_cs'  # PCMCIA: SMC91c92-based cards
#ETH_DRV_1='tulip_cb'     # PCMCIA: DEC 21040-family cards
#ETH_DRV_1='wavelan_cs'   # PCMCIA: WaveLAN
#ETH_DRV_1='wavelan2_cs'  # PCMCIA: WaveLAN2
#ETH_DRV_1='wwlan_cs'     # PCMCIA: Lucent WaveLAN/IEEE 802.11
#ETH_DRV_1='xirc2ps_cs'   # PCMCIA: Xircom: CE2, CEM28, CEM33, or CE3
```

Fli4L

```
#ETH_DRV_1='wl24_cs'      # PCMCIA: ELSA Airlancer MC-2
#ETH_DRV_1='cs89x0_cs'    # PCMCIA: IBM EtherJet Ethernet Adapter

ETH_DRV_1_OPTION=""      # additional option, e.g. 'io=0x340' for ne

#-----
# Ether networks used with IP protocol:
#-----
IP_ETH_N='1'              # number of ip ethernet networks, usually 1
IP_ETH_1_NAME=""          # optional: other device name than ethX
IP_ETH_1_IPADDR='192.168.12.1' # IP address of your n'th ethernet card
IP_ETH_1_NETWORK='192.168.12.0' # network of your LAN
IP_ETH_1_NETMASK='255.255.255.0' # netmask of your LAN

#-----
# Additional routes, optional
#-----
IP_DEFAULT_GATEWAY=""     # normally not used, read documentation!
IP_ROUTE_N='0'            # number of additional routes
#IP_ROUTE_1='192.168.7.0 255.255.255.0 192.168.6.99' # network netmask gateway

#-----
# Masquerading:
#-----
MASQ_NETWORK='192.168.12.0/24' # networks to masquerade (e.g. our LAN)
MASQ_MODULE_N='1'              # load n masq modules (default: only ftp)
MASQ_MODULE_1='ftp'            # ftp
#MASQ_MODULE_2='h323'          # h323 (netmeeting)
#MASQ_MODULE_3='icq'           # icq (use with caution!)
#MASQ_MODULE_4='irc'           # irc
#MASQ_MODULE_5='raudio'        # raudio
#MASQ_MODULE_6='vdolive'       # vdolive
#MASQ_MODULE_7='quake'         # quake
#MASQ_MODULE_8='cuseeme'       # cuseeme
#MASQ_MODULE_9='mms'           # MSN-Filetransfer
#MASQ_MODULE_10='pptp'         # pptp
#MASQ_MODULE_11='ipsec'        # ipsec
#MASQ_MODULE_12='dplay'        # dplay (direct play)
#MASQ_MODULE_13='msn-0.02'     # msn zone (use version 0.01 or 0.02)
#MASQ_MODULE_14='udp_dloose'    # pseudo mod: some internet games need it

MASQ_FTP_PORT_N='0'          # using ftp masq-module on different ports
#MASQ_FTP_PORT_1='21'         # standard ftp port
#MASQ_FTP_PORT_2='2021'       # additional port (le cas de free.fr mais marche aussi sans)

#-----
# Optional package: PORTFW
#
# If you set OPT_PORTFW='yes', you can also edit opt/etc/portfw.sh
#-----
OPT_PORTFW='no'              # install port forwarding tools/modules
PORTFW_N='0'                 # how many portforwardings to set up
```

Fli4L

```
#PORTFW_1='8080 192.168.6.15:80 tcp' # sample 1: forward ext. port 8080 to int.  
# host 192.168.6.15 to port 80 (use tcp)  
#PORTFW_2='3000-3010 192.168.6.15 tcp' # sample 2: forward portrange to int. host  
# 192.168.5.15 (use tcp)
```

