A.U.T.E.L., une Approche Utilisateur de la Transmission En Ligne

Fabien Priotto
Centre Informatique de Gestion et Réseau
Faculté de médecine de Marseille, Université de la méditerranée
Fabien.Priotto@medecine.univ-mrs.fr

Résumé

Pour la transmission de fichiers de tout type et de grande taille tout en s'impliquant dans les habitudes de l'utilisateur sans contraintes, A.U.T.E.L. est une solution qui permet l'envoi de pseudo pièces jointes via messages Email en se basant sur les principes de base de l'Internet.

Mots clefs

EMAIL, GZIP, HTTPS, MIME, PHP, SMTP, SSL, ZLIB

1 Description du projet

Nommé A.U.T.E.L., l'idée de ce projet est née d'une observation quotidienne de l'utilisation de la bureautique et de l'Internet au sein des facultés de Médecine, Odontologie et Pharmacie de Marseille. Celles-ci hébergent une centaine de laboratoires de recherche, unités pédagogiques, et divers services administratifs.
Il s'agit d'une APPROCHE UTILISATEUR DE LA TRANSMISSION EN LIGNE des documents.

1.1 Un besoin

Pour les différents utilisateurs, scientifiques, administratifs, enseignants et étudiants, le besoin d'échanger des fichiers électroniques est fréquent et souvent impromptu. Par exemple, un chercheur veut transmettre un document de taille importante à l'issue d'une conversation téléphonique juste avant de prendre un avion. Peut importe le format (texte ASCII, texte encodé, binaire), le type de fichier (traitement de texte, image, fichier exécutable, document multimédia…) et sa taille, il souhaite le transmettre le plus simplement possible SANS SE SOUCIER DU CONTEXTE INFORMATIQUE DE SON CORRESPONDANT.

1.2 Un objectif

L'objectif du projet A.U.T.E.L. est de mettre en œuvre UNE SOLUTION POUR LA TRANSMISSION DE FICHIERS avec pour souci constant, la parfaite IMPLICATION DE SON USAGE DANS LES HABITUDES DE L'UTILISATEUR sans lui créer de nouvelles contraintes. Cela implique de ne pas avoir à impacter les postes utilisateurs. De plus, il faut garder à l'esprit que l'Internet constitue un réseau hétérogène sur lequel on ne peut préjuger de la qualité de la bande passante. On cherchera à MAITRISER LE TRAFFIC DES DONNEES et Un MECANISME DE COMPRESSION permettra de LIMITER LA TAILLE DU FLUX lors de la transmission des données en toute TRANSPARENCE pour les utilisateurs.

1.3 Une problématique complexe

S'il existe un grand nombre d'applications et protocoles (GroupWare, Peer2Peer, NFS Network File System, SMB Server Message Block, AppleShare, Netbios…) pour le partage en réseau local, la question d'un échange universel sur le "World Wide Web" demeure une problématique complexe et, tel que le souligne la conclusion dans l'article de Olivier Perret (GroupWare à l'Institut Pasteur, JRES 2001):
<< La possibilité de maquetter les différentes solutions intégrées avec des outils de base a permis de montrer les limites des outils intégrés…>>

1.4 Un constat

Dans la grande variété de l'usage de l'outil informatique (collaboration scientifique, rapports administratifs, usage personnel, multimédia…), le constat est frappant : L'échange de courrier électronique est devenu un réflexe pour la plupart des utilisateurs :
L'EMAIL S'IMPOSE SPONTANEMENT COMME LE MOYEN D'ACHEMINEMENT DES DOCUMENTS.
On observe cependant assez facilement le manque de sensibilisation général tant sur la constitution des messages (format du message, technique d'encodage, renseignement des champs d'en-tête …) que sur le mécanisme d'acheminement.

Aussi, l'échange de messages sophistiqués (Texte enrichi, pièces jointes, insertion d'objets) réserve bien souvent des surprises aux utilisateurs.

Il faut noter que la plupart des utilisateurs accordent une certaine confiance, parfois aveugle, au principe du courrier électronique. Cela est sans doute dû à une ILLUSOIRE SIMPLICITE ET CONFIDENTIALITE d'un échange de particulier à particulier.

