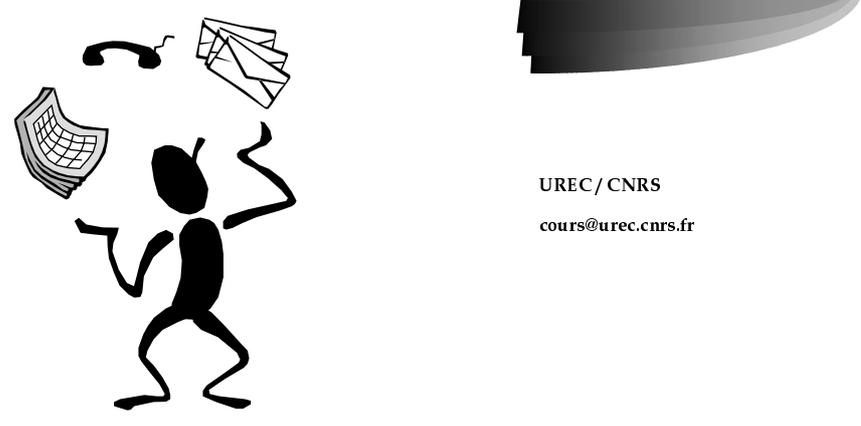




# Messagerie de l'Internet



UREC / CNRS  
[cours@urec.cnrs.fr](mailto:cours@urec.cnrs.fr)



## *Contributions*

- Création : Jean Paul Gauthier 1996
- Modifications :
  - 1997 Jean Paul Gauthier
  - 1998 Philippe Leca
  - 03/1999 Philippe Leca

Cour réalisé à partir :  
d'un cours de Pierre David ( Université de St Quentin en Yvelines)  
d'un tutorial d'Alain Durand (IMAG)

## *Programme sendmail*

- Conçu en 1982 par Eric Allman
- *le* routeur de courrier sur les systèmes Unix
  - Berkeley.
  - Système V
- souple, puissant
- abscons

