Le HOWTO sur le Courrier Electronique

Guylhem Aznar `guylhem at danmark.linux.eu.org` v2.0 (Adaptation française par Gacquer Frédéric `gacquer at neuronnexion.fr`) 15 juin 1998 V 0.9 (Puis relu par ...) Janvier 1998

Ce document décrit l'installation, la maintenance et l'utilisation du courrier électronique sous linux. Ce document vous sera utile si vous envisagez de communiquer localement ou vers des sites distants via le courrier électronique. Vous n'aurez vraisemblablement *pas* besoin de lire ce document si vous n'échangez pas de courriel avec d'autres utilisateurs, sur votre système ou avec d'autres sites.

Table des matières

1 Introduction, copyright et avertissement (Disclaimer)

1.1 Le courrier électronique et le courrier non sollicité (spamming)

Premièrement, convertir tous les "at" dans les courrier électroniques par des "@".

C'est plus simple pour les humains, mais pas pour les automates indexant le web pour spammer; De plus c'est suffisant pour empêcher les généreux collaborateurs d'être spammés !

1.2 Objectif

Le but de ce document est de répondre à quelques questions et commentaires et correspond à la définition de la "Foire Aux Questions", sur les logiciels de courriel sous Linux en général, et la version des distributions Linux Debian et Redhat en particulier.

1.3 Nouvelles versions

Les nouvelles versions de ce document seront régulièrement postées sur `comp.os.linux.announce`, `comp.answers` et `mail.answers`. Ils seront ajoutés dans plusieurs sites ftp anonymes qui archivent en particulier ces documents, notamment `sunsite.unc.edu:/pub/Linux/docs/HOWTO`.

De plus, vous pourrez en général trouver ce document à la page d'accueil de `Linux WorldWideWeb` à `<http://sunsite.unc.edu/mdw/linux.html>`.

1.4 Vos impressions

Vos impressions, par courriel, concernant ce document m'intéressent, qu'elles soient positives ou négatives. Précisément, contactez moi si vous trouvez des erreurs ou d'évidentes omissions.

Je lis, mais ne réponds pas forcément, à tous les courriels que je reçois. Les demandes d'amélioration seront examinées et prises en compte selon la formule du temps disponible, l'intéret de la demande et ma pression artérielle du jour :-)

Les remarques désobligeantes, courriels incendiaires (flames) iront gentimment dans `/dev/null` donc pas de problème.

Les remarques concernant le format en cours doivent être adressées au coordinateur du HOWTO : Greg Hankins (gregh at sunsite.unc.edu).

1.5 Copyright

Le HOWTO du Mail est sous copyright (c) 1998 Guylhem Aznar. Il est distribué sous la licence de copyright LDP. Pour toutes questions, veuillez contacter Guylhem Aznar, le coordinateur du Linux HOWTO, à gregh at sunsite.unc.edu.

1.6 Garantie limitée

Bien sûr, je désavoue tout responsabilité pour le contenu de ce document. L'utilisation des concepts, exemples, et/ou autre contenu de ce document est entièrement à votre risque.

2 Autres sources d'information

2.1 USENET

Il n'y a rien de "particulier" pour configurer et utiliser les logiciels de courrier sous linux (plus maintenant). Donc, vous n'aurez certainement *PAS* à poster des questions relatives au courrier électronique dans le forum de discussion générique `comp.os.linux.*`.

Ne postez pas dans la hiérarchie de `comp.os.linux` sauf si c'est réellement lié à linux, comme par exemple : "Avec quelles options ont été compilé le sendmail de la Debian 1.2 ?" ou "Le smail de la Redhat 5.0 se plante quand je l'exécute".

Laisser moi le redire.

Dorénavant il n'y a virtuellement aucune raison de poster quoi que ce soit, de relatif au courriel, dans la hiérarchie `comp.os.linux`. Il existe des forums de discussion dans la hiérarchie `comp.mail.*` qui répondront à *TOUTES* vos questions.

SI VOUS POSTEZ DES QUESTIONS QUI NE SONT PAS SPECIFIQUES A LINUX DANS COMP.OS.LINUX.*, VOUS CHERCHEZ DE L'AIDE AU MAUVAIS ENDROIT. LES EXPERTS DU MEL SE NICHENT DANS LES ENDROITS INDIQUES CI-DESSUS ET EN GENERAL N'UTILISENT PAS LINUX.

POSTER DES QUESTIONS NON RELATIVES A LINUX DANS LA HIERARCHIE LINUX GACHE VOTRE TEMPS ET CELUI DES AUTRES ET ALLONGE FREQUEMENT LE DELAI DE REPONSE A VOTRE QUESTION.

LES BONS ENDROITS sont :

- comp.mail.elm Le logiciel de courrier électronique ELM.
- comp.mail.mh Le logiciel Rand Message Handling.
- comp.mail.mime Multipurpose Internet Mail Extensions.
- comp.mail.misc Discussions générales à propos du courrier électronique.
- comp.mail.multi-media Le courrier électronique multimédia.
- comp.mail.mush Le Mail User's Shell (MUSH).
- comp.mail.sendmail L'agent sendmail de BSD.
- comp.mail.smail L'agent de courrier électronique smail.
- comp.mail.uucp Le courrier électronique dans un environnement uucp.

2.2 Les listes de diffusion

Il existe plusieurs listes de diffusion sendmail, smail et qmail.

Vous trouverez les adresses dans `/usr/doc/celuiquevousavezchoisi`.

2.3 Les autres documents de LDP.

Il y a plusieurs éléments essentiels fournis dans les autres documents HOWTO de linux et du projet DOC de linux.

En particulier, vous pouvez jetez un oeil à :

- Sur votre propre système dans `/usr/doc/smail` ou `/usr/doc/sendmail` :-)
- Le Guide d'Administration Réseau de Linux.
- Le HOWTO Communication Série
- Le HOWTO Ethernet
- Le HOWTO UUCP si vous êtes alimenté via UUCP.

2.4 Livres

Ce qui suit est une liste non exhaustive qui pourra vous aider :

- "Managing UUCP and USENET" de O'Reilly and Associates est selon moi le meilleur ouvrage disponible pour résoudre les programmes et les protocoles impliqués dans un site USENET.

- "Unix Communications" de 'The Waite Group' contient bonne une description de tous les éléments (et plus) et comment ils s'imbriquent.
- "Sendmail" de O'Reilly and Associates semble être la référence la plus complète sur Sendmail-v8 et sendmail+IDA. C'est un incontournable pour quiconque souhaite comprendre sendmail sans perdre son calme. Il existe en version française sous le titre "Sendmail seconde édition" (NDT).
- "The Internet Complete Reference" d'Osborne est livre de référence détaillé qui explique les services variés disponible sur Internet et est une excellent source d'information sur les forums de discussion, le courrier électronique et divers autres ressources Internet.
- "The Linux Networking Administrators' Guide" d'Olaf Kirch du Projet de Documentation Linux (LDP) est disponible sur le net. Il est publié (au moins) par O'Reilly et SSC. Existe en version française sous le titre "Guide d'administration réseau sous linux" (NDT).

3 Configuration requise

3.1 Matériel

Il n'y a pas de configuration matérielle spécifique pour le courriel sous Linux.

Vous aurez besoin d'un quelconque logiciel de 'transport' pour vous connecter à des systèmes distants, c'est-à-dire soit TCP/IP, soit uucp.

Cela peut signifier aussi que vous aurez besoin d'un modem ou d'une carte ethernet, suivant votre installation.

Dans la plupart des cas, vous pourrez souhaiter le modem le plus rapide possible, soit aujourd'hui 57600 bps. En général, vous voudrez une UART 16550 sur votre carte série ou implémentée dans votre modem pour gérer les vitesses supérieures à 9600 baud.

Si vous ne voyez ce que signifie cette dernière phrase, veuillez consultez le groupe `comp.dcom.modems` ou les différentes excellentes FAQ (Foire Aux Questions) sur les modems et les communications séries. Ainsi que les courriels postés sur USENET.