Pourtant, le protocole SMTP, conçu initialement pour l'acheminement de messages simples définis par le RFC 822, met en jeu plusieurs intermédiaires, agent utilisateurs, agents de transfert de courrier et passerelles de messagerie. Chaque agent impactant le contenu du message, beaucoup d'implémentations du RFC 822 sur l'Internet supposent qu'aucune ligne dans un message n'ait une longueur supérieure à 1000 octets et qu'aucun message n'ait une taille supérieure à 64Ko. Nous verrons plus loin que l'extension MIME (Multipurpose Internet Mail Extensions) AMELIORE les possibilités des messages électroniques.

De plus, des mesures de plus en plus sécuritaires tendent à augmenter le filtrage des messages.
Dans ce contexte, de nombreux problèmes peuvent venir PERTURBER la transmission: La taille du message peut dépasser les quotas en vigueur, les mesures anti-Spam et anti-virus peuvent provoquer le rejet du message (Filtrage sur l'en-tête, sur le type de pièce jointe) et cela sur chacun des serveurs SMTP sollicités. De plus, la boite aux lettres du correspondant peut être saturée, certaines pièces jointes peuvent être refusées par le logiciel client de messagerie dont l'utilisateur maîtrise rarement le paramétrage complet …

Le « périple SMTP » d'un message Email pourrait être schématisé par la figure 1 :

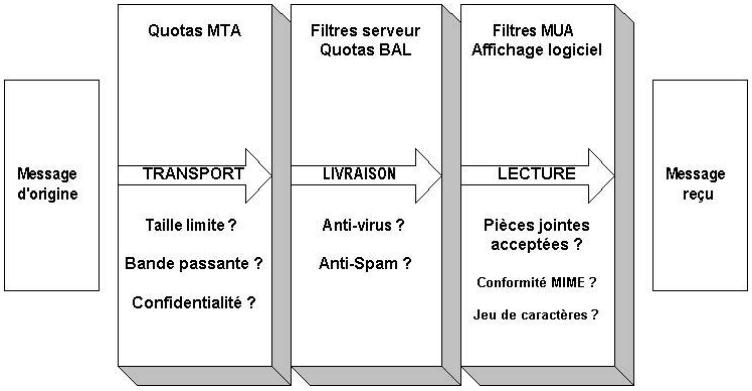

Figure 1 – *Le périple SMTP d'un message électronique*

On se réfèrera à la sous section 6.3 pour plus de détails sur la conformité MIME.

D'autre part, on remarque que la navigation Web est aujourd'hui le moyen d'accès aux données numériques le plus courant tout public confondu. Avec ou sans utilisation de Proxy, les protocoles HTTP (Port TCP 80) et HTTPS (Port TCP 443) constituent des portes de sorties incontournables pour tout réseau local ouvert sur l'extérieur digne de ce nom.

LE NAVIGATEUR WEB S'IMPOSE COMME L'INTERFACE D'ACCES.

1.5 Des détails

De cette observation, certains détails émergent notamment en ce qui concerne le comportement des utilisateurs et leur appréhension de l'outil informatique :
Les utilisateurs recourent très souvent à l'insertion de pièce jointe dans leurs messages pour acheminer les documents. L'insertion d'un ou plusieurs documents dans un message est désormais un geste familier, spontané voir systématique. A ce sujet, on peut déplorer une conséquence directe de cette attitude :
L'amélioration des bandes passantes des réseaux est suivie de près par l'augmentation de la taille des messages incluant des pièces jointes de plus en plus grosses. De plus, le nombre de destinataires multiplie la consommation de bande passante nécessaire à l'acheminement d'un même message.

L'AMELIORATION DE LA BANDE PASSANTE EST RATTRAPPEE PAR LA TAILLE DES FLUX si l'utilisation du réseau n'est pas canalisée.

On admet aujourd'hui que l'ouverture et la sauvegarde sur disque dur des pièces jointes sont des GESTES COURANTS ET FAMILIERS.