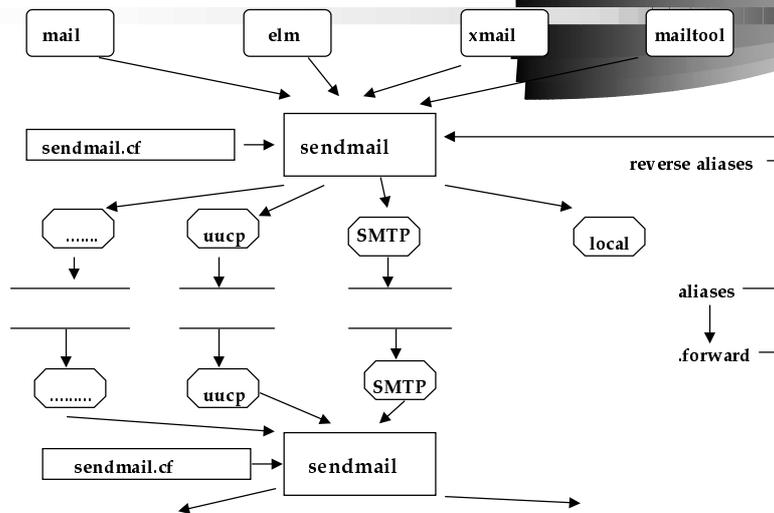
## *PLAN*

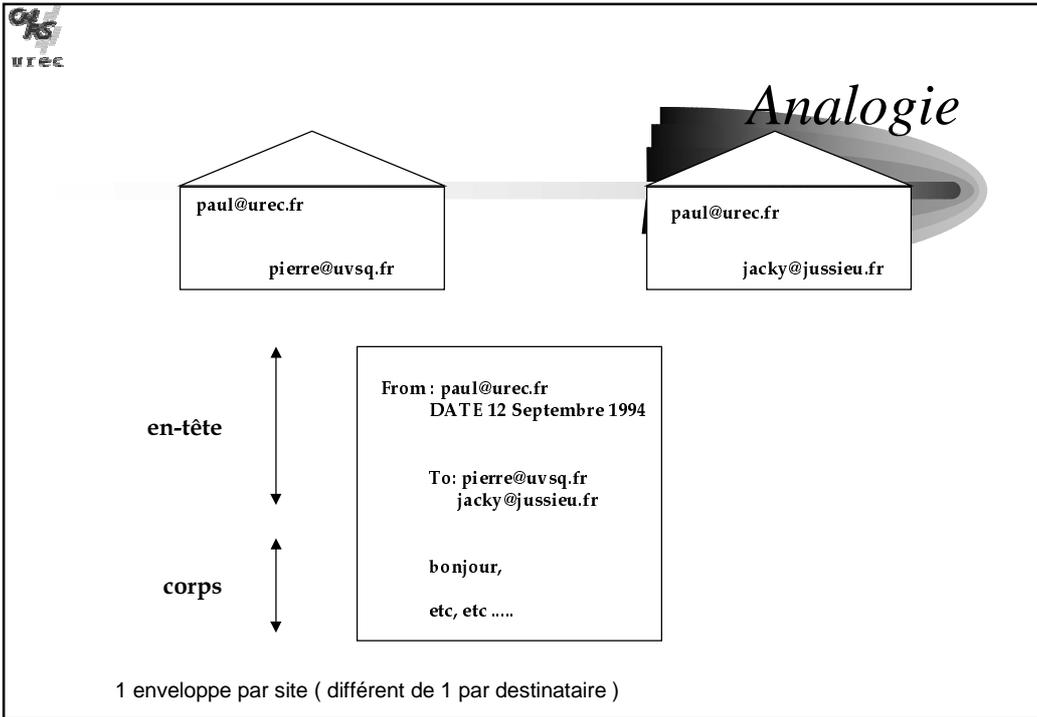
- Présentation
- Format des messages, SMTP,  
Interaction avec le DNS
- MIME
- sendmail
  - interface
  - aliases
  - Fichier de configuration
  - Tests

## Présentation : Terminologie

- agent d'utilisateur (*user agent : interface utilisateur*).
  - mail, elm, mailtool, xmh, ...
  - eudora, outlook, MS Mail, Netscape messenger, ...
- agent de transfert de messages (*message transfert agent*)
  - sendmail (Unix et NT) , IMS (NT) , Exchange (NT), Eudora Worldmail server (NT), ...
- agent de transport de messages (*mailer*)
  - SMTP, UUCP, ...
- remise (*local delivery*)
  - mail, procmail, ...
- mémoire de messages (*mailbox - fichier texte*)
  - /usr/spool/mail/jean, /var/mail/jean ...
- protocoles de transport
  - tcp/ip, X25, ...

## Rôle de Sendmail





## Les RFC

- Des RFC (Request For Comments) définissent :
  - L'envoi,
  - La réception
  - La structure des adresses
  - Le format des lettres
- Lectures recommandées
  - RFC 822 Format des messages
  - RFC 821 Protocole SMTP
  - RFC 974 Courrier et DNS
  - RFC 1035 DNS
  - RFC 1123 prérequis pour les sites Internet

..... mais lectures difficiles !



## *Format des messages : RFC 822*

- Structure d'un message
  - En-tête
  - Ligne blanche
  - Corps du message
    - suite de lignes terminées par CR/LF



## *Format des messages : RFC 822*

- Format des lignes d'en-tête
  - FROM: expéditeur
  - TO: destinaire(s)
  - CC: copie à
  - BCC: copie aveugle
  - REPLY-TO: adresse de réponse
  - ERROR-TO: adresse en cas d'erreurs
  - DATE: date expédition
  - RECEIVED: informations de transferts
  - MESSAGE-ID: identificateur unique de msg
  - SUBJECT: sujet