3.2 Logiciels

Bon, le problème est là. Que pouvons nous faire avec les logiciels de courrier électronique ?

- Nombre de sites Alimenter à plus de 100 sites avec des options compliquées pour les noms de domaines ? Prendre sendmail !

  Alimenter à moins de 100 sites sans avoir à trop jouer avec les noms de domaines ? Prendre smail !
- Sécurité Alimenter à plus ou moins 100 sites mais avec une sécurité élevée ? Prendre qmail !

  Alimenter moins de 100 sites mais avec un niveau normal de sécurité ? Prendre smail !
- Différentes manières de recevoir du courrier électronique

Alimenter et être alimenté par UUCP ou FIDO (via ifmail) ? Alimenter ou être alimenté par POP et le SMTP internet ? Prendre smail !

Bien sûr, vous êtes libre de choisir n'importe quel logiciel de courrier électronique, les informations que vous trouverez ici ne sont là que pour vous aidez dans votre choix. Sendmail est parfait pour beaucoup de sites avec des options compliquées, qmail est parfait pour une sécurité élevée ; l'intermédiaire entre sendmail et qmail est smail, le meilleur compromis. Si vous savez ce que vous faites, choisissez sendmail (et vous ne devriez pas lire un HOWTO !) ; sinon Je recommende en général smail.

4 Smail v3.1

Smail3.1 semble un standard de fait comme agent de transport pour des sites n'utilisant que uucp, ainsi que pour quelques sites smtp. Il est facile à configurer, compile sans avoir à patcher les sources et il est assez sûr.

4.1 Configurer smail

Installez les binaires smail à partir de votre distribution (je vous recommande cette façon) ou récupérez les sources de smail et compilez le. Si vous compilez smail à partir des sources, vous devrez avoir ce qui suit dans votre fichier `os/linux` pour que 'sed' vous donne des scripts qui fonctionnent correctement.

```
CASE_NO_NEWLINES=true
```

Une fois configuré, le fichier `config` ira certainement dans `/etc/smail` (mais votre configuration peut être différente si vous utilisez d'anciennes distributions) ; Editons les !

4.2 Le fichier `config`

```
# From
smart_path=polux
smart_transport=uux

# To
hostname=danmark
domains=linux.eu.org

visible_name=danmark.linux.eu.org
uucp_name=danmark.linux.eu.org

# max_message_size=512k
# auth_domains=foo.bar
# more_hostnames=barberouge:barberouge.polux.freenix.fr
```

Bon, tout d'abord, qui vous alimente? "polux" m'alimente via uucp (c'est-à-dire par transport uux); naturellement vous aurez besoin de changer ce fichier selon votre propre organisation. Par exemple, "bargw.bar.foobar.com" pourrait vous alimenter via "smtp", dans ce cas vous n'avez pas besoin de fichier de transport et pouvez définir "-transport_file" pour indiquer que vous n'en avez pas besoin.

Vous pouvez aussi utiliser "postmaster_address = votrenom", cacher la topologie du réseau pour les adresses sortantes (si vous êtes une passerelle) en utilisant "visible_name", positionner quelle adresse d'alias pourra aussi être utilisée pour le courriel que vous recevez, en utilisant "more_hostnames".

Lire la documentation smail pour plus de détails ou les exemples dans `/usr/doc/smail/examples`pour voir si l'un d'eux correspond à votre situation.

4.3 Le fichier `directors`

```
# aliasinclude - développe les adresses ":include:filename" produite par les fichiers
# alias.
# Cette entrée et la suivante sont du réchauffé. Les raisons
# d'un changement significatif sont mineures. Leur unique raison est
```

```
# de reconnaitre et de développer les adresses de la forme :
#       :include:pathname
# qui peuvent apparaitre dans les fichiers alias ou les fichiers de listes de
# diffusions/faire suivre (produite par toute directive dont le pilote (driver)
# est forwardfile).
aliasinclude:
        driver = aliasinclude,          # utiliser ce pilote pour le cas spécial
        nobody;                         # associer l'utilisateur nobody avec les adresses
                                        #  si violation de permissions moyenne (???)
                                        # sont rencontrées
        copysecure,                     # prendre les permissions du directeur alias
        copyowners,                     # prendre le droit propriétaire du directeur alias

# forwardinclude - développe les adresses ":include:filename" produites
# par les fichiers forward
forwardinclude:
        driver = forwardinclude,        # utiliser ce pilote pour cas spécial
        nobody;
        copysecure,                     # prendre les permissions du directeur forwarding
        copyowners,                     # prendre le droit propriétaire du directeur
                                        #  forwarding

# aliases - cherche à développer les alias stockés dans la base de donnée
# C'est le fichier aliases standard. Il est utilisé pour les affaires courantes,
# comme mapper root, postmaster, MAILER-DAEMON et uucp aux administrateurs de site,
# créer quelques petits systèmes de développement d'alias, ou autres. Dans
# cette configuration de site, le fichier aliases est utilisé principalement
# pour des informations d'aliases/forwarding liés aux machines. Les informations générales
# pour faire suivre le courrier doivent être mises dans la base de donnée "forward".
aliases:
        driver=aliasfile,               # directeur d'alias généraliste
        -nobody,                        # toutes les adresses sont associés à nobody
                                        # par défaut, donc mettre l'option
                                        # n'est pas trés utile.
        sender_okay,                    # ne pas enlever l'expéditeur lors des
                                        # développements d'adresses.
        owner=owner-$user;              # les problèmes vont à une adresse propriétaire
        file=/etc/aliases,
        modemask=002,                   # l'accès en écriture ne doit pas être autorisé à tous
        optional,                       # ignore si le fichier n'existe pas
        proto=lsearch,                  # fichier ASCII non trié.

# forward - chercher à développer selon le contenu de la base forwarding
# C'est la base de forwarding utilisateur pour un grand sous-domaine.
# Les entrées sont gardées ici pour les utilisateurs en cours ou passé,
# pour faire suivre leur courrier à leur machine préférée. La base forward
# est installée sur la couche réseau TCP/IP lors des changements, pour
# garder le réseau consistent.
#forward:
#       driver = aliasfile,             # Directeur d'alias généraliste
#       -nobody,                        # toutes les adresses sont associés à nobody
#                                       # par défaut, donc mettre l'option
#                                       # n'est pas trés utile.
#       owner = real-$user;             # les problèmes vont à une adresse propriétaire
#
```

```
#       file = /etc/forward,
#       modemask = 002,
#       proto = dbm,                    # Utiliser la librairie dbm(3X) pour les accès.

# dotforward - développe les fichiers .forward dans les répertoires personnels
#  des utilisateurs.
# Pour les utilisateurs qui ont une entrée dans la base "forward", un fichier
# .forward n'est utilisé que sur la machine personnelle, comme identifiée
# dans la base forward. Si c'est utilisé, c'est traité comme une liste d'adresse
# indiquant à qui doit être envoyé les courriels, plutôt (ou en plus) qu'à
# l'utilisateur référencé dans l'adresse locale.
dotforward:
        driver = forwardfile,           # le directeur généraliste forwarding
        owner = postmaster, nobody, sender_okay;

        file = ~/.forward,              # Le fichier .forward dans le répertoire personnel
        checkowner,                     # l'utilisateur peut être propriétaire de ce fichier
        owners = root,                  # ou bien root
        modemask = 002,                 # Ne doit pas être accessible à tous en écriture
        caution = daemon:root,          # ne rien exécuter en tant que root ou daemon
        # être trés attentif aux accès à distance des répertoires utilisateurs (?)
        unsecure = "~uucp:/tmp:/usr/tmp:/var/tmp"

# forwardto - développe le "Forward to " dans les fichiers boîtes aux lettres utilisateur
# Cela émule le méchanisme du suivi de courrier V6/V7/System-V qui utilise
# une ligne des adresses forward au début des fichiers boites aux lettres (BAL) utilisateur
# avec comme préfixe la chaîne "Forward to "
forwardto:
        driver = forwardfile,
        owner = postmaster, nobody, sender_okay;

        file = /var/spool/mail/${lc:user},      # pointe sur la BAL utilisateur
        forwardto,                      # autorise la fonction "Forward to "
        checkowner,                     # l'utilisateur peut être propriétaire de ce fichier
        owners = root,                  # ou root
        modemask = 0002,                # sous System V, le groupe mail à la permission écrire
        caution = daemon:root           # ne rien exécuter sous root ou daemon

# user - apparie un utilisateur sur l'hôte local et délivre le courrier
# dans leur boîte aux lettres (BAL).
user:   driver = user;                  # pilote pour apparier les utilisateur
        transport = local               # le transport local va dans les BALs

# real_user - apparie les utilisateurs préfixés avec la chaîne "real-"
# Trés utile pour permettre à une adresse de délivrer expllicitement dans
# une BAL utilisateur. Par exemple, les erreurs dans le développement d'un fichier
#.forward pourra être délivrer ici, ou des boucles de suivi de courrier entre
# plusieurs machines peuvent être résolues en utilisant une adresse real-utilisateur.
# Aussi, les utilisateurs ui souhaite utiliser le courriel comme moyen de transfert de
# données vers une machine qui n'est pas leur machine personnelle peuvent envoyer un
# courriel à real-nom-de-login@hôte-distant.
real_user:
        driver = user;
        transport = local,
        prefix = "real-"                # par exemple, apparie real-root
```

```
# lists - développe les listes de diffusion stockées dans le répertoire list
# Les listes de diffusions peuvent être crées simplement en créant un fichier
# dans le répertoire /etc/smail/lists.
lists:  driver = forwardfile,
        caution,                        # transmet toutes les adresses avec précaution
        nobody,                         # puis associe l'utilisateur nobody
        owner = owner-$user;            # Les sites system V peuvent vouloir utiliser
                                        # o-$user, puisque owner-$user peut être
                                        # trop long pour un fichier de 14 caractères.
        file = lists/${lc:user}         # Les listes sont sous $smail_lib_dir

# owners - développe les listes de diffusion stockées dans le répertoire list owner.
# Les listes de diffusion listes propriétaires peuvent être simplement crées en
# créant un fichier dans le répertoire /etc/smail/lists/propriétaire. Les listes de
# diffusion propriétaires sont envoyées localement, les erreurs générées s'arrangent
# avec la liste de diffusion du même nom. Pour créer une liste propriétaire pour une
# liste de diffusion,  créer un fichier avec le nom de la liste dans
# /etc/smail/lists/propriétaire. Cela va créer une adresse de liste de nom
# propriétaire-nom_de_liste, comme utilisée par le directeur "lists" plus haut.
owners: driver = forwardfile,
        caution,                        # transmet toutes les adresses avec précaution
        nobody,                         # puis associe l'utilisateur nobody
        owner = postmaster;             # Les sites system V peuvent vouloir utiliser
                                        # o-$user, puisque owner-$user peut être
                                        # trop long pour un fichier de 14 caractères.
        prefix = "owner-",
        file = lists/owner/${lc:user}   # Les listes sont sous $smail_lib_dir

# request - développe les listes de diffusions stockées dans un répertoire de liste
# de requête. Les listes de diffusion listes de requête peuvent être crées simplement
# en créant un fichier dans le répertoire /etc/smail/lists/request. Les adresses requête
# sont utilisées typiquement comme adresse standard pour les interrogations à propos
# d'une liste de diffusion. Par exemple, les requêtes pour s'abonner ou se désabonner
# d'une liste seront généralement envoyées à "list-request", qui devra être configurée
# pour faire suivre le courrier à la ou les personnes appropriées.
request: driver = forwardfile,
        caution,                        # transmet toutes les adresses avec précaution
        nobody,                         # puis associe l'utilisateur nobody
        owner = postmaster;             # Les sites system V peuvent vouloir utiliser
                                        # o-$user, puisque owner-$user peut être
                                        # trop long pour un fichier de 14 caractères.
        suffix = "-request",
        file = lists/request/${lc:user} # les listes sont sous $smail_lib_dir
```