2 Le cahier des charges

Le cahier des charges a pour objectif de délimiter le périmètre fonctionnel du projet. Il décrit les principaux besoins et traitements à inclure. Il donne une vision d'ensemble, ce qui permet de cerner l'infrastructure technique adéquate.
L'architecture d'un système de transmission se voulant universel doit forcément s'appuyer sur LES PRINCIPES DE BASE du canal utilisé, en l'occurrence ceux de l'Internet :
- Le CONCEPT D'HYPERTEXTE s'emploie pour construire des documents qui référencent d'autres documents, permettant leur délocalisation et constituant ainsi les très efficaces liens hypertexte.
- L'URL (Uniform Ressource Locator) est l'adresse d'une ressource du réseau et inclue les arguments spécifiques à un protocole.
- HTTP (Hyper Text Transfer Protocol) est la méthode principale de transfert employée par les protocoles du Web pour déplacer les données d'un serveur vers un client. Protocole Client/Serveur, HTTP communique sous un modèle Requête/Réponse qui utilise MIME pour encapsuler les données. Caractéristique importante, le trafic de données entre un client et un serveur HTTP ressemble, sur le plan conceptuel, à un trafic Email. Il consiste en données (Le corps du message) et en méta données (les en-têtes de message).
- HTML (Hyper Text Markup Language) est un langage simple de marquage hypertexte utilisé pour le Web. HTML fait usage de marqueurs pour décrire et délimiter un document. Ces marqueurs permettent l'envoi d'en-têtes HTTP.
- MIME (Multipurpose Internet Mail Extensions) définit la structure des informations qui sont transférées entre le serveur HTTP et le navigateur. MIME permet la récursivité; en d'autres termes, un message MIME peut contenir un corps qui est lui-même un message MIME, et ainsi de suite. Transporté via SMTP, l'extension MIME permet d'échanger des messages imbriqués et enrichis. Précision importante, le type texte/Plain employé pour transporter le corps du message permet l'emploi de jeux de caractères US-ASCII (ASCII-7 bits) jusqu'au jeu ISO-8859-10.

2.1 Les besoins fonctionnels de la solution

- Permettre la mise à disposition immédiate du document :
Le document étant déjà enregistré sur le disque dur de l'expéditeur, celui doit pouvoir en une seule procédure mettre à disposition son document et en informer les destinataires.
- Limiter les contraintes :
Pour cela, le nom et mot de passe seront demandés à l'expéditeur seulement au premier accès à son espace. Cela impose au propriétaire de veiller au verrouillage de son espace lorsqu'il se déconnecte. Le destinataire aura accès à la lecture du document en un simple clic.
- Baser l'accès sur la Navigation Web HTTP 1.1:
Les grandes différences qui distinguent HTTP 1.1 de HTTP 1.0 sont, entre autres, l'amélioration des performances, la compression/décompression des fichiers ou encore les transferts de plages d'octets utiles dans le cas d'une reprise de connexion.
- Garantir la compatibilité avec un maximum de navigateurs
On se limitera à l'interprétation du langage HTML 3.2, version ayant introduit notamment l'utilisation des tables. Aucun interpréteur ni extension client ne sera nécessaire (Pas de JavaScript, Cascading Style Sheet, Sgml, Dhtml, ni Frames…)
- Baser la structure des messages (email mais aussi téléchargement des documents) sur MIME 1.0. MIME permet de spécifier le type du message (texte, image, son, vidéo ou plusieurs types à la fois), et le codage du message (7 bits, 8 bits, quoted-printable ou base64). MIME est décrit par les RFC 1341 à 1345, puis 2045 à 2049.
- Garantir la sécurisation et la confidentialité de l'échange :
Il est impératif que l'envoi des informations (Passage de mot de passe, transfert des documents) se fasse de manière cryptée. Le protocole HTTP ne remplit pas cette condition. Il faut donc utiliser le mode SSL (Secure Socket Layer) ou HTTPS qui crypte les échanges entre client et serveur.
La confidentialité impose la définition d'un espace utilisateur que nous appellerons " chambre ".
Une chambre est définie par :
Le nom de chambre et un mot de passe pour l'accès du résident.
Un compte et mot de passe pour l'accès en lecture des visiteurs.
Une adresse Email.

Un étage (niveau d'accès):
- R/Ch. : Le résident a les droits d'administration du service. C'est le maître d'Autel.
- Etage 1 et plus : Les possibilités de l'utilisateur sont réduites au fur à mesure que le numéro d'étage augmente.
Les options proposées à chaque étage:
- L'option 1 permet :
La modification du mot de passe du compte Pop associé.
- L'option 2 permet :
La modification de la réservation de la chambre (mot de passe d'accès du résident, adresse Email de résident).
La modification de l'accès visiteur (nom et mot de passe de l'accès visiteur).
- L'option 3 permet :
La modification du compte et mot de passe d'accès des visiteurs.

Pour illustrer, un résident du premier étage dispose des options 1, 2 et 3 alors qu'un résident de l'étage 4, comme par exemple le compte « demo », ne dispose d'aucune option (réservation non modifiable, mots de passe figés)

La taille maximale des chambres sera fixée par un contrôle au niveau du code applicatif et garantie par un quota Unix.