## *Format des messages : RFC 822*

- Adresse électronique : identifie de manière unique chaque boîte aux lettres.
  - `Personne@Machine.Domaines`      `jean@smtphost.dim.jussieu.fr`
  - `Personne@Domaine`                  `Gautier@urec.cnrs.fr`
  - extension RFC 1123 , routage ( % ) : `jean%jussieu.fr@uvsq.fr`
  - les adresses littérales sont déconseillées : `jean@134.157.4.21`
  
  - Pas de différence entre minuscules et majuscules pour la partie distante mais en théorie importante pour la partie locale ( de nombreux systèmes ne la font pas )
  - Attention aux caractères autorisées ( limitation par RFC du DNS)
- Postmaster
  - Utilisateur qui reçoit tous les messages en erreur.
  - Boîte aux lettres obligatoire pour Postmaster.



## *Autre types d'adresses*

- Adresse **UUCP**    `site1!site2!...!siteN!user`
  - `user@site.uucp` --> nécessite une passerelle uucp/Internet
- Adresses **Bitnet**    `user@site` ou `site` est un nom de machine
  - `user@site.bitnet`
- Adresse **decnet** `user::machine`
- Adresse **X400** `<C=fr/A=adm/P=section/N=user>`



## *Simple Mail Transfer Protocol : RFC821*

- Permet d'échanger du courrier électronique (E-Mail),
- C'est la messagerie de l'Internet
- Version sendmail 8.9.1a (**Mars 1999**) pour les systèmes Unix.
- Transfert direct
  - Entre l'ordinateur émetteur et un ordinateur destinataire:
    - **ce dernier peut-être celui où le destinataire du message a sa boîte aux lettres, mais ce n'est pas une obligation.**
  - Utilisation des enregistrements MX du DNS pour définir l'ordinateur destinataire
    - **MX : Mail Exchanger; DNS : Domain Name Server**
  - La remise du message dans la boîte aux lettres du destinataire du message est fonction de la politique messagerie du site.



## *Structure d'un courrier standard*

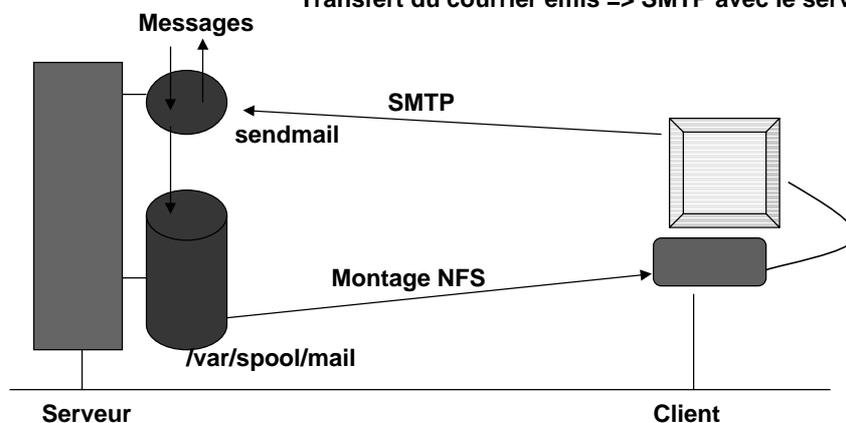
- Limitations (minimum admissible) :
  - Tout est sous forme de lignes ASCII sur 7 bits
  - 2 parties
    - l'entête (**définit les services attendus**)
    - corps (**le texte de la lettre**) terminé par une ligne avec "." comme **premier et unique caractère**
  - nom utilisateur < 64 caractères
  - nom de domaine < 64 caractères
  - nombre de destinataires < 100
  - ligne à ligne avec CR/LF
    - ligne < 1000 caractères.
  - Les besoins d'extensions et l'arrivée de mime ont donné lieu à  
=> Extended SMTP (RFC 1425)
  - L'envoi d'un binaire n'est pas possible sans structuration préalable.