4.4 Le fichier `fidopaths`

Créez un tel fichier seulement si vous utilisez ifmail et FIDO.

4.5 Le fichier `routers`

```
# forces - forcer certains chemins d'accès
  # L'existence de cette base de donnée est un moyen de coder en dur les chemins des
  # diverses machines ou domaines. Utilisé pour créer des distorsions temporaires
```

```
# des autres bases de données de routage (?). Pour changer la base, éditer le
# fichier maps/force.path puis faire un "make" dans le sous-répertoire maps/
forces:
        driver = pathalias,             # route au fichier de recherche de chemins
        method = /etc/smail/maps/table; # Les parties transports sont dans ce fichier
        file = forcepaths,              # fichier d'information contenant le forçage de
                                        #  chemins
        proto = lsearch,                # utilise le fichier de chemin trié
        optional,
        reopen                          # ferme quand pas utilisé

uucp_neighbors:
        driver=uuname,                  # utilise un programme renvoyant les voisins
        transport=uux;
        cmd="/usr/bin/uuname -a",       # spécifiquement, utilise le programme uuname
#        domain=uucp                    # enlève le suffixe ".uucp"

# smart_host - un directeur spécifique particulier d'hôte intelligent
# Si l'attribut smart_path du fichier de configuration est défini comme
# chemin entre l'hôte local et l'hôte distant, alors les noms d'hôtes
# qui ne seraient pas reconnus sinon seront envoyées au ?. L'attribut
# du fichier de configuration smart_transport peut être utilisé pour
# indiquer un transport différent.
# Si l'attribut smart_path n'est pas spécifié, ce routeur (?) est ignoré.
smart_host:
        driver = smarthost,             # Pilote du cas spécial (?)
        transport = uux                 # Par défaut livre via UUCP
#       path=phreak

# ifmail - pour envoyer des courriels à fidonet et vice versa.
ifmail:
        driver=pathalias,
        transport=ifmail;
        file=fidopaths,
        proto=lsearch
```

Vous ne devez include la partie ifmail que si vous l'utilisez pour les courriels FIDO. A noter, vous pouvez aussi changer le mode de transport de "uux" (c'est-à-dire UUCP) pour, par exemple, "smtp" ou même coder en dur les chemins vers différentes machines ou domaines dans `/etc/smail/maps/table`.