Pour ne pas impacter le poste client, on s'interdira l'utilisation des cookies ainsi que des sessions.

3 Principe de fonctionnement

Le principe est décrit par la figure 2 :

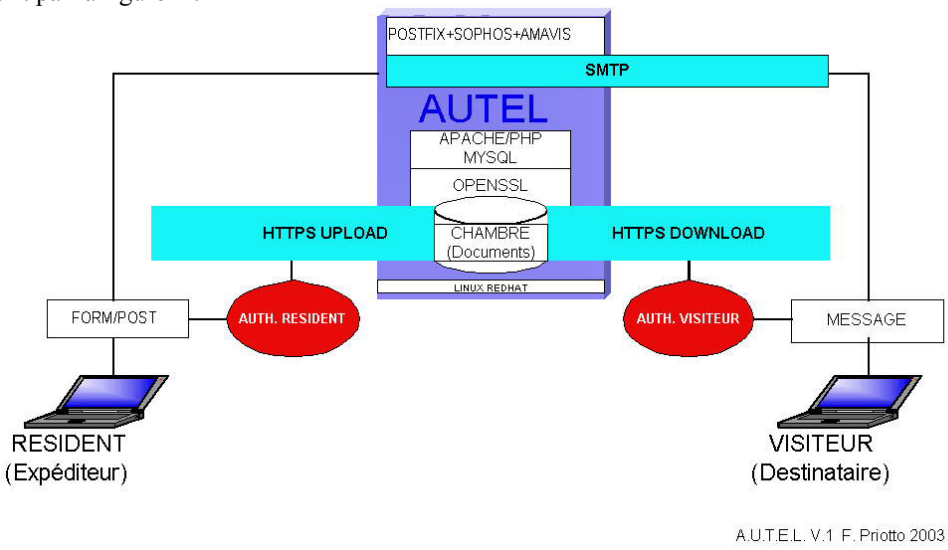

Figure 2 – *Principe de fonctionnement du service A.U.T.E.L.*

Le fonctionnement se base sur la MISE A DISPOSITION du document dans un espace réservé (chambre) sur un serveur via le protocole HTTPS dont un lien dynamique vers l'accès en lecture peut-être communiqué par l'utilisateur (le résident) via l'envoi d'un Email à son correspondant (le visiteur).
La CONFIDENTIALITE de l'accès aux chambres est basée sur la lecture du "fichier d'authentification" contenant une ligne pour chaque résident composée du nom, du mot de passe crypté MD5, de l'adresse de messagerie et de son niveau d'autorisation.
La CONFIDENTIALITE des accès visiteurs est basée sur la lecture "d'un fichier verrou spécifique à chaque chambre" contenant une ligne pour le compte du résident et une ligne pour le compte visiteur.
L'accès au dossier contenant les documents disponibles est interdit à toute session HTTP du fait de la présence d'un fichier .HTACCESS à la racine du dossier stipulant un "deny from all" prohibitif. Rappelons que l'effet d'un fichier .htaccess s'applique sur toute l'arborescence depuis la racine. Seul le script CGI lecture.php permet l'ouverture des documents car il effectue une ouverture locale au serveur et envoi au navigateur le contenu du document après avoir envoyé les en-têtes HTTP nécessaires. Le script de lecture est détaillé en sous-section 3.3.

La mise à disposition d'un document est réalisée en combinant l'Upload du fichier, géré intégralement par PHP* et son enregistrement dans la table des documents de la base de donnée MYSQL.

* Méthode mise en œuvre par la solution WebFTP sur le réseau Math.jussieux.fr et présentée par Jirung Albert SHIH aux JRES 2001.

Le CODE APPLICATIF devra prendre en charge les étapes suivantes :