## Envoi-Réception entre deux ordinateurs

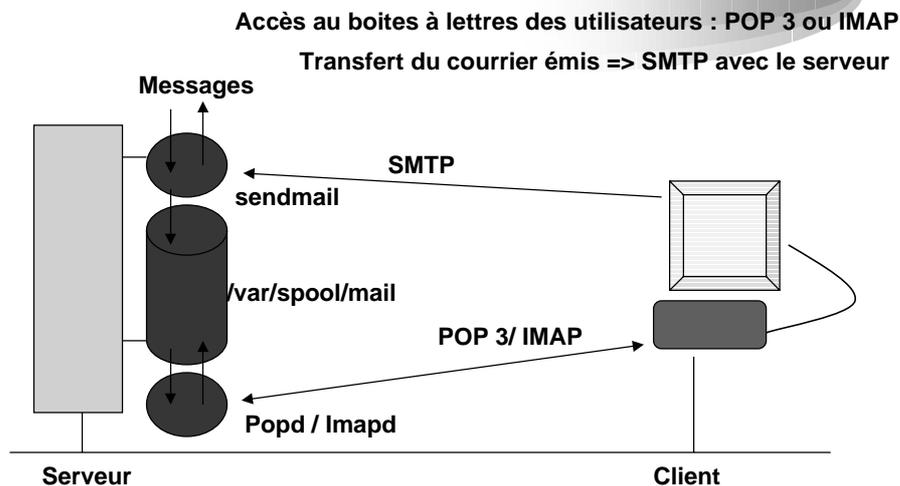
- Définit par le protocole SMTP
- Mode client - serveur
  - Programme sendmail, fichier de configuration: sendmail.cf
  - client : commande de base Mail sous Unix /bin/mail
  - serveur : démon Sendmail
- Utilise TCP, le serveur est en attente sur le port 25
  - Pour test : telnet nom\_du\_serveur\_smtp 25
- Un seul exemplaire du message est envoyé à un site ayant plusieurs destinataires
- Le dialogue est en ASCII
- Il n'y a pas de vérification sur l'origine du message

## Le modèle Client/Serveur avec stations de travail

Accès au boîtes a lettres des utilisateurs : Montage NFS  
 Transfert du courrier émis => SMTP avec le serveur



## Le modèle Client/Serveur avec Micros



## POP et IMAP

- Des protocoles permettant à des interfaces utilisateur (Eudora, Outlook, ... ) de relever le courrier sur un serveur
- **POP**: Post Office Protocol
  - protocole simple
  - transfère les nouveaux messages de la boîte au lettre, sur le serveur, vers la machine cliente (Mac ou PC)
- **IMAP**: Interactive Mail Access Protocol
  - évolution de POP, permet de gérer les dossiers sur le serveur
  - les courriers restent sur le serveur, sont triés et rapatriés à la demande de l'utilisateur, puis remis en place à la fin de la session



## Session SMTP (1)

- Depuis l'ordinateur calypso.urec.fr
- Commande : Mail -v gross@imag.fr
  - Les réponses de la machine distante sont précédées par un nombre
  - Les commandes envoyées par la machine calypso sont précédées de >>> (ce n'est pas dans le protocole)
- calypso appelle en IP la machine imag
  - ce n'est pas encore le dialogue
  - gross@imag.fr... Connecting to imag.imag.fr. (smtp)...
- le daemon sendmail de imag répond
  - 220-imag.imag.fr Sendmail 8.6.11/8.6.9 ready at Wed, 7 Feb 1996 10:59:34 +0100



## Session SMTP (2)

- les stations entament le dialogue
  - 220 ESMTP spoken here
  - >>> EHLO calypso.urec.fr
  - 250-imag.imag.fr Hello calypso.urec.fr [134.157.4.15], pleased to meet you
  - 250-EXPN
  - 250-SIZE
  - 250 HELP
- calypso donne le nom de l'expéditeur
  - >>> MAIL From:<Jean-Paul.Gautier@urec.fr> SIZE=200
  - 250 <Jean-Paul.Gautier@urec.fr>... Sender ok
- calypso donne le nom du destinataire
  - >>> RCPT To:<gross@imag.fr>
  - 250 <gross@imag.fr>... Recipient ok

