Deux applications réseaux

SMTP

Protocole HTTP

Service de messagerie

- Autre nom : courrier électronique, e-mail, couriel

  ➡ permet d'échanger des messages et des fichiers

- Il nécessite un serveur de messagerie accessible à partir d'internet. Le serveur dispose d'une boîte à lettre (BAL) pour chaque client géré par la messagerie.

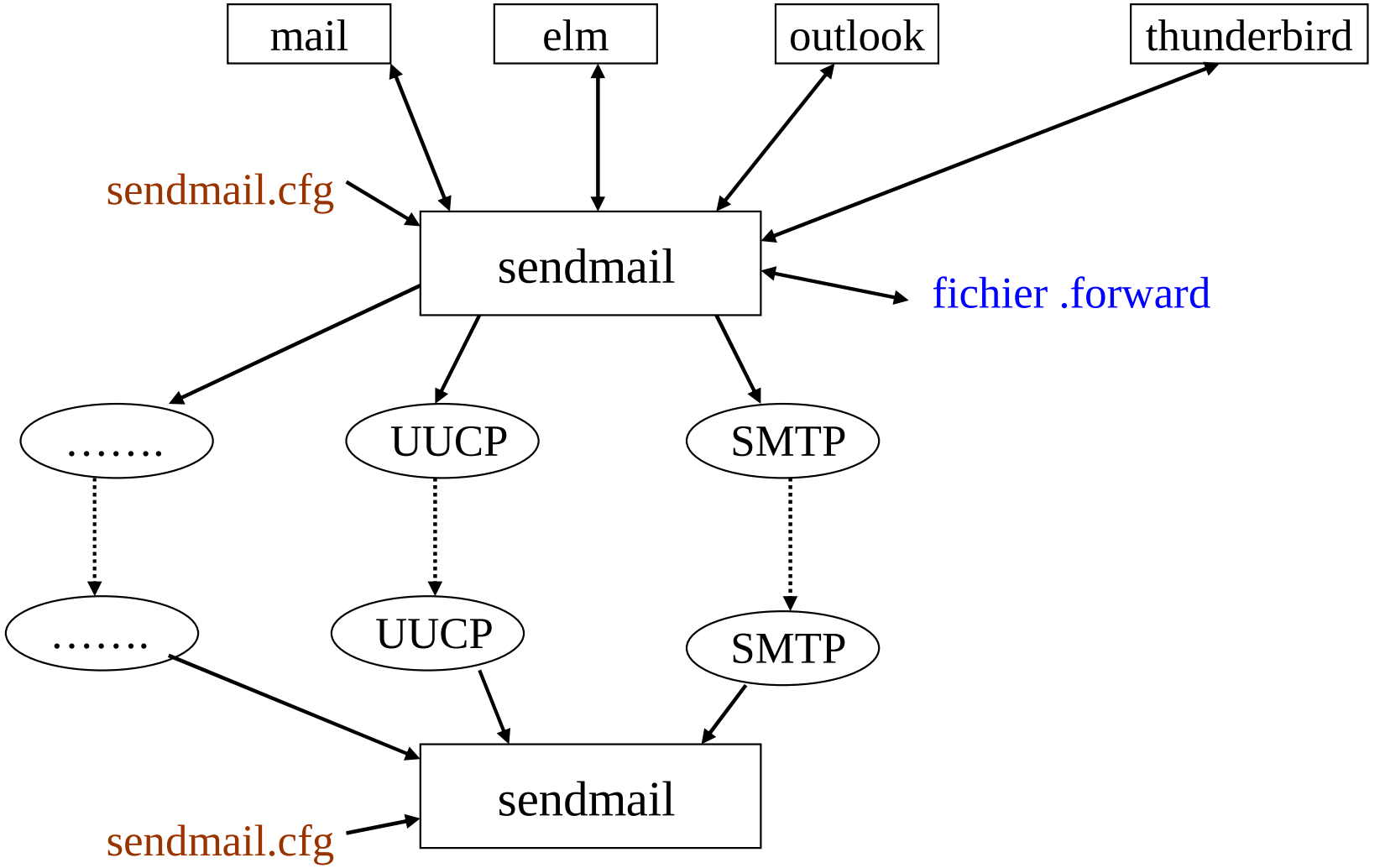
- Les messages sont stockés par le serveur de messagerie, en attendant que le client vienne consulter sa boîte aux lettres.