4.6 Le fichier `transports`

```
# local - délivre le courriel aux utilisateurs locaux
# Indique à smail d'ajouter directement dans les fichiers BAL dans le répertoire
# /var/spool/mail
#local: driver = appendfile,            # ajoute le message à un fichier
#       -return_path,                   # ajoute le champ  Return-Path:
#       local,                          # utiliser la forme locale pour la livraison
#       from,                           # Ajoute la ligne d'enveloppe From_
#       unix_from_hack;                 # insère > avant From dans le corps
#
#       file = /var/spool/mail/${lc:user},      # Utiliser cet emplacement pour Linux
#                                               # A noter, le spool courriel doit être 1777
#       file = ~/mailfile,      # Utiliser cet emplacement pour une sécurité accrue
```

```
#       group = mail,            # groupe pour être propriétaire des fichiers pour System V
#       mode = 0660,             # sous System V, le groupe mail doit pouvoir accéder
#       suffix = "\n",           # ajouter un code nouvelle ligne supplémentaire
#       append_as_user,

# Cela permet à chaque utilisateur d'avoir un fichier ~/.procmailrc pour contr^oler
# le filtrage du courriel et sauver le courriel des listes de diffusions dans des BAL
# séparées s'ils le souhaitent
local:  +inet,
        -uucp,
        driver = pipe,                  # ajoute un message à un fichier
        return_path,                    # inclue un champ Return-Path:
        local,                          # utilise la forme locale pour la livraison
        from,                           # Ajoute la ligne d'enveloppe From_
        unix_from_hack;                 # insère > avant From dans le corps

        cmd = "/usr/bin/procmail",      # utilise procmail pour délivrer le courrier local
        parent_env,                     # environment info from parent addr
        pipe_as_user,                   # utilise le user-id associé avec l'adresse
        umask = 0022,                   # umask pour le processus fils
#       -ignore_status,                 # On devrait croire au code de retour
#       -ignore_write_errors,           # réessayer si un tube est cassé

# pipe - délivre le courriel à des commandes shells.
# Cela est utilisé implicitement quand smail rencontre des adresses commençant
# avec un caractère barre verticale, comme "|/usr/lib/news/recnews talk.bizarre".
# La barre verticale est enlevée de l'adresse avant d'être transmise au transport.
#pipe:  driver = pipe,                  # tuber le message à un autre programme
#       return_path, local, from, unix_from_hack;
#
#       cmd = "/bin/sh -c $user",       # envoyer l'adresse au Bourne Shell
#       parent_env,                     # les info d'environnement de l'adresse parent
#       pipe_as_user,                   # utilise le user-id associé avec l'adresse
#       umask = 0022,                   # umask pour le processus fils
#       -log_output,                    # ne pas enregistrer stdout/stderr dans les logs
#       ignore_status,                  # le code de retour peux être faux, l'ignorer
#       ignore_write_errors,            # ignorer les tubes cassées.

# file - délivrer le courrier dans des fichiers
# Utilisé implicitement lorsque smail rencontre des adresses qui commencent
# par un slash (/) ou un caractère unique, comme  "/usr/info/list_messages"
# ou peut être "~/Mail/inbox".
#file:  driver = appendfile,
#       return_path, local, from, unix_from_hack;
#
#       file = $user,                   # le fichier est pris dans l'adresse
#       append_as_user,                 # utilise le user-id associé avec l'adresse
#       expand_user,                    # développe ~ et $ dans l'adresse
#       check_path,
#       suffix = "\n",
#       mode = 0644
# uux - deliver to the rmail program on a remote UUCP site
#
# Dans une transaction UUCP on peut délivrer à l'hôte distant jusqu'à
# cinq adresses destinataires
```

```
uux:    driver = pipe,
        -uucp,
        inet,
#       uucp,                         # utilise la forme d'adresse style UUCP
        from,                         # fourni une ligne d'enveloppe From_
        max_addrs = 5,                # au plus 5 adresses par invocation
        max_chars = 200;              # au plus 200 caractères d'adresses
# le drapeau -r la livraison immédiate, des parenthèses autour de la variable
# $user emp^eche l'interprétation des caractères spéciaux par uux
        cmd = "/usr/bin/uux - -r -g$grade $host!rmail $((${strip:user})$)",
#        cmd="/usr/bin/uux - $host!rmail $(($user)$)",
        ignore_write_errors,          # ignore les tubes cassés.
        umask = 0022,
#       pipe_as_sender,

# uux_one_addr - délivre le courrier par UUCP à un hôte distant qui ne peut
#               prendre qu'une adresse à la fois.
# Cela est souvent nécessaire quand on livre le courrier à un site qui
# tourne sous une version non modifiée de 4.1BSD.
uux_one_addr:
        driver = pipe,
        uucp,                         # utilise le style UUCP pour les formes d'adressage
        from;                         # fourni une ligne d'enveloppe From_
# le drapeau -r empêche une livraison immédiate
        cmd = "/usr/bin/uux - -r -g$grade $host!rmail (${strip:user})",
        umask = 0022,
        pipe_as_sender

queueonly:
        driver = pipe;                # envoi de message à un tube
        cmd = "/usr/lib/sendmail -Q -f $sender -bm $user",
                                      # utilise getmail pour les livraisons locales
        user=root,                    # exécute getmail sous "root"
        group=mail,                   # exécute getmail sous "mail"
        parent_env,                   # Les infos d'environement de l'adresse parent
        -pipe_as_user,                # utiliser le user-id associé avec l'adresse
        umask = 0007,                 # umask du processus fils

# pour delivrer un message. Le transport smtp est inclue seulement si
# le réseau BSD existe.
# L'attribut uucp peut être positionné pour le transfert dans une zone uucp.
# L'attribut inet doit être positionné pour les transferts vers l'internet.
# NOTE: c'est presque optimal, un backend ??? devrait exister qui prendrait en charge
#       plusieurs messages par connection.
# AUSSI: il peut être nécessaire de restreindre max_addrs à 100, puisqu'il
#  s'agit de la limite inférieure dont a besoin l'implémentation SMTP pour
# prendre en charge un message.

smtp:   driver=tcpsmtp,
        inet,                         # si UUCP_ZONE n'est pas défini
#       uucp,                         # si UUCP_ZONE est défini
        -max_addrs, -max_chars;       # pas de limite sur le nombre d'adresses

        short_timeout=5m,             # le timeout pour les opérations courtes
        long_timeout=2h,              # le timeout pour les opérations SMTP plus longues
```

```
        service=smtp,                   # se connecter à ce port de service
# Pour l'utilisation internet : ne pas commenter les 4 lignes qui suivent
       use_bind,                        # résoud les enregistrements MX et A multiple A
       defnames,                        # utiliser la recherche de domaine standard
       defer_no_connect,                # essayer à nouveau si le serveur est tombé
       local_mx_okay,                   # évite un MX vers l'hôte local

ifmail:
        from,received,max_addrs=5,max_chars=200,
        driver=pipe;
        pipe_as_sender,
        cmd="/usr/local/bin/ifmail -x9 -r$host $((${strip:user})$)"
```

Vous ne devez inclure la partie ifmail que si vous l'utilisez pour les courriels FIDO. A part cela, vous ne devriez pas avoir besoin d'éditer quoi que ce soit dans ce fichier définissant les agents de transport (comme uux, smtp ...) vous pouvez les utiliser comme paramètres dans l'autre fichier `config`.

A noter que j'ai commenté quelques parties, comme les "pipes" ou "file", pour améliorer la sécurité.

4.7 Le répertoire `maps/`

Il contient les fichiers `map` et `tables` :

Commençons par le fichier `map`

```
#N      foo.bar foo2.bar2
#S      AT 486/RedHat Linux 1.2.13
#O      organization
#C      contact
#E      administration (email)
#T      phone
#P      address
#R
#U      hosts connected via uucp
#W      created/edited by
#
hname polux

hname linux.eu.org

hname = polux
hname = polux.linux.eu.org
```

Encore une fois, éditez ce fichier pour correspondre à votre organisation (je suis alimenté par `polux.linux.eu.org`).

Maintenant le fichier `table`

```
*       uux
```

Vous pouvez définir différent transport pour différents chemins, par exemple "smtp" pour vos machines dans votre réseau local, "uux" (c'est-à-dire uucp) pour le reste du monde ou vice-versa (j'utilise uucp pour tous courriers électroniques sortant, et donc j'utilise "*" !).

4.8 D'autres bons exemples

Les fichiers précédents sont ceux que j'utilise actuellement pour mon site, vous ne devriez rencontrer aucun problème en les utilisant comme exemples/bases pour vos propres fichiers.