## Session SMTP (3)

- *calypso* indique qu'il va transférer le message
  - >>> DATA

354 Enter mail, end with "." on a line by itself

*la lettre avec son entête et son corps est envoyée ici*
- *imag* indique qu'il l'a bien remis à *gross*
  - 250 KAA18774 Message accepted for delivery  
gross@imag.fr... Sent (KAA18774 Message accepted for delivery)
- *calypso* désire terminer la transaction (QUIT)
  - Closing connection to imag.imag.fr
  - >>> QUIT
  - 221 imag.imag.fr closing connection

## Lettre reçue (1)

- **Les "Received" indiquent le chemin suivi, dans l'ordre inverse.**

*Ils sont ajoutés par les machines ( relais SMTP ) à travers lesquelles le message a transité*

*Ils indiquent: l'origine, la destination, date et parfois l '@ de l'enveloppe de destination.*

*Ils permettent de retrouver l'origine du message.*

*Utile pour le suivie et la correction an cas de problèmes.*

*Evite les bouclages de messages ( arrêt entre 17 et 25 champs received ).*



## *Lettre reçue (2)*

- Received: from nez-perce.inria.fr by calypso.urec.fr (8.6.10/urec-1.0) with ESMTP; Mon, 5 Feb 1996 10:51:13 +0100
- Received: from cf01 (cledf.edf.fr [192.54.193.133]) by nez-perce.inria.fr (8.7.1/8.7.1) with ESMTP id KAA11083 for <art-hdt@inria.fr>; Mon, 5 Feb 1996 10:51:12 +0100 (MET)
- Received: from cli53nb.der.edf.fr (clucomx.der.edf.fr [130.98.2.21]) by cf01 (8.6.12/8.6.12) with SMTP id KAA02888 for <art-hdt@inria.fr>; Mon, 5 Feb 1996 10:51:01 +0100
- Relayed; 05 Feb 96 10:43:27+0100
- X400-received: by /PRMD=EDFDER/ADMD=ATLAS/C=fr/;
- Relayed; 05 Feb 96 10:43:27+0100



## *Lettre reçue (3)*

- *Destinataire*
  - To: gross@imag.fr
- *Copie conforme*
  - Cc: tuy@urec.fr
- *Sujet du courrier*
  - Subject: GERET
- *Date d'envoi (départ - attention à la synchronisation des horloges)*
  - Date: Fri, 10 Jan 92 15:39:16 +0000
- *Origine*
  - From: jpg@urec.fr
- *les champs X- sont libles et non normalisés*
- *Ligne blanche de fin d'en-tête et de début de corps,*
  - Je vous avais demande en octobre dernier de faire partie du groupe GERET et a être informe du planning de ses réunions..

## Courrier et DNS (1)

- "Resource record" du DNS de type MX

urec.fr.	IN	MX	100	dione.urec.fr.
			200	pamir.inria.fr.

- Il est utile d'avoir plusieurs MX ( le poids le plus faible est préféré )

- Tolérance aux pannes et répartition de charges:

urec.fr	IN	MX	10	smtp.urec.fr
smtp.urec.fr.	IN	A		134.157.4.24
				134.157.4.32

## Courrier et DNS (2)

- Traitement

- s'il n'y a pas de règles spécifiques de routage (local, mailertable), utilisation des MX
- s'il n'y a pas de MX, alors envoyer le courrier directement à la machine en utilisant l'adresse IP. Si il n'y a pas d'adresse IP, rejet du message.
- Si on apparaît dans la liste des MX, retirer les MX de poids égaux et supérieurs (moins prioritaires).
- essayer les MX par ordre de priorité décroissante. En cas d'échecs, on conserve le message et on réessaye à intervalle régulier.
- Toujours mettre un champ MX même sur le serveur ( dione.urec.fr IN MX 0 dione.urec.fr )
- Si on est le MX préféré et qu'il n'y a pas de règles locales, le message est rejeté.
- Attention, certains routeurs modifient ce traitement...