3.1 L'hébergement

1) Réservation d'une chambre accessible via HTTPS par l'administrateur du service (le maître d'autel) depuis la chambre admin de niveau d'accès 0 (le root du service, pour parler Unix…).
Le niveau d'accès des chambres réservées est ajustable et permet de décliner les options détaillées en sous-section 2.1.
- Vérification du nom de chambre, du mot de passe et de l'adresse Email exécutée sur le serveur (Pour éviter tout javascript).
- Ajout de la ligne utilisateur dans le fichier d'authentification. Le caractère ":" fait office de séparateur et la forme de la ligne est la suivante: nom_espace:mot_pass_crypte_md5:adresse_email:niveau_accès
- Création du dossier associé à la chambre.
- Création du dossier contenant le fichier d'autorisation des visites.
- Création du fichier d'autorisation des visites.
2) Accès à la chambre
- Envoi d'un en-tête HTTP d'authentification par méthode basique via la fonction demande_identite() décrite en annexe.
- L'utilisateur saisi son nom et mot de passe.
- Vérification selon le fichier d'authentification.
- Si l'authentification est correcte : Accès à la chambre
- Récupération dans la base de données des documents de cet espace utilisateur.
- Liste des documents:
Les types de fichiers sont illustrés par une image Icône.
Certains attributs du document sont listés. (Nom original, date de dépôt, taille en octets).
Un indicateur de charge indique l'espace disponible dans la chambre.

3.2 Le séjour

Le résident peut :
- Déposer un document par Upload via HTTPS: Sélection d'un document à la fois par formulaire HTML (Entrée de type FILE, parcours du poste local). Copie du fichier depuis le poste client en cache serveur puis du cache serveur vers la chambre du résident.
- Transmettre des documents : Sélection par entrée type CHECKBOX des documents déjà disponibles ou bien d'un nouveau document par l'entrée de type FILE. Une fois le formulaire validé et traité (Mise en forme des chaîne, validation des adresses de messagerie), un message Email est alors constitué. Ce message inclut un fichier attaché au format HTML qui embarque le lien vers le document disponible en ligne.

Pour comprendre la confection de ce message, il est important de considérer que la structure d'un message SMTP est proche de celle d'une page Web. Elle imbrique des données de types différents (images, texte, scripts…) et l'agent client de messagerie utilise les types MIME pour l'interprétation de ces données tout comme le navigateur.

Initialisation du tableau incluant le contenu du fichier qui sera joint au message. Il constitue un "pointeur" vers la ressource par l'utilisation du META-TAG de re-direction: <META HTTP-EQUIV="Refresh" CONTENT="1; URL=url">. Selon la définition du langage HTML, lorsque HTTP-EQUIV est spécifié au lieu de NAME, le marqueur <META> est destiné à être utilisé comme en-tête HTTP.
La structure du tableau est la suivante: "Nom de part, Contenu, Type MIME".

3.3 L'accès visiteur

L'accès en lecture au document est donc réalisé via l'ouverture du lien embarqué. Un en-tête d'authentification basique est envoyé au navigateur. Le visiteur doit alors rentrer le nom et mot de passe pour l'accès en lecture.
L'URL pointant l'accès en lecture au document sélectionné est de la forme suivante :
https://autel.pharma.univ-mrs.fr/lecture?r=256&c=chambre_resident ou lecture.php est le script PHP d'ouverture en lecture des documents, r=256 donne la référence dans la table documents de la base de donnée et c=chambre_resident donne le nom de la chambre.

3.4 L'authentification

La phase d'authentification est confiée à l'en-tête WWW-AUTHENTICATE qui se charge de récupérer les champs Login et Password et de les rechercher dans le fichier d'authentification pour l'accès des résidents et dans le verrou lecture pour les visiteurs.
Le dossier contenant les documents est verrouillé localement par droits UNIX et n'est pas accessible par connexion http directe. Seul le script de lecture permet l'ouverture des documents via http.

3.5 L'interface d'accès

La figure 5 est une copie d'écran de l'accès à la chambre de démonstration juste avant un envoi de message.
Alors que les autres documents seront transmis par Email sous forme de pseudo pièces jointes, le document doc.pdf sera supprimé de la chambre dans la foulée.
La sélection du choix « Transmettre l'ensemble des documents à vos correspondants » permet de transmettre un pointeur vers la chambre en lecture seule.
Notez que les adresses des destinataires sont séparées par une virgule.