  - le message peut être lu :
    - en *mode online* - message stocké sur le serveur et lu à distance
    - en *mode offline* – message déplacé sur la station client et effacé du serveur

Terminologie

- Logiciels de messagerie (user agent)
  - Thunderbird, Eudora, Lotus Notes, Outlook, Mail, elm, xmh, …
- Logiciels de gestion de messagerie (message transfert agent)
  - Exchange server, Domino mail server, eudora worldmail server, sendmail,…
- Protocole de transport des messages (couche applicative)
  - SMTP, UUCP, X400, X500,…
- Stockage des messages
  - /usr/spool/mail/…, /var/mail/…

Rôle des entités



SMTP

SMTP: RFC 821

Simple Mail Transfert Protocol

- Permet d'échanger du courrier électronique

- le format des adresses des utilisateurs fait figurer le nom de l'utilisateur suivi du nom de domaine : laurenco@isima.fr

  identifie de manière unique chaque boîte aux lettres (personne@machine.domaine, personne@domaine)

- le codage utilisé pour le message et les fichiers attachés :
  - texte pur codé en ASCII 7 ou 8 bits (RFC 822) pour une prise en compte des caractères accentués
  - standard MIME (Multipurpose Internet Mail Extension) pour du texte formaté, des images ou du son

RFC 822

Structure d'un message

- En tête
- Ligne blanche
- Corps du message (suite de lignes terminés par CR/LF)

En tête

- From : expéditeur
- To : destinataire(s)
- CC : copies à
- Bcc : copie aveugle (destinataire caché)
- Reply-to : adresse de réponse
- Error-to : adresse en cas d'erreur
- Date : date et heure de l'envoi
- Message-id : numéro unique permettant de référencer le message
- Received : informations de transfert
- Subject : sujet

Structure d'un courrier

- Limitations
  - Tout est sous forme de lignes ASCII
  - 2 parties
    - l'entête **(définit les services attendus)**
    - corps **(le texte de la lettre est terminé par une ligne avec "." comme premier et unique caractère)**
  - nom d'utilisateur < 64 caractères
  - nom de domaine < 64 caractères
  - nombre de destinataires < 100
  - une ligne < 1000 catactères
  - utilisation du format MIME (RFC 1521) pour envoyer des fichiers non ASCII

Session SMTP (1)

- 3 phases
  - Etablissement de la connexion au niveau smtp et identification de la source et la destination
  - Envoi du message avec les différents entête
  - Libération de la connexion

- Session
  - Connexion tcp sur le port 25
  - Le serveur renvoie le code 220 (service disponible) suivi de son nom

    ex : 220 sp.isima.fr SMTP sendmail….
  - Envoi d'une requête de connexion par la commande : *HELO*
  - Réponse par le code 250 (OK), et nom, please to meet you

    la connexion au niveau SMTP est réalisé

Session SMTP (2)

- Session (suite)
  - Envoi du nom de l'expéditeur, réponse 250 (OK) *(MAIL FROM)*
  - Envoi du nom du destinataire, réponse 250 (OK) *(RCPT TO)*
  - Début du transfert du message *(commande DATA)*
    réponse : *354 Enter mail, end with "." on a line by itself\r\n*
  - Envoi du message : Message-id, From, To, Date, Message, …
  - Pour terminer : QUIT
    réponse : 221 sp.isima.fr closing connection

- 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 permettent de retrouver l'origine du message
  - cela évite le bouclage de messages (max 25 champs received)

ESMTP

- Plus récent que SMTP
- Apporte des fonctionnalités supplémentaires
  - taille des messages
  - transport des messages en 8 bits
  - plus de fonctions disponibles

  - Le message de bienvenue est *EHLO.* En cas de réponse négative, le client doit basculer vers l'ancien protocole.

Format MIME (1)

- Types d'encodage
  - Texte 7 bits, ASCII
  - Texte 8 bits
    - les textes sont composés de caractères 8 bits
    - il faut préciser l'alphabet : ex : iso-latin1
  - Base 64
    - pour les messages binaires
    - groupe de 24 bits, segmentés en 6 bits
      (3 octets, 4*6 bits 0 – 25 → A – Z , 26 – 51 → a –z, 52 – 61 → 0 – 9, 62 → +, 63 → / (Man = TWFu))
  - Quoted-Printable
    - codage ASCII normal
    - pour ce qui n'est pas ASCII, valeur = hexa.