Les fichiers ci-dessous sont seulement fournis en tant que bons exemples pour configurer smail de différentes façons.

```
#ident "@(#) transports,v 1.2 1990/10/24 05:20:46 tron Exp"

  # Voir smail(5) pour une description complète du contenu de ce fichier

  # local - délivre le courrier aux utilisateurs nouveaux
  #
  # Indiquer à smail d'ajouter directement aux fichiers BAL utilisateur dans le
  # répertoire /usr/mail.
  local:  driver = appendfile,            # ajouter le message au fichier
          return_path,                    # inclue le champ Return-Path: field
          local,                          # utiliser les formes locales pour la livraison
          from,                           # fourni une ligne enveloppe From_
          unix_from_hack;                 # insère > avant From dans le corps

          file = /usr/mail/${lc:user},    # utiliser cet emplacement pour System V
          group = mail,                   # groupe propriétaire de fichier pour System V
          mode = 0660,                    # sous System V, le groupe mail a accès
          suffix = "\n",                  # ajoute à nouveau un retour chariot supplémentaire
          append_as_user,

  # pipe - délivre du courrier à des commandes shell
  #
  # Ceci est utilisé implicitement lorsque smail rencontre des adresses qui
  # commencent par une barre verticale, comme "|/usr/lib/news/recnews talk.bizarre".
  # La barre verticale est enlevée de l'adresse avant d'être donnée au
  # transport

  pipe:   driver = pipe,                  # tube le message à un autre programme
          return_path, local, from, unix_from_hack;

          cmd = "/bin/sh -c $user",       # envoi l'adresse au Bourne Shell
          parent_env,                     # info d'environement de l'adresse parent
          pipe_as_user,                   # utilise le user-id associé avec l'adresse
          umask = 0022,                   # umask du processus fils
          -log_output,                    # ne pas enregistrer stdout/stderr
          ignore_status,                  # le code de retour n'est pas fiable, l'ignorer
          ignore_write_errors,            # ignorer les tubes cassés

  # file - délivre du courrier dans des fichiers
  #
  # Ceci est utilisé implicitement quand smail rencontre des adresses qui commencent
  # avec un caractère slach ou tilde, comme "/usr/info/list_messages" ou aussi
  # "~/Mail/inbox".
  file:   driver = appendfile,
          return_path, local, from, unix_from_hack;

          file = $user,                   # le nom de fichier est pris dans l'adresse
```

```
        append_as_user,                 # le user-id associé avec l'adresse
        expand_user,                    # développe ~ et $ dans l'adresse
        suffix = "\n",
        mode = 0644

# uux - délivre au programme rmail pour un site UUCP distant
#
# Dans une transaction UUCP on peut délivrer à l'hôte distant jusqu'à
# cinq adresses destinataires
uux:    driver = pipe,
        uucp,                           # utilise le style UUCP comme forme d'adressage
        from,                           # fourni une ligne d'enveloppe From_
        max_addrs = 5,                  # au plus 5 adresse par invocation
        max_chars = 200;                # au plus 200 caractères pour l'adresse

        # le drapeau -r empeche une livraison immédiat, les parenthèses autour
        # de la variable $user empeche l'interprétation des caractères spéciaux par uux.
        cmd = "/usr/bin/uux - -r -g$grade $host!rmail $((${strip:user})$)",
        umask = 0022,
        pipe_as_sender
# uux_one_addr - délivre le courrier à travers UUCP à un hote distant qui prends
# une adresse à la fois
#
# Cela est souvent nécessaire lorsque l'on délivre du courrier à un site utilisant
# une version non modifiée de 4.1BSD.
uux_one_addr:
        driver = pipe,
        uucp,                           # utilise le style  UUCP comme forme d'adressage
        from;                           # fourni une ligne d'enveloppe From_

        # le drapeau -r empeche la livraison immédiate
        cmd = "/usr/bin/uux - -r -g$grade $host!rmail (${strip:user})",
        umask = 0022, pipe_as_sender

# demand - délivre à un programme rmail distant, polling on demand
demand: driver = pipe,
        uucp, from, max_addrs = 5, max_chars = 200;

        # sans le drapeau -r, essaye de contacter le site distant immédaitement
        cmd = "/usr/bin/uux - -g$grade $host!rmail $(($user)$)",
        umask = 0022, pipe_as_sender

# uusmtp - délivre à un programme rsmtp vers un site UUCP distant
#
# Deliver utilise un protocole de lot de commandes simple vers la machine distante
# Cela permet d'utiliser beaucoup plus d'adresses arbitraires.
# lève les limites sur les adresses destinataires par invocation de uux.
uusmtp: driver = pipe,
        bsmtp,                          # envoi des lots de commande SMTP
        -max_addrs,                     # il n'y a pas de limite sur le nombre
        -max_chars;                     # et la taille totale des adresses destinataires

  # mettre -r pour que la livraison ne soit pas immédiate, les adresses destinataires
  # sont stockées dans les données envoyées à l'entrée standard de smtp.
        cmd = "/usr/bin/uux - -r -g$grade $host!rsmtp",
```

```
        umask = 0022, pipe_as_sender

# demand_uusmtp - délivre à un programme rsmtp distant, polling on demand
demand_uusmtp:
        driver = pipe,
        bsmtp, -max_addrs, -max_chars;

        # sans le drapeau -r, essaye de contacter le site distant immédiatement
        cmd = "/usr/bin/uux - -g$grade $host!rsmtp",
        umask = 0022, pipe_as_sender

# smtp - délivre le courrier en utilisant SMTP sur TCP/IP
#
# Se connecter à un hote distant en utilisant TCP/IP et initie une conversation SMTP
# pour delivrer un message. Le transport smtp est inclue seulement si le réseau BSD existe.


# NOTE: il peut etre nécessaire de restreindre max_addrs à 100, puisque c'est la limite
#       inférieure requise pour SMTP pour prendre en charge un message.
smtp:   driver = smtp,
        -max_addrs,
        -max_chars

#ident "@(#) table,v 1.2 1990/10/24 05:20:31 tron Exp"

# Ce fichier indique les transports utilisés pour livrer le courrier
# à des hotes spécifiques à partir de bargw.

#host           transport
#--------       ---------
curdsgw         demand_uusmtp   # deliver utilisant le SMTP par lots
oldbsd          uux_one_addr    # Les sites 4.1BSD ne peuvent prendre plus d'une adresse
sun             demand          # appelle sun lorsqu'il y a du courrier à envoyer
*               uux             # pour tous les autres, poll at intervals
```

4.9 Relancer inetd

Pour exécuter smail en tant que démon smtp, ajoutez une des lignes suivantes dans `/etc/inetd.conf`:

```
smtp stream tcp nowait  root  /usr/bin/smtpd smtpd
```

ou :

```
smtp stream tcp nowait  root  /usr/sbin/tcpd  /usr/sbin/in.smtpd
```

Le courrier électronique sortant est envoyé automatiquement, quand vous utilisez elm.

4.10 Smail et smtp

Généralement, les fournisseurs d'accès (ISP) utilisent smtp, par conséquent vous ne devriez pas avoir de problème pour envoyer vos courriels. Si votre ligne internet est hors service lors de l'envoi de courriel, dans ce cas les courriels résident dans `/var/spool/smail/input`. Lorsque la ligne est à nouveau opérationnelle, runq est démarré et envoi les courriels en attente. Toutefois, la réception de courriel est le problème puisque votre fournisseur d'accès a plusieurs clients à gérer, et pas seulement vous !

Habituellement, vous pouvez récupérer vos courriels via le protocole POP, voir la section POP ci-dessous.

5 Sendmail+IDA

Pour les sites d'importance, sendmail vaut la peine d'être choisi, en raison de son incroyable facilité d'utilisation, mais vous devez vous décider entre sendmail+IDA et sendmail 8.x :

- Si vous utilisez un noyau ancien (1.0) : sendmail+IDA
- Si vous utilisez un noyau pas si ancien (1.2) : sendmail+IDA et édition du code source.
- Pour un noyau récent (2.0), le choix sera sendmail 8.x

Rappelez-vous, les débutants en linux ou les personnes concernées par la sécurité/facilité de la configuration devraient plutôt essayer d'utiliser smail ou qmail, qui sont plus facile à utiliser et plus sûr.

5.1 Installation du source

- `cd / ; tar -zxvf sendmail5.67b+IDA1.5.tpz`
- cd vers `/usr/local/lib/mail/CF` et copier le fichier `sample.m4` vers "votre_nom_d'hôte.m4". Editez la répartition de hostname, aliases, smarthost et en répartir la bonne dans votre site. Le fichier par défaut est pour un site n'utilisant qu'uucp qui a des entêtes de domaine et qui communique avec l'hôte intelligent (Smart host) . Puis faire "make votre_nom-d'hôte.cf" et mettre le fichier résultant dans `/etc/sendmail.cf`.
- Si vous n'utilisez que uucp, vous n'avez *AUCUN* besoin de créer une des tables mentionnées dans le fichier `README.linux`. Vous n'aurez qu'à faire un `touch` des fichiers pour que le `Makefile` fonctionne. Editez simplement le fichier `.m4`, `make sendmail.cf` et vous pouvez le tester.
- Si vous n'utilisez que uucp et que vous communiquez avec des sites en plus de votre "smart-host", vous aurez à ajouter des entrées uucpxtable pour chacun (ou leur envoyer un courriel passera par l'hôte intelligent) et exécuter dbm sur le fichier uucpxtable ainsi modifié.
- Si vous utilisez la distribution binaire 5.67a originale de Rich Braun, vous aurez à geler la configuration si vous changez votre fichier .cf avec "/usr/lib/sendmail -bz" pour que les changements soient pris en compte.