```
#-----
```

```
# Routing without masquerading
```

```
#-----
```

```
ROUTE_NETWORK="" # optional: route from/to network, no masq
```

```
#-----
```

```
# Routing: internal hosts to deny forwarding
```

```
#-----
```

```
FORWARD_DENY_HOST_N='0' # number of denied hosts
```

```
#FORWARD_DENY_HOST_1='192.168.6.5' # optional: 1st denied host
```

```
#FORWARD_DENY_HOST_2='192.168.6.6' # optional: 2nd denied host
```

```
#-----
```

```
# Routing: ports to reject/deny forwarding (from inside and outside!)
```

```
#-----
```

```
FORWARD_DENY_PORT_N='1' # no. of ports to reject/deny forwarding
```

```
FORWARD_DENY_PORT_1='137:139 REJECT' # deny/reject forwarding of netbios
```

```
FORWARD_TRUSTED_NETS="" # but allow forwarding between LANs
```

```
#-----
```

```
# Firewall: ports to reject/deny from outside (all served ports)
```

```
#
```

```
# here we leave two ports untouched:
```

```
#
```

```
# 53 dns
```

```
# 113 auth
```

```
#-----
```

```
FIREWALL_DENY_PORT_N='6' # no. of ports to reject/deny
```

```
FIREWALL_DENY_PORT_1='0:52 REJECT' # privileged ports: reject or deny
```

```
FIREWALL_DENY_PORT_2='54:112 REJECT' # privileged ports: reject or deny
```

```
FIREWALL_DENY_PORT_3='114:1023 REJECT' # privileged ports: reject or deny
```

```
FIREWALL_DENY_PORT_4='5000:5001 REJECT' # imond/telmond ports: reject or deny
```

```
FIREWALL_DENY_PORT_5='8000 REJECT' # proxy access: reject or deny
```

```
FIREWALL_DENY_PORT_6='20012 REJECT' # vbox server access: reject or deny
```

```
FIREWALL_DENY_ICMP='no' # deny icmp (ping): yes or no
```

```
FIREWALL_LOG='yes' # log access to rejected/denied ports
```

```
#-----
```

```
# Domain configuration:
```

```
#-----
```

```
START_DNS='yes' # start dns server: yes or no
```

```
DNS_FORWARDERS='212.27.32.5' # DNS servers of your provider, e.g. MSN (free.fr)
```

```
DNS_VERBOSE='no' # log queries in /usr/local/ens/ens.log
```

```
DOMAIN_NAME='mydomain.dd' # your domain name
```

```
DNS_FORBIDDEN_N='0' # number of forbidden domains
```

```
#DNS_FORBIDDEN_1='foo.bar' # 1st forbidden domain
```

Fli4L

```
#DNS_FORBIDDEN_2='bar.foo'      # 2nd forbidden domain
HOSTS_N='3'                     # number of hosts in your domain
HOST_1='192.168.12.1 fli4l'     # 1st host: ip and name, choice is yours
HOST_2='192.168.12.2 client2'   # 2nd host: ip and name
HOST_3='192.168.12.3 client3'   # 3rd host: ip and name
#HOST_4='192.168.12.4 client4'  # 4th host: ip and name

#-----
# Special DNS configuration
#-----
DNS_N='0'                       # number of special dns servers, normally 0
#DNS_1='firma.de 192.168.1.12'   # 1st special dns server for firma.de
#DNS_2='lan.firma.de 192.168.2.12' # 2nd special dns server for lan.firma.de

#-----
# imond configuration:
#-----
START_IMOND='yes'               # start imond: yes or no
IMOND_PORT='5000'               # TCP-Port, see also FIREWALL_DENY_PORT_x!
IMOND_PASS='zzzzz'              # imond-password, may be empty
IMOND_ADMIN_PASS='yyyyyyyyyy'   # imond-admin-password, may be empty
IMOND_LED=""                    # tty for led: com1 - com4 or empty
IMOND_BEEP='yes'                # beep if connection going up/down
IMOND_LOG='no'                  # log /var/log/imond.log: yes or no
IMOND_LOGDIR='/boot'            # log-directory, e.g. /var/log
IMOND_ENABLE='yes'              # accept "enable/disable" commands
IMOND_DIAL='yes'                 # accept "dial/hangup" commands
IMOND_ROUTE='yes'               # accept "route" command
IMOND_REBOOT='yes'              # accept "reboot" command

#-----
# Generic circuit configuration:
#-----
IP_DYN_ADDR='yes'               # use dyn. ip addresses (most providers do)(nécessaire pour
I(SDN)
DIALMODE='manual'               # standard dialmode: auto, manual, or off

#-----
# optional package: syslogd
#-----
OPT_SYSLOGD='no'                # start syslogd: yes or no
SYSLOGD_DEST_N='1'              # number of destinations
SYSLOGD_DEST_1='*.* /dev/console' # n'th prio & destination of syslog msgs
SYSLOGD_DEST_2='*.* @192.168.6.2' # example: loghost 192.168.6.2
SYSLOGD_DEST_3='kern.info /var/log/dial.log' # example: log infos

#-----
# optional package: klogd
#-----
OPT_KLOGD='no'                  # start klogd: yes or no