## *Evolutions*

- *sendmail* est le routeur de courrier depuis 1982
- 1982 : RFC 822
  - Format des Messages de l'Arpanet
- 1993 : MIME - RFC 1521
  - Multipurpose Internet Mail Extensions
- 1994 : SMTP service Extensions - RFC 1652, 1869
  - Comment transférer des caractères 8 bits

## *MIME : les buts*

- Standardiser les méthodes de transfert de données 8 bits,
- Structurer le corps du message en contenus (body-parts),
- Standardiser les différents contenus possibles.

*Un en-tête est rajouté à ceux définis dans le RFC 822*

- Mime-version:1.0



## *MIME : types d'encodage*

- Texte 7 bits, US-ASCII
- Quoted-Printable
  - Caractère non US-ASCII remplacé par une séquence =XY
    - XY est le code hexadécimal du caractère.
  - Il faut préciser l'alphabet utilisé.
  - Essentiellement utilisé pour le texte.
- Base 64
  - Texte, image, son
  - 24 bits découpés en 4 caractères US-ASCII
    - jeu de caractères = alphabet de 6 bits.
- 8Bits
  - les lignes sont composées de caractères 8 bits
  - Il faut préciser l'alphabet : iso-latin1
- Binary



## *MIME : structure du message*

- Des en-têtes supplémentaires décrivent le contenu du message
- Les contenus sont standardisés en 7 types.
  - protocole ouvert à des extensions.
- Chaque type de contenu est qualifié par un sous-type

## *MIME : structure du message*

- **Multipart**
  - Multipart/mixed
    - **plusieurs parties avec affichage en parallèle.**
  - Multipart/parallel
    - **d'autre(s) message(s) inclus dans le message**
  - Multipart/digest
    - **partie du message affichée suivant l'environnement du correspondant.**
  - Multipart/alternative
- **Message**
  - Message/rfc822
  - Message/partial
    - **découpage d'un long message**
  - Message/external-body
    - **accès à une référence, à un fichier : FTP, TFTP, Local File, mail Server.**

## *MIME : formats de données*

- Text
  - Text/plain : charset=iso-8859-1
  - Text/richtext
- Image
  - Image/gif
  - Image/jpeg
- Audio
  - Audio/basic
- Video
  - Video/mpeg
- Application
  - Application/octet-stream : exemple word
  - Application/postscript

## *MIME : exemple de message*

```
- Mime-Version:1.0
- From:
- TO:
- Subject:
- Content-Type: multipart/mixed;
  boundary=unique-boundary-1
.
--unique-boundary-1

- Content-type: text/plain; charset=US-ASCII

  blabla

--unique-boundary-1
```

## *MIME : exemple de message*

```
- Content-Type: multipart/parallel;
  boundary=unique-boundary-2
.
.
--unique-boundary-2
- Content-type: audio/basic
  Content-Transfer-Encoding: base64

  fichier audio

--unique-boundary-2
- Content-type: image/gif
  Content-Transfer-Encoding: base64

  fichier image
```

## *MIME : une bonne UA*

- Reconnaître et afficher du texte US-ASCII,
- Reconnaître les autres jeux de caractères,
- permet de sauvegarder les contenus non reconnus dans un fichier pour traitement ultérieur,
- Reconnaître et afficher les contenus de type Message/RFC822
- Reconnaître le type Multipart/mixed
- Reconnaître le type Multipart/alternative
- Traiter les Multipart non reconnus comme Multipart/mixed
- Décoder les contenus de Application/\* si l'encodage quoted-printable ou base64 est utilisé, puis offrir de sauver le résultat dans un fichier.