Vous devrez aussi mettre à jour votre version en au moins la 5.67b puisqu'il y a un vilain trou de sécurité dans la 5.67a et les précédentes. Une autre amélioration est que si mail.debug est indiqué et que vous utilisez syslogd, votre courriel entrant et sortant sera tracé. Voir le fichier `/etc/syslog.conf` pour les détails.

Les sources de sendmail+IDA se trouvent sur `<vixen.cso.uiuc.edu>` ; Ils ne nécessitent aucun patch pour fonctionner sous linux si vous êtes sous quelque chose comme le noyau 1.00.

Si vous avez un noyau > 1.1.50, vous aurez le plaisir d'appliquer la plupart des patch linux qui sont maintenant dans les sources vanilla. (Je vous *avez* dit que ce sendmail était seulement pour les noyaux anciens :-)

Il est particulièrement évident de savoir où cela doit être fait : taper simplement `make` et lorsque cela plante, aller à la ligne dans le source et commenter le code Linux à cet endroit.

Si vous avez l'intention d'utiliser sendmail+IDA, je vous recommande fortement de prendre la version sendmail5.67b+IDA1.5 puisque tous les patch Linux sont maintenant dans les sources vanilla et plusieurs trous de sécurité qui ETAIENT (!!!) dans les anciennes versions que vous auriez récupérées ou compilées avant le premier décembre 1993 ont été comblés.

Aujourd'hui le noyau est le 2.0, vous devriez utiliser sendmail 8.x au lieu de sendmail+IDA, mais je vous avez déjà dit de choisir plutôt sendmail 8.x :-)

5.2 Le fichier `sendmail.m4`

Sendmail+IDA requiert de créer le fichier `sendmail.m4` plutôt que d'éditer directement le fichier `sendmail.cf`. Le bon coté des choses est qu'il est plus simple de mettre en place des configurations de courrier électronique qui sont particulièrement difficile (pour ne pas dire impossible à la plupart des gens de le mettre en place correctement) avec smail ou le traditionnel sendmail.

Le fichier `sendmail.m4` qui correspond aux exemples ci-dessus de smail ressemble à :

```
dnl #------------------ SAMPLE SENDMAIL.M4 FILE ------------------
dnl #
dnl # (la chaine 'dnl' est l'équivalent m4 pour commenter une ligne)
dnl #
dnl # En général on ne cherche pas à outrepasser le chemin compilé en dur de LIBDIR
dnl #define(LIBDIR,/usr/local/lib/mail)dnl    # où tous les fcihiers support vont
define(LOCAL_MAILER_DEF, mailers.linux)dnl    # mailer pour la livraison locale
define(POSTMASTERBOUNCE)dnl                   # postmaster récupère les courriers rebonds
define(PSEUDODOMAINS, BITNET UUCP)dnl         # ne pas utiliser le DNS sur eux
dnl #
dnl #---------------------------------------------------------------
dnl #
dnl # les noms par lesquels nous sommes connus
define(PSEUDONYMS, myhostname.subdomain.domain myhostname.UUCP)
dnl #
dnl # notre nom primaire
define(HOSTNAME, myhostname.subdomain.domain)
dnl #
dnl # notre nom uucp
define(UUCPNAME, myhostname)dnl
dnl #
dnl #---------------------------------------------------------------
dnl #
define(UUCPNODES, |uuname|sort|uniq)dnl       # notre voisin uucp
define(BANGIMPLIESUUCP)dnl                    # s'assurer que le courriel uucp
define(BANGONLYUUCP)dnl                       #  est traité correctement
define(RELAY_HOST, my_uucp_neighbor)dnl       # notre hôte relais intelligent
define(RELAY_MAILER, UUCP-A)dnl               # nous atteignons moria via uucp
dnl #
dnl #--------------------------------------------------------------------
dnl #
dnl # les différentes tables de recherche dbm
dnl #
define(ALIASES, LIBDIR/aliases)dnl            # alias systèmes
define(DOMAINTABLE, LIBDIR/domaintable)dnl    # mettre le domaine aux hôtes
define(PATHTABLE, LIBDIR/pathtable)dnl        # chemins base de données
define(GENERICFROM, LIBDIR/generics)dnl       # adresses From génériques
```

```
define(MAILERTABLE, LIBDIR/mailertable)dnl    # mailers par hôte ou domaine
define(UUCPXTABLE, LIBDIR/uucpxtable)dnl      # chemins vers les hôtes que nous fournissons
define(UUCPRELAYS, LIBDIR/uucprelays)dnl      # chemins court-circuits
dnl #
dnl #--------------------------------------------------------------------
dnl #
dnl # inclue le code 'réel' qui fait tout fonctionner
dnl # (fourni avec le code source)
dnl #
include(Sendmail.mc)dnl                         # ENTREE REQUISE !
dnl #
dnl #------------ END OF SAMPLE SENDMAIL.M4 FILE -------
```

5.3 Définir un agent de courriel local

A la différence de la plupart des distributions Unix, Linux n'a pas par défaut un agent pour délivrer le courriel local. Dorénavent, deliver ou procmail est généralement installé, ainsi aucune difficulté ne sera ajoutée à cette installation déjà suffisemment complexe.Je vous conseille d'utiliser les programmes deliver ou procmail couremment disponible, qui peuvent être des progiciels optionnels sur quelques distributions linux.

Afin d'y arriver, vous devez définir LOCAL_MAILER_DEF dans le fichier `sendmail.m4` dont le contenu ressemble à :

```
# -- /usr/local/lib/mail/mailers.linux --
#     (local mailers for use on Linux )
Mlocal, P=/usr/bin/deliver, F=SlsmFDMP, S=10, R=25/10, A=deliver $u
Mprog,  P=/bin/sh,       F=lsDFMeuP,   S=10, R=10, A=sh -c $u
```

Il y a aussi une définition par défaut pour deliver dans le fichier `Sendmail.mc` qui sera inclue dans le fichier `sendmail.cf`. Pour l'indiquer, vous n'utiliserez pas le fichier `mailers.linux` mais définirez plutôt ce qui suit dans le fichier `sendmail.m4`:

```
dnl --- (in sendmail.m4) ---
define(LOCAL_MAILER_DEF, DELIVER)dnl       # mailer for local delivery
```

Malheureusement, `sendmail.mc` suppose que deliver est installé dans /bin, ce qui n'est pas le cas dans la Slackware1.1.1 (qui l'installe dans /usr/bin). Dans ce cas vous aurez soit à le tromper avec un lien ou recompiler deliver à partir des sources pour qu'il réside dans /bin. Veuillez noter que procmail est en général meilleur que deliver, par exemple pour le filtrage de courriel.

5.4 Les tables dbm de sendmail+IDA

Mettre en place un comportement particulier, pour les sites ou les domaines, est effectué à travers un nombre de tables dbm optionnelles plutôt que d'éditer le fichier `sendmail.cf` directement.

Consultez l'édition Juillet 1994 du Linux Journal (si vous pouvez encore le trouver ! :-), pour les documentations dans les sources, ou le chapitre sur sendmail dans la dernière version du Guide d'Administration Réseau du Projet de Documentation Linux (LDP) qui sera disponible vraiment-bientôt-maintenant pour plus de détails.

- mailertable - défini les comportements particuliers pour les sites distants ou les domaines.
- uucpxtable - force uucp à délivrer le courriel aux hôtes qui sont au format DNS

– pathtable - défini les chemins BANG??? UUCP pour les sites distants ou les domaines.