#-----
# optional package: y2k correction
```

Fli4L

```
#-----  
OPT_Y2K='no'          # y2k correction: yes or no  
Y2K_DAYS=""          # correct hardware Y2K-Bug: add x days
```

```
#-----  
# Optional package: PNP  
#-----  
OPT_PNP='yes'         # install isapnp tools: yes or no
```

POUR ISDN :

- Aller dans `~/isdn-routeur/fli4l-2.0.4/config/`.
- Editer le fichier *isdn.txt*.
- L'adapter à sa configuration et l'enregistrer.

Voici ma configuration pour la carte Gazel :

```
#-----  
# Optional package: ISDN  
#-----  
OPT_ISDN='yes'        # use ISDN: yes or no  
ISDN_TYPE='34'        # type, e.g. Gazel ISA. Voir liste  
ISDN_IO='0x240'       # io, e.g. 0x240 for Gazel ISA. Voir isapnp.conf !!!  
ISDN_IO0=""          # io0  
ISDN_IO1=""          # io1  
ISDN_MEM=""          # mem  
ISDN_IRQ='12'        # irq, e.g. 12 for Gazel ISA (mon cas)  
ISDN_DEBUG_LEVEL='31' # debug level (hisax driver)  
ISDN_VERBOSE_LEVEL='2' # verbose level
```

```
#-----  
# ISDN compression (EXPERIMENTAL):  
#-----  
OPT_ISDN_COMP='no'    # use LZS or BSD compression: 'yes' or 'no'  
ISDN_LZS_DEBUG='1'    # debug level lzscmp (0..3)  
ISDN_LZS_COMP='8'     # compression level lzscmp (0..9)  
ISDN_LZS_TWEAK='7'    # tweak lzscmpr (at present: 0..7)
```

```
#-----  
# ISDN Circuits:  
#-----  
ISDN_CIRCUITS_N='1'   # no. of ISDN circuits, DSL: see pppoe.txt
```

```
#-----  
# Circuit 1: Internet-By-Call-Provider MSN  
#-----  
ISDN_CIRC_1_NAME='mydomain.dd' # circuit MSN  
ISDN_CIRC_1_USEPEERDNS='no'    # use dns server of your provider: yes or no  
ISDN_CIRC_1_TYPE='ppp'        # circuit uses sync ppp (ipppd)  
ISDN_CIRC_1_BUNDLING='no'     # channel bundling: yes or no  
ISDN_CIRC_1_BANDWIDTH=""      # if bundling=yes: opt. bandwidth on demand  
ISDN_CIRC_1_LOCAL=""          # local ip address of your ISDN card - dummy  
ISDN_CIRC_1_REMOTE=""         # remote ip address (ISDN) - dummy
```

Fli4L

```
ISDN_CIRC_1_NETMASK='255.255.255.0' # netmask (ISDN) - dummy
ISDN_CIRC_1_MTU='1500'             # max transmission unit (use 1500 if comp!)
ISDN_CIRC_1_MRU='1524'             # maximum receive unit
ISDN_CIRC_1_COMPRESSION='no'       # compress headers/frames, see docum.
ISDN_CIRC_1_FRAMECOMP='default'    # type of framecompression, see docum.
ISDN_CIRC_1_IPX_NETWORK=""         # optional: ipx network
ISDN_CIRC_1_IPX_NODE=""            # optional: ipx nodes local:remote
ISDN_CIRC_1_REMOTENAME='aaaaa;bbb' # optional: remote hostname for dialin
ISDN_CIRC_1_USER='my.id.login'      # User-ID to login into provider's gateway
ISDN_CIRC_1_PASS='my-password'     # Password for login
ISDN_CIRC_1_ROUTE='0.0.0.0'        # this is the default route
ISDN_CIRC_1_DIALOUT='0860922000'   # dialout: ISDN number of provider
ISDN_CIRC_1_DIALIN=""              # dialin: no dialin
ISDN_CIRC_1_CALLBACK='off'         # callback mode: 'in', 'out', or 'off'
ISDN_CIRC_1_CBDELAY='3'             # callback delay, only used for callbacks
ISDN_CIRC_1_EAZ='5255'             # your MSN (without area code)
ISDN_CIRC_1_SLAVE_EAZ=""           # optional slave MSN for channel bundling
ISDN_CIRC_1_DEBUG='yes'            # enable ippdp debugging, 'yes' or 'no'
ISDN_CIRC_1_AUTH=""               # require the peer to authenticate itself
ISDN_CIRC_1_HUP_TIMEOUT='175'      # Hangup after 175 seconds idle time
ISDN_CIRC_1_CHARGEINT='180'        # Value of charge interval (in seconds)
ISDN_CIRC_1_TIMES='Mo-Su:00-24:0.0148:Y' # times/charges when LCR
```