## *MIME : quelques UA*

- Domaine public
  - elm, mutt, pine, mh
  - xmh : ne pas utiliser
  - mixmh : xmh + caractères accentués
  - exmh : difficile à installer (mh, metamail, glimpse), assez lent.
  - meuf
  - ml
- Payant
  - Z-mail : origine Nec, installation facile, configuration aisée (PC, Mac, station)
  - Eudora : versions 1.5 et plus; serveur popd de qualcomm sur une station.
  - Netscape Mail, IE Mail Microsoft, Outlook Microsoft

## *ESMTP(1)*

- Ajoute des fonctionnalités nouvelles
  - Transport de messages 8 bit MIME
  - Taille maximale de message
  - Fonctions autorisées (EXPN, VRFY, ...)
  - Autres extensions (Pipelining, extensions privées)
- Le message de bienvenue ESMTP est EHLO (au-lieu de HELO), en cas de réponse négative le client doit basculer vers l'ancien protocole

## *ESMTP(2)*

- A la connexion, le serveur indique les extensions qu'il supporte
  - Exemple
    - EHLO calyspo.urec.cnrs.fr
    - 250-dione.urec.fr Hello calyspo.urec.cnrs.fr?
    - 250-8BITMIME
    - 250-SIZE 2048000
    - 250-EXPN
    - 250-DSN
    - 250-VERB
    - 250-HELP



## *ESMTP(3)*

- **SIZE** : indique, avant l'envoi, la taille maximale de message (au client de chercher un autre chemin si son message est plus large)
- **8BITMIME** : le serveur accepte les messages 8 bit au format MIME (donc avec des jeux de caractères autres que ASCII) et fera les transformations nécessaires s'il doit renvoyer à un autre serveur non 8BITMIME
- **EXPN** : la fonction EXPN (expansion d'aliases) est autorisée
- **DSN** : Delivery Status Notification (accusé de délivrance, ...)



## *Problèmes de sécurité(1)*

- Divers problèmes:
  - pertes de messages
    - par l'agent de transport ( le MTA )
    - par l'utilisateur
    - par un incident matériel
  - Ecoute des messages
  - Falsification des messages

## *Sécurité (2)*

- Réponses
  - extension DSN informe l'expéditeur sur ce qui est arrivé au message.
  - Chaque relais sait indiquer si le message est envoyé correctement ou non au relais suivant.
  - Limitation possible du nombre de relais par SMTP
  
  - Utilisation du Chiffrement entre les serveurs (PGP)

## *SPAM (1) : Le problème*

- Courrier non sollicité envoyé à plusieurs personnes ( idem au prospectus des boites aux lettres )
- Les adresses sont récupérées via les News, les listes de diffusion, les pages Web ( analyse des champs mailto ).
- Apparue avec l'explosion du nombre d'utilisateur de l'Internet ( solutions apparues en 1997 ).
- Un commerce florissant ( vente de fichiers d'utilisateurs... )

## SPAM (2) - Solutions

- Reconnaître l'auteur d'un SPAM
- Adresse « abuse »
- Filtrage au niveau personnel
- Filtrage au niveau d'un site
  - liste noir des « spammeurs » connus
  - interdire le relaying
  - refuser les adresses invalides
  - refuser les adresses IP d'expéditeurs non valides
- L'avenir : règles anti-spam de sendmail, initiative MAPS RBL ( Mail Abuse Protection System - realtime Blackhole List ) de P.Vixie ( constitution d'une arborescence DNS composée des @IP de domaines spammeurs - d.c.b.a.maps.vix.com )

## Appel de sendmail

- par un agent d'utilisateur
  - mail jean@jussieu.fr
- directement par l'utilisateur
  - /usr/lib/sendmail jean@jussieu.fr
  - => il faut alors renseigner tous les champs manuellement
  - From :
  - To :
  - Subject :
  