– uucprelays - court circuite le chemin pathalias vers les hôtes distants connus

– genericfrom - converti les adresses internes en adresses génériques visible pour le monde extérieur.

– xaliases - converti les adresses génériques en/à partir d'adresses internes valides.

– decnetxtable - converti les adresses RFC-822 en adresses style DECnet.

5.5 Bon quelles entrées sont vraiment nécessaires?

Lorsque vous n'utilisez aucune des tables optionnelles dbm, sendmail délivre le courrier via le RELAY_HOST et RELAY_MAILER définis dans le fichier `sendmail.m4` utilisé pour générer `sendmail.cf`. Il est aisé de passer outre ce comportement à travers des entrées dans domaintable ou uucpxtable.

Un site générique sur Internet et utilisant le Service de Noms de Domaines (DNS), ou un site n'utilisant qu'UUCP et transmets tous les courriels via UUCP à travers un smart RELAY_HOST, n'aura probablement pas besoin d'entrées spécifiques du tout.

De fait, tous les systèmes devraient utiliser les macros DEFAULT_HOST et PSEUDONYMS, qui définissent le nom canonique et l'alias du site par lequel il se fait connaitre.

Si tout ce que vous avez est un hôte relais et un agent relais de courrier, vous n'avez pas à prendre les options par défaut puisque cela marche automagiquement. Les hôtes UUCP auront certainement besoin de mettre leur nom officiel UUCP dans UUCPNAME.

Ils auront probablement aussi besoin d'utiliser RELAY_MAILER et RELAY_HOST qui permettent le routage smart-host (?) à travers un relais de courriel.

Le logiciel de transport (l'agent transporteur) de courriel à utiliser est indiqué dans RELAY_MAILER et devrait habituellement être UUCP-A pour les sites UUCP. Si votre site n'a que SMTP et utilise le 'Service de Nom de Domaine' (DNS), vous voudrez changer RELAY_MAILER.

Si votre site utilise SLIP, vous pourriez choisir la manière simple et juste transmettre tous vos courriel sortant vers votre fournisseur d'accès et le laisser s'en occuper. Ainsi vous devrez définir votre domaine dans ISOLATED_DOMAINS et VALIDATION_DOMAINS, vous devrez définir votre fournisseur d'accès dans RELAY_HOST et RELAY_MAILER à TCP. Bien sûr, vous voulez demander la permission avant de monter un système comme relais général.

5.6 Sendmail 8.x

Le Sendmail 8.7.x de Berkeley a été la dernière version majeure depuis sendmail5. Il incorpore un merveilleux support interne (?) pour compiler sous linux : faire simplement "make linux" et tout fonctionne.

Vous serez probablement mieux servi en récupérant une des nombreuses distributions binaires de l'habituel site d'archives Linux plutôt que de vous battre avec des choses comme les dbm de Berkeley.

Il y a une distibution sympa de sendmail 8.6.12 de Jason Haar - `j.haar at lazerjem.demon.co.uk` sur `sunsite.unc.edu` à `/pub/Linux/system/Mail/delivery/sendmail-8.6.12-bin.tgz` qui incorpore la documentation et une description rapide sur comment l'utiliser avec sendmail v8 pour les configurations courantes.

Le plus important avec sendmail v8 c'est que vous devez configurer au moins le minimum pour que cela fonctionne ; ce qui suit est un exemple qui devrait au moins vous en rapprocher.

5.7 Un fichier mc d'exemple pour la 8.7.x

De même que sendmail+IDA, sendmail v8 utilise m4 pour compiler un fichier de configuration en un sendmail.cf complet utilisable par sendmail. Ce qui suit est mon fichier courant pour mon site (ppp vers internet pour le courriel sortant, uucp pour le courriel entrant).

```
dnl divert(-1)
 #--------------------------------------------------------------------
 #
 # Fichier .mc pour un hôte linux qui est configuré comme suit:
 #
 #       - connecté à Internet pour le courrier sortant (ici ppp)
 #       - connecté via UUCP pour le mail entrant
 #       - entête avec domaine
 #       - pas d'agent de transport de mail local (utilise plutot 'deliver')
 #       - pas de DNS lancé donc ne pas canoniser ce qui sort via DNS
 #       - all non-local outbound mail goes to the RELAY_HOST over smtp
 #       -
 #      (nous utilisons ppp et laissons notre fournisseur d'accès faire le travail)
 #
 #                                    vds 3/31/95
 #
 #--------------------------------------------------------------------
 include(`../m4/cf.m4')
 VERSIONID(`linux nodns relays to slip service provider smarthost')dnl
 Cwmyhostname.myprimary.domain myhostname.UUCP localhost
 OSTYPE(linux)
 FEATURE(nodns)dnl
 FEATURE(always_add_domain)dnl
 FEATURE(redirect)
 FEATURE(nocanonify)
 dnl MAILER(local)dnl
 MAILER(smtp)dnl
 MAILER(uucp)dnl
 define(`RELAY_HOST', smtp:my.relay.host.domain)
 define(`SMART_HOST', smtp:my.relay.host.domain)
 define(`UUCP_RELAY', smtp:my.relay.host.domain)
 define(`LOCAL_MAILER_PATH', `/bin/deliver')
 define(`LOCAL_MAILER_ARGS', `deliver $u')
```

5.8 Les .... de Sendmail v8

Je suppose qu'il y a quelques différences pour les 'bigots IDA' parmi nous.

Au lieu de 'runq', taper 'sendmail -q' pour traiter la file d'attente !

5.9 Les agents distributeurs locaux

A la différence de la plupart des systèmes d'exploitation, Linux n'avait pas de courriel intégré : vous deviez avoir un programme pour distribuer le courrier local, comme "lmail", "procmail" ou "deliver".

Toutefois, toutes les distributions récentes inclues un distributeur local !

La documentation sur l'utilisation aussi bien pour la distribution locale est dans la révision binaire de sendmail5.67b+IDA1.5 (sur sunsite) citée plus haut.

6 Le courriel POP

Cette section concerne aussi IMAP, pas trés différent de POP.

6.1 Historique

Sur un réseau de station de travail, le courriel a toujours été un problème :

- Soit vous utilisez ""user@computer.foo.com" avec des soucis lorsque "computer" est hors service, faire connaitre votre réseau aux personnes extérieures, avoir différentes adresses pour une personne passant à une autre machine, ...
- Soit vous installez un centre de tri (mail hub), "mailhost.foo.com" avec des règles de réécritures, ainsi tous les utilisateurs semblent faire leur envoi de la même adresse, même s'ils sont sur des ordinateurs différents.

Mais dans ce cas, comment l'utilisateur peut-il lire son courrier ? Utiliser rsh avec elm ? :-) Cela surchargerai notre centre de tri (mail hub) ! Une méthode était retransmettre (forwarding) ou UUCP, smtp, etc. mais c'est trop compliqué.

Alors vint POP/IMAP, chacun avec des problèmes de sécurité au début, (maintenant résolus en utilisant ssh sur les nouvelles versions) :

6.2 Récupérer son courrier

Ici se trouve un des principaux désavantages de POP : le mot de passe est envoyé en texte clair sur le réseau, et quelques agents utilisateurs ne connaissent simplement pas POP : vous devez utiliser un lecteur de courrier connaissant POP, comme Pine, Emacs, Netscape, Mutt ...

Les problèmes de mot de passe peuvent être résolus en créant un "cannal" crypté que POP utilisera ou en utilisant les extensions de APOP ou RPOP. Le problème du lecteur de courriel peut être resolu soit en changeant d'agent utilisateur (mozilla est prêt pour POP, de même Emacs ou Pine) ou en utilisant un "récupérateur de courriel" POP associé à un agent distributeur local.

gwpop (a Good Way to POP, une bonne façon de faire du POP) vaut le coup puisqu'il crée un "cannal" crypté et dépose le courrier directement dans le "spool". Toutefois, il repose sur Perl ...

Je peux aussi recommander fetchmail, qui est activement supporté.