```
#-----
# Circuit 2: bidirectional connection to another router with raw tcpip
# not activated yet, only example for non-default-circuit
#-----
```

```
#-----
# telmond configuration:
#-----
OPT_TELMOND='yes'                 # start telmond: yes or no
TELMOND_PORT='5001'               # TCP-Port, see also FIREWALL_DENY_PORT_x!
TELMOND_LOG='no'                  # log /var/log/telmond.log: yes or no
TELMOND_LOGDIR='/var/log'         # log-directory, e.g. /var/log
TELMOND_MSN_N='0'                 # number of msn->ip mapping entries
TELMOND_MSN_1='123 192.168.6.2'   # e.g. map MSN 123 to client 192.168.6.2
TELMOND_CMD_N='0'                 # no. of commands to be executed if call-in
TELMOND_CMD_1='123 * sleep 5; imonc dial' # e.g. dial out via default circuit
```

Et maintenant construisons la disquette !

- Aller dans ~/isdn-routeur/fli4l-2.0.4 et introduire une disquette vierge dans le lecteur.
- Formater la disquette : **fdformat /dev/fd0u1680** (de préférence, choisissez une disquette de bonne qualité pour utiliser ce format).
- Pour plus de sécurité : **./mkclean.sh**
- Compresser les fichiers nécessaires : **./mktgz.sh**
- Copier les fichiers sur la disquette : **./mkfloppy.sh -h** (le paramètre -h est nécessaire pour les disquette de 1680 ko)

Puisque j'ai choisi une numérotation manuelle sous X-Window, construisons le programme gtk-imonc !

Remarque : pour les mordus du clavier, vous pouvez utiliser une application en console :

imonc, à télécharger sur le site de fli4l.de).

- Aller dans ~/gtk-imonc-0.1.6.
- Exécuter **./configure**.
- Exécuter **./make**.
- Exécuter **su** afin de basculer sous le compte root.
- Exécuter **./make install** afin d'installer le programme *gtk-imonc* dans /usr/local/bin.

Vous pouvez par la suite repérer le fichier *gtk-imonc.desktop* et le copier ou le glisser à la souris vers le Bureau.

Voilà ! C'est aussi simple que cela.

Ne pas oublier si ce n'est pas déjà fait, de saisir les commandes suivantes (en root) :

ifconfig eth0 192.168.12.x netmask 255.255.255.0 up

route add default gw 192.168.12.1 eth0

Pour autant que votre réseau est le 192.168.12.0 !!

Elles réinitialisent la carte réseau avec l'adresse IP correspondante au réseau sur lequel le routeur ISDN officie et ajustent la passerelle par défaut.

Remplacez *192.168.12.x* par l'adresse IP du poste de travail.

La minute de vérité !

Introduisons la disquette dans l'ordinateur qui doit fonctionner en tant que routeur et mettons-le en route.

...syslinux.....

...kernel.....

Messages d'erreur ??? Impossible ;-)

Login :

Saisissez votre login, puis votre mot de passe.

ATTENTION, le clavier est en qwerty-US. Choisissez donc un mot de passe qui reste facile à mémoriser et qui ne fait pas appel aux lettres A, Q, Z, W et M.

De retour à votre poste de travail, lancez gtk-imonc. Cliquez sur le bouton DIAL. Un témoin de connexion s'affiche dans le bas de la fenêtre.

Have fun !

PS : toute la documentation officielle est dans le répertoire ~/isdn-routeur/fli4l-2.0.4, malheureusement en Allemand. Une liste de diffusion existe mais uniquement en Allemand !

Je conseille de faire une deuxième disquette et, dans le fichier base.txt, de changer le paramètre : (pour le cas où :))

MOUNT_BOOT='ro' # mount boot device (floppy): ro, rw, no

Divers :

A voir sur : <http://www-public.tu.bs.de:8080/~y0011440>.

Voici une 'one-floppy' passerelle construite par Torben Baras. Il l'a nommé '*Silent Router*'. Devinez pourquoi ;-)



Construit à partir d'une carte mère 486-DX2 Intel 66 Mhz CPU dans un rack de 19" et fli4l.