  - message
  - .
- par un agent de transport (ex : uucp)
- par un autre *sendmail* : connexion SMTP

## Les fichiers

– sendmail	le programme
– sendmail.cf	fichier de configuration
– sendmail.fc	configuration compilée
– sendmail.hf	aide lors du dialogue SMTP
– sendmail.st	informations comptables
– aliases	les aliases en clair
ancienne version de sendmail (< 8.5)	
• <b>aliases.pag</b>	<b>version "compilée" des aliases</b>
• <b>aliases.dir</b>	"" ""
nouvelles version de sendmail	
• <b>aliases.db</b>	
realiases	les reverse aliases en clair
<b>realiases.db</b>	<b>version "compilée" des reverse-aliases</b>

**Remarque : suivant les "Unix", les répertoires peuvent être différents**

## Aliases, Reverse aliases.

- Les aliases permettent de redéfinir le routage du courrier local
  - nom complet d'utilisateur Paul.Durand: paul
  - utilisateur virtuel paul: paul@jussieu.fr
  - programme
  - liste de diffusion liste: pierre@uvsq.fr, jean@urec.fr  
owner-liste: pierre@uvsq.fr  
liste-request: pierre@uvsq.fr
  - liste de diffusion gérée par un utilisateur  
liste : :include:/home/paul/liste
  - *Pas de nom non local*
- Les reverses aliases permettent de modifier le champ *From* à l'envoi d'un message
  - nom d'utilisateur paul: Paul.Durand
- Ne pas oublier d'utiliser *newrealiases*

## *fichier .forward*

- fichier géré par l'utilisateur pour effectuer le re-routage des messages qui lui sont adressés
  - droits d'accès au fichier : 600
- même syntaxe que les aliases
- consulté après les aliases

## *Configuration : sendmail.cf (1)*

- règles de réécriture
  - réécriture des adresses
    - **dupont@voisin.uucp**                    =>    **voisin!toto**
    - adresse canonique (format unique) :**
    - **partie-locale<@machine-cible>routage**
  - sélection du routage du courrier
  - elles sont regroupées en ensembles:
    - **repérés par un numéro, certaines ont des rôles prédéfinis;**
      - 0 : sélection du mailer
      - 3 : canonisation des adresses
    - **accomplissant une transformation sur une adresse**

## Configuration de sendmail (2)

- définition de *macros*
  - souplesse de configuration
  - accès aux spécificités du système
  - exemples :
    - **DDurec.cnrs.fr**
    - **DMSmtphost.\$D**
    - **De\$j Sendmail \$v ready at \$b**

## Configuration : sendmail.cf (2)

- définition de *mailers*
  - facilité d'addition de "réseaux" de messagerie
  - c'est un nom, un programme et ses paramètres, des caractères de fin de ligne, des flags, une taille maximum de messages, des règles de réécriture pour les adresses
  - *exemple* : **Mlocal, Path=/bin/mail, Flags=DFMPImS, Sender=10, Recipient=20, Argv=mail -d \$u**
  - mailers obligatoires
    - local : remise des messages.
    - prog : utilisé pour les alias sous forme de programme
    - error : implicitement défini
  - SMTP , cas particulier
    - pas de programme externe
    - inclus dans sendmail



## *Tests*

- Pouvez-vous envoyer un courrier local ?
  - /usr/lib/sendmail -v paul
  - -v : mode verbose, très intéressant lors de la mise en oeuvre
  - si OK alors beaucoup de règles sont correctes !
- telnet localhost smtp
  - si la réponse est du type  
220 shiva.jussieu.fr Sendmail ready at .....
  - si vous quittez la session correctement
  - alors sendmail est à l'écoute des connexions SMTP.
- vérifier qu'un site distant peut vous atteindre :
  - mail moi%mon-site@site-distant
- vérification des règles de réécriture
  - /usr/lib/sendmail -bt
  - en cas de problème majeur : /usr/lib/sendmail -bt -d21.99