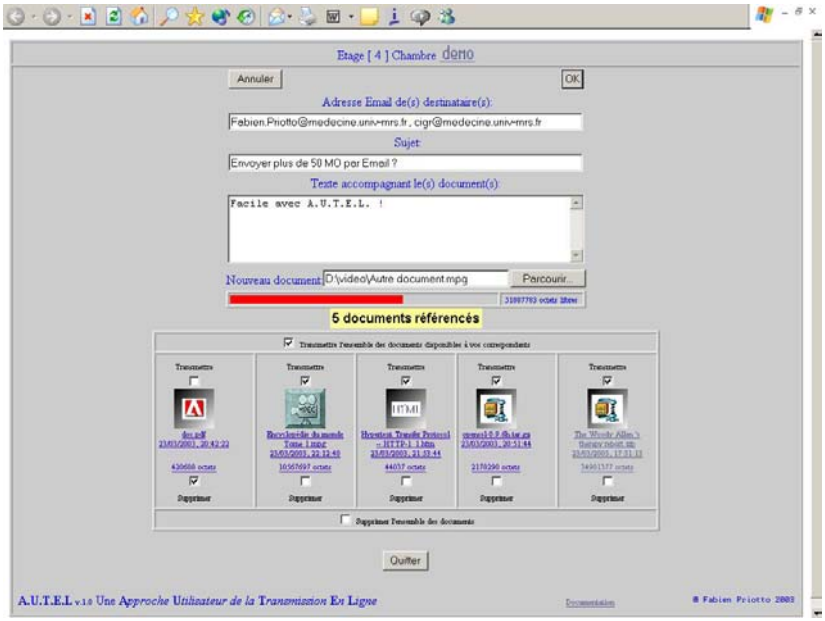

Figure 3 – *L'interface d'accès à la chambre*

Le message transmis est mis en image dans la figure 4 :

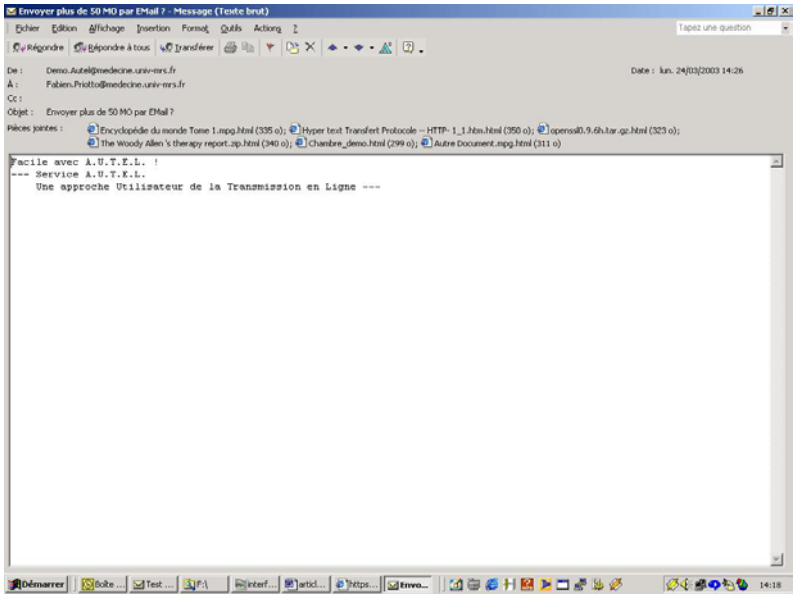

Figure 4 – *Message transmis depuis le service A.U.T.E.L.*

4 Environnement technique

Nous allons décrire un à un tous les éléments techniques qui composent l'architecture retenue pour répondre aux besoins décrits dans le cahier des charges. Voici les principaux :
Système d'exploitation du serveur, serveur Web, protocole sécurisé, système de base de données, langage de script.
L'architecture retenue résulte de l'étroit rapport entre les moyens mis à disposition (matériels, logiciels, temps et argent) et ma technicité personnelle.

4.1 Système d'exploitation : LINUX REDHAT 7.3

Pour un serveur dédié ou une utilisation bureautique et multimédia est exclue, Linux offre une robustesse intéressante et la distribution RedHat 7.3 est suffisamment packagée pour que l'installation ne soit pas pénalisante pour la mise en œuvre du projet. Linux est le système d'exploitation le plus complet du marché. Fruits d'un travail en collaboration, toutes les briques logicielles de Linux sont estampillées au label OSS (Open Source Software) à l'inverse des logiciels propriétaires, un logiciel OSS est libre de diffusion et d'exploitation. Aucune licence n'est requise.