Format MIME (2)

- Contenu est divisé en 7 types différents, eux mêmes re-divisés en sous-types.
  - Text
    - plain, richtext
  - Image
    - Gif, jpeg
  - Audio
    - basic
  - Video
    - mpeg
  - Application
    - octet-stream (WORD), postscript
  - Message
    - rfc822, partial, external-body (référence à un fichier dans internet)
  - Multipart
    - mixed, alternative (plusieurs formats) , parallel, digest (plusieurs messages)

Format MIME (3)

- Exemple
  - Mime-Version : 1.0
  - From :
  - To :
  - Subject

  - Content-type : multipart/mixed; boundary="coucou"
    --coucou
    content-type : text/plain; charset=iso-8859-1

    bla-bla

    --coucou
    content-type : audio/basic
    content-transfert-Encoding: base64

    fichier audio

    --coucou

POP

- Protocole permettant de relever le courrier sur un serveur
  - POP 3 : Post Office Protocol (version 3) port 110
    - permet l'authentification (login, passwd en clair)
    - réception seulement des courriers sur un serveur (envoi par smtp)
    - réception des messages d'erreur ou d'acquittement

    - Il est nécessaire de télécharger l'intégralité du courrier sur la station avant la lecture, sans possibilité de manipuler directement les messages sur le serveur

    - POP utilise une syntaxe en 4 caractères :
      - STAT : récupère le nombre et la taille des messages en attente
      - RETR msg : permet de récupérer un msg
      - DELE msg : suppression
      - USER, PASS : login, passwd
      - QUIT : fin
        (ex : DELE 10, rep = OK, RETR 11, rep=OK+ msg, DELE 11,….)

IMAP

- Protocole permettant de relever le courrier sur un serveur
  - IMAP : Internet Message Access Protocol port 143
    - permet l'authentification si nécessaire de façon cryptée
    - gère les mails sur le serveur, donc pas besoin de télécharger
    - permet de trier les mails, faire des répertoires, etc…
    - utilise des drapeaux pour la gestion des mails

    - IMAP est très utilisé pour les serveurs webmail.

Quelques fichiers

- Alias
  - permet de créer une liste de personne
    → liste de diffusion
  - permet de redéfinir le routage du courrier local
    → un utilisateur aura plusieurs noms possibles

    ex : laurenco@isima.fr, patrice.laurencot@isima.fr

- Le fichier .forward
  - fichier géré par l'utilisateur pour effectuer le re-routage des messages qui lui sont adressés
  - fichier consulté après les aliases

SPAM

- Courrier non sollicité envoyé à plusieurs personnes
- Actuellement , 95 % mail → SPAM, futur 98%
- Les adresses sont récupérées via les listes de diffusion, les pages Web, ou même achetés…

- **Solutions :**
  - Reconnaître l'auteur d'un SPAM
  - Filtrage au niveau personnel
  - Filtrage au niveau d'un site
    - liste noire des "spammeurs" connus
    - refuser les adresse invalides
    - interdire le relayage
    - refuser le mail pour qu'ils soit renvoyé plus tard (efficace, mais cher en BP)

Protocole HTTP

- HTTP : Hypertex Transfert Prottocol (RFC 1945)
  - protocole de rapatriement des documents
  - protocole de soumission de formulaire
  - Fonctionnement simplifié
    - Connexion en TCP
    - Demande (GET) d'un document → envoi d'une requête
    - Renvoi du document ou d'une erreur →envoi d'une réponse
    - Déconnexion
  - Mais possibilités d'optimiser les connexions, d'avoir des dialogues plus complexes

Requête HTTP

- Une requête http comprend:
  - une ligne de requête contenant
    - la méthode ou commande
    - l'URL
    - la version du protocole utilisé (http/1.1)
  - les champs d'en-tête de la requête
    - des informations supplémentaires
      info : valeur
  - le corps de la requête

  - Exemple :
    - GET http://www.isima.fr HTTP/1.1
      Accept : text/html, application/xml
      Accept-Encoding : gzip, deflate
      Connection : keep-alive

Commande HTTP

- Quelques commandes
  - méthode GET
    - récupération d'un document
  - méthode HEAD
    - récupération des informations sur le document
  - méthode POST
    - envoi de données au programme situé à l'URL spécifié
  - méthode PUT, DELETE, LINK, UNLINK
    - envoi de documents et gestion du site