Sinon vous pouvez utilisez un des nombreux client pop disponibles pour linux, par exemple si votre compte est paul et votre mot de passe AssezSecret, vous exécuterez :

```
$ popclient -3 -v mail.acme.net -u paul -p "AssezSecret" -k -o MEL-INTERNET-PAUL
```

(voir les pages du manuel pour la signification exacte des différentes options ci-dessus !)

6.3 Envoyer du courrier

Pour cela, vous devez utiliser un logiciel de courrier connaissant SMTP, comme qmail, sendmail, smail ou mozilla (celui là fait tout : lecteur de mail, récupération par POP, envois par smtp !)

Se référer à l'une des sections prédédentes pour installer et configurer celui que vous préférez le plus. Alors, vous serez au stade du "Test", essayez d'envoyer quelques courriers à un compte local sur le centre de tri (mail hub).

6.4 Lire son courriel

Si votre programme ne fait pas tout lui même, vous pouvez installer elm, pgp, mush, pine ...beaucoup de bons programmes sont disponible gratuitement pour les plateformes linux !

6.5 Tests

Pour vérifier que votre serveur de courriel a pop, essayez :

```
$ telnet mailhost 110
```

Si cela fonctionne, vous aurez quelque chose comme "OK Pop server (...) starting" : tapez "quit" !

Pour installer un "cannal" crypté ssh, testez tout d'abord votre serveur de courriel en tapant :

```
$ ssh mailhost date
```

Si vous avez la date, cela devrait être bon. Remarquez que ssh ne demandera pas de mot de passe, donc vous devez créer un fichier ".shosts" sur le serveur de mail, contenant le nom du client. Pour tester la redirection du port ssh (que gwpop utilise), tapez :

```
$ ssh -n -f -L 12314:localhost:110 mailhost sleep 30
```

puis

```
$ telnet localhost 12314
```

Alors vous aurez bon espoir de voir la bannière du centre de tri pop. Si vous n'utilisez pas ssh, n'oubliez pas de mettre en commentaire $ssh dans le script gwpop. Pour vérifier si procmail est en cours, essayez "procmail -v".

6.6 Utilisation

Maintenant vous pouvez éditer le script Perl de gwpop pour vérifier que tout va bien, puis exécuter gwpop :

```
$ gwpop -v votre-nom-utilisateur
POP password on mailhost: votre-mot-de-passe-secret
```

Si les "messages d'erreurs" de gwpop sont normaux, le courriel du centre de tri (mail hub) sera téléchargé vers votre machine locale à l'endroit indiqué à gwpop. (S'il vous plait testez avec quelques Courriels !).

Vous pouvez aussi utilisez gwpop comme démon :

```
$ gwpop -d $HOME/tmp votre-nom-utilisateur
```

Les messages de gwpop sont envoyés à syslog et gwpop s'exécute sans fin ; un signal "HUP" obligera gwpop à chercher vos Courriels.

Vous pourrez trouvez les logiciels de POP mentionnés ici :

- <ftp://ftp.pasteur.fr/pub/Network/gwpop>
- <ftp://ftp.informatik.rwth-aachen.de/pub/packages/procmail>
- <http://www.cs.hut.fi/ssh/>

7 Les "agents utilisateurs" de Courriel

Cettes section contiens les informations relatives aux "agents utilisateurs", c'est-à-dire les logiciels que l'utilisateur voit et utilise. Ces logiciels reposent sur les "agents transporteurs" mentionnés plus haut. Beaucoup d'autre "Agents Utilisateurs" de Courriel sont disponibles maintenant (pine, mush ...) mais je n'ai pas trouvé d'information dedans concernant linux. Me dire SVP si j'ai oublié quelque chose !

7.1 Elm

Elm se compile, s'installe et s'exécute sans faute sous linux. Pour plus d'informations, voir les sources d'elm et les instructions d'installation. Elm et les filtres doivent avoir les permissions 2755 (groupe mail) avec les permissions 775 et le groupe mail pour `/var/spool/mail`.

Si vous utilisez une distribution binaire, vous devrez créer un fichier `/usr/local/lib/elm/elm.rc` pour outrepasser les informations précompilées de hostname et de domain :

- remplacer "sousdomaine.domaine" avec votre nom de domaine
- "monnomdhôte" avec le nom canonique du nom d'hôte

```
#---------- /usr/local/lib/elm/elm.rc ------------------
#
# Nom d'hôte non qualifié
hostname = monnomdhôte
#
# domaine local
hostdomain = sousdomaine.domaine
#
# Nom d'hôte entièrement qualifié
hostfullname = monnomdhôte.sousdomaine.domaine
#
#-------------------------------------------------------
```

Une chose dont vous voudriez avoir conscience est que si vous utilisez Elm compilé avec l'option MIME, vous aurez besoin de metamail installé et dans votre chemin ou alors Elm ne sera pas capable de lire le courrier MIME reçu. Metamail est disponible à `<thumper.bellcore.com>` et bien sûr via "archie".

Dans la catégorie "trop beau pour être vrai", il y a une distribution, Elm-2.4.24, avec "PGP intégré". Pour l'essayer, récupérer le fichier `<ftp://ftp.viewlogic.com/pub/elm-2.4pl24pgp3.tar.gz>`, qui est l'elm2.4.24 avec l'option PGP ajoutée. Vous le configurez et le compilez de la même manière que l'Elm normal, ce qui veut dire que vous devrez ajouter les patchs mentionnés plus haut. Pour ce que cela vaut, je l'utilise ici et l'apprécie beaucoup. Bien sûr, il doit y avoir des versions disponibles plus récentes , y compris elm-ME+.

Bien que cet élément ne soit pas spécifique à linux, il est néanmoins perçu (de façon erronée) comme un bug pénible d'Elm. Nous avons tous entendu parler d'Elm qui, de temps en temps, échoue avec un message indiquant qu'il est incapable d'allouer un nombre important d'octets avec malloc(). La solution pour contourner ce problème est d'enlever les alias de Courriel post-générés (aliases.dir et aliases.pag).

CE N'EST PAS UN BUG DANS ELM, c'est une erreur dans la configuration d'Elm de l'endroit où a été récupéré la distribution binaire.

Elm a un format amélioré et non compatible d'aliases ; vous devez vous assurez que le chemin qu'utilise Elm pour aliases est différent de celui que sendmail/smail utilise. Vu le nombre de remarques sur ce problème, il est évident qu'au moins une distribution majeure 'dans la rue' a été mal configurée (de scot at catzen.gun.de (Scot W. Stevenson)).

Le progiciel metamail courant requiert csh pour quelques uns de ses scripts. Csh manquant (ou tcsh) causera des erreurs intéressantes ...

7.2 Mailx

Epargnez vous des ennuis : aller simplement chercher le kit mailx de la Slackware 2.1.0 ou supérieure, qui a la bonne implémentation de mailx5.5. Si vous voulez partir des sources, mailx v5.5 se compile sans avoir à patcher sous linux, si vous avez "pmake" installé. Si quelqu'un l'utilise encore, je l'engage fermement à enlever les trucs anciens d'"edmail" de la SLS1.00 et la remplacer par mailx.

7.3 D'autres agents utilisateurs

Ce qui suit est connu pour fonctionner sous linux. Consulter "archie" pour connaitre la façon de les trouver ...

- Pine - de l'Université de Washington
- Metamail - autorise l'utilisation MIME
- mh - une autre façon de gérer le Courriel
- deliver - fichier/gestion des Courriels basé sur des règles
- procmail - fichier/gestion des Courriels basé sur des règles
- Majordomo - gère les listes de diffusion
- Mserv - permet les fichiers par Courriel

8 Remerciements

Les personnes qui suivent m'ont aidé au rassemblement de l'information et des expériences qui ont permis la réalisation de ce document :

Steve Robbins, Ian Kluft, Rich Braun, Ian Jackson, Syd Weinstein, Ralf Sauther, Martin White, Matt Welsh, Ralph Sims, Phil Hughes, Scot Stevenson, Neil Parker, Stephane Bortzmayer et en particulier de grands merci à Vince Shankan pour sa participation importante.

Si j'oublie quelqu'un, toutes mes excuses : simplement Courrielez moi !