4.2 Langage de script : HTML3.2 + PHP4.3

L'utilisation du langage HTML est incontournable pour constituer l'affichage des pages Web. Cette partie du développement sera volontairement réduite à son strict minimum. L'objectif étant plus de constituer une solution de transmission qu'un véritable site Web.
Outre le fait que PHP est un langage souple et complet qui s'appuie sur les instructions et structures classiques, il est conçu pour s'adapter au besoin d'application Web. Les fonctions et manipulations de données nécessaires au projet sont proposées en standard :
- Procédure d'Upload de fichier pour la mise à disposition des documents.
- Fonctions de traitement des URL pour l'acheminement de l'information.
- Fonctions de gestion des chaînes de caractères pour "traiter » (Echapper) les caractères spéciaux lors du passage entre méthode POST, la base de donnée et les noms de fichiers sous les différents OS.
- Fonctions de gestion des tableaux pour faciliter les algorithmes de traitement.
- Fonctions d'accès aux bases de données.
- Fonctions de messagerie pour envoyer dynamiquement les messages aux visiteurs.

Ainsi PHP procure une syntaxe claire, facile à condenser qui permet de n'écrire que le strict nécessaire. Cela procure une grande facilité pour sauvegarder les code (Il tient sur une disquette !) et surtout pour le maintenir.

4.3 Serveur Web : Apache v.1.3.6

Apache est la référence en matière de serveur HTTP. Il fait partie des logiciels OSS du monde Linux. Son nom vient du fait qu'il est constitué de codes existants ainsi que de certains modules qui lui sont adjoints (Patches)
Le serveur Apache s'adapte selon les besoins, par l'ajout ou la suppression de modules spécifiques. Il est maintenant possible de charger des modules dynamiquement.
Pour notre projet, les modules suivants sont nécessaires :
- Mod_ssl : Ce module permet l'utilisation du système de chiffrement SSL (Secure Socket Layer). Avec ce système, toutes les requêtes sont chiffrées afin d'améliorer la confidentialité. C'est le protocole HTTP sécurisé (HTTPS). Le chiffrement s'effectue sur 128 bits.
Pour une plateforme de production, la mise en place d'un certificat s'imposera.
Au sujet des certificats, voir l'article de Jean-Luc Archimbaud à l'adresse suivante :
http://www.urec.cnrs.fr/securite/articles/certificats.kezako.pdf

- Mod_php : Ce module est nécessaire pour que le langage PHP soit interprété par le serveur HTTP. PHP aurait pu être installé en tant que CGI, mais l'installation en module améliore nettement les performances du serveur. L'interconnexion entre le serveur HTTP et les environnements d'exécution des scripts est ainsi plus efficace. PHP installé en tant que CGI normal, le serveur HTTP crée un nouveau processus, puis lance l'interpréteur PHP, ce qui constitue une opération très coûteuse en temps et en ressources machines. Grâce au module mod_php, ce n'est plus le cas, tout est alors intégré dans le serveur.
- Mod_gzip : Ce module permet la compression « à la volé » du flux http en conjonction avec l'utilisation des fonctions de la libraire ZLIB.

http://www.apache.org

4.4 Base de donnée : MYSQL 3.2

Gratuit et facile à mettre en œuvre, MYSQL est un Système de Gestion de Bases de Données Relationnelles (SGBDR) OSS. PHP offre la fonctionnalité d'accéder à une base MYSQL en mode natif, c'est à dire sans intermédiaire autre que l'API de son middleware associé. Il n'y a pas de couche logicielle à ajouter à l'architecture du système par opposition à un accès mettant en œuvre un pilote ODBC sous Windows.

http://www.mysql.org

4.5 Logiciel Anti-virus : SOPHOS+SWEEP

Sophos Anti-Virus pour Unix est un logiciel de détection et de désinfection de virus. Il vérifie si des virus sont présents dans les systèmes de fichiers locaux et distants ainsi que lors des téléchargements.. Il fonctionne en modes sur accès, à la demande et programmé. Son architecture détermine intelligemment quels sont les fichiers qui doivent subir un contrôle viral, maximisant ainsi la transparence utilisateur et minimisant les pertes de performances.

Sweep est un module de vérification virale sur demande des fichiers. On peut aisément l'activer depuis un script programmé par crontab.

http://www.sophos.com

4.6 Serveur de messagerie : POSTFIX+AMAVIS

Postfix est un MTA présentant une alternative à sendmail qui offre des fonctionnalités intéressantes de lutte anti-spam et de filtrages pour une configuration simplifiée et facile à mettre en œuvre comparée à celle de Sendmail…

http://www.postfix.org

Amavis est un ensemble de scripts Perl qui s'intercalent avant la délivrance des messages par Postfix.
Les message sont scannés puis déposés dans la boite de l'utilisateur s'il ne sont contaminés. Les messages dont le contenu est infecté sont rejetés et mis en quarantaine.
http://www.amavis.org/amavis.html

Tous les détails de l'installation sont donnés par M. Libes à l'adresse suivante :
http://www.com.univ-mrs.fr/ssc/info/cours/install-amavis-postfix.html

5 Détails techniques sur l'accès aux documents

L'accès aux documents utilise es fonctionnalités intéressantes qui distinguent HTTP 1.1 de HTTP 1.0. Ce sont, entre autres, les connexions persistantes, la compression/décompression des données et les transferts de plages d'octets utiles dans le cas d'une reprise de connexion.