Réponse HTTP

- Une réponse http comprend:
  - une ligne de statut contenant
    - la version du protocole utilisé
    - le code de statut
    - la signification du code
  - les champs d'en-tête de la réponse
    - des informations supplémentaires
      info : valeur
  - le corps de la réponse

  - Exemple :
    - HTTP/1.1 301 Moved Permanently
      Date : Mon, 05 feb
      Transfer-Encoding : chunked
      Connection : close

Codes de réponse

- 100 à 199
  - informationnel
- 200 à 299
  - Succès de la requête
    - 200 : OK
    - 201 : Created
    - 202 : Accepted …
- 300 à 399
  - Redirection
    - 301 : moved
    - 302 : found …
- 400 à 499
  - Erreur due au client
    - 400 : bad request
    - 401 : unauthorized
    - 402 :payment required
    - 403 : forbidden
    - 404 : not found
- 500 à 599
  - Erreur due au serveur
    - 500 : internal error
    - 501 not implemented
    - 502 : bad gateway …

Commandes HTTP

- Pour récupèrer un document
  - commande GET : les paramètres sont passés dans la ligne de l'URL

    (le " " est remplacé par +, caractères spéciaux par %code ascii)
    - ex: GET /toto?name1=val1&name2=val2 HTTP/1.1
  - commande POST permet de mettre les arguments dans le corps de la requête (→ invisible)
    - ex : POST /toto HTTP/1.1

      accept : text/html ….

      * une ligne blanche*

      name1= val1 &

      name2 = val2

Suivi de session (1)

Motivations

- la notion de session est importante dans une application conversationnelle
  - commerce électronique
- Hors, HTTP est un protocole sans état

  → une connexion à chaque demande

  le serveur ne maintient pas d'informations liées aux requêtes d'un même client

- Comment implanter la notion de session sur plusieurs requêtes HTTP ?

Suivi de session (2)

- Le serveur génère un identificateur de session et associe un état à une session

  le client renvoie l'identificateur de session à chaque requête

- Plusieurs solutions possibles:
  - Input HIDDEN dans les formulaires
    - on renvoie à chaque fois un identifiant
  - la Ré-écriture dans chaque URL
    - on remet l'identifiant dans les réponses
  - Les cookies

VLAN

VLAN (1)

Utilisation du LAN :

- **Les réseaux sont liés aux concentrateurs ou aux commutateurs (HUB ou Switch)**
- **Les utilisateurs sont regroupés géographiquement**
- **Pas de sécurité sur un segment**
  (améliorations avec les commutateurs)
- **La mobilité entraîne souvent un changement d'adresse**
- **Plan d'adressage difficile**

→ Domaine de collision réduit avec les commutateurs

VLAN (2)

Pourquoi un VLAN (Virtual LAN):

- **Augmenter la sécurité entre les utilisateurs**
- **Limiter les domaines de broadcast entre utilisateurs**
- **Permettre une certaine mobilité aux utilisateurs**
- **Permettre à des utilisateurs dispersés géographiquement de partager des données**
- **But : règle 80/20**

Mise en place d'un réseau « logique »

➡ Pb : nécessite une couche 3 pour la communication entre VLANs

VLAN (3)

Un réseau VLAN de bout en bout a les caractéristiques suivantes:

- Les utilisateurs sont regroupés en VLAN qui dépendent de leur groupe de travail ou de leur fonction, mais pas de leur localisation physique.

- Tous les utilisateurs d'un VLAN doivent avoir les mêmes modèles de flux de trafic 80/20.

- Lorsqu'un utilisateur se déplace sur le campus, son appartenance à un VLAN ne doit pas changer.

- Chaque VLAN est caractérisé par un ensemble commun de besoins de sécurité pour tous les membres.

VLAN (4)

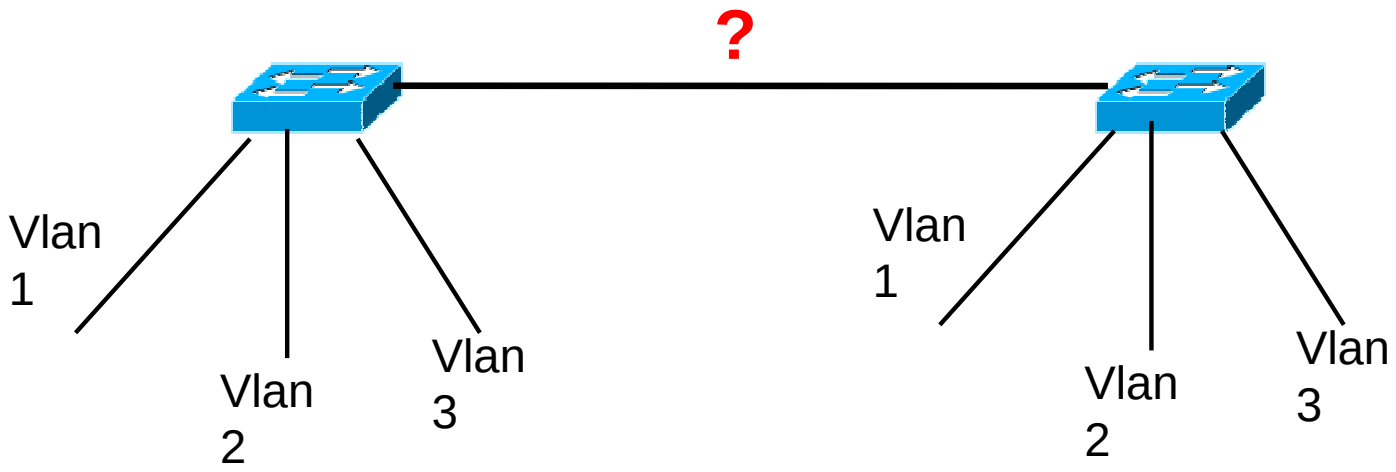
3 types de VLAN

- **VLAN axés sur le port** (Vlan statique)
  (simple à mettre en œuvre)
- **VLAN axés sur l'adresse MAC**
  (nécessite de la mémoire)
- **VLAN axés sur le protocole**
  (seulement pour protocole routable)

Vlan dynamique

Une adresse réseau de couche 3 unique doit être affectée à chaque VLAN. Cela permet aux routeurs de commuter les paquets entre les VLAN.

VLAN (5)



Combien de liaisons sont nécessaires entre éléments actif ? (Switch, routeur)

1 seule -> notion de trunking (lien agrégé)

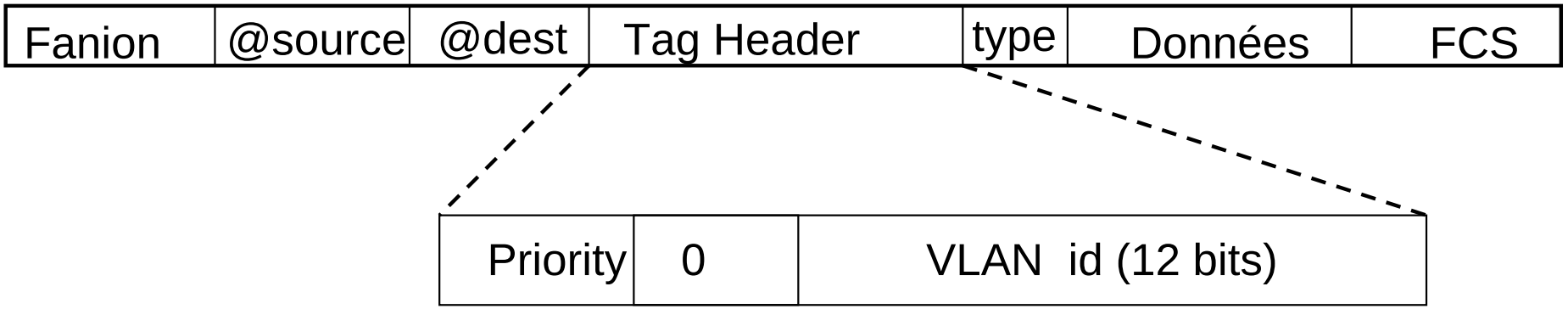
Plusieurs normes possibles : ISL (Inter-Switch Link)

→ Cisco (encapsulation)

IEEE 802.1 Q → normalisé (étiquetage)

VLAN (6)

Trame 802.1Q sur ethernet:



Autres protocoles : VTP → VLAN Trunking Protocol

- Facilite la gestion des VLANs sur les liens agrégés
- Trois états possibles pour un commutateur :
  - Serveur
  - Client
  - Transparent