5.1 Connexions persistantes

Le serveur ne referme plus systématiquement la connexion une fois qu'il a fini de transmettre la réponse à la requête, afin de permettre l'envoi de nouvelles requêtes qui sont liées à la première réponse.
Par défaut, les connections sont persistantes, mais le client aussi bien que le serveur peut notifier l'autre de son désir de fermer la connexion à la fin de la transaction courante.

Un en-tête Transfer-Encoding: chunked a été introduit pour permettre de continuer à utiliser des connexions persistantes même quand certaines des ressources transmises sont produites dynamiquement.

5.2 Envoi par paquets

Si le navigateur présente le message d'en-tête Transfer-Encoding: chunked, la ressource peut alors être envoyée par blocs précédés chacun de l'indication de la longueur du bloc. Un bloc vide (longueur 0) indique la fin de la ressource.
Alors que le protocole HTTP 1.0 établit systématiquement une nouvelle connexion TCP à chaque requête du client vers le serveur, HTTP 1.1 permet de réaliser des connexions persistantes sans se soucier de la taille du contenu qui sera généré.
Le navigateur doit simplement accepter la valeur CHUNKED pour le champ Transfert-Encoding.

Au niveau des codes retournés, le code de status 100 est envoyé par le navigateur pour autoriser la suite d'un transfert. Le serveur répond par 206 et envoi le contenu partiel.
La figure 5 dévoile la différence de trafic généré sur l'interface réseau du serveur entre un téléchargement classique (pic de consommation de bande passante, première courbe) et un envoi par paquet. L'envoi par paquets procure une meilleure maîtrise de la bande passante.

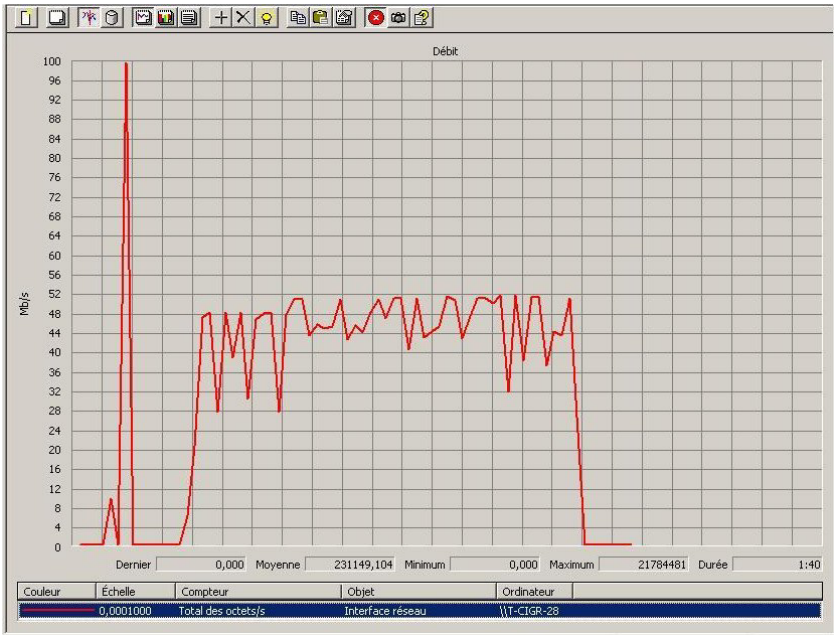

Figure 5 – *Différence de trafic http lorsque les données sont transmises par paquets*

5.3 Compression du flux HTTP

Un nouveau message d'en-tête Transfer-Encoding permet d'indiquer le type d'encodage utilisé à chaque étape du transfert d'une ressource. Le message d'en-tête Content-Transfer-Encoding: permet la compression du contenu entre le serveur et le

client si ce dernier l'accepte. Un type d'encodage de compression comme par exemple GZIP ou DEFLAT peut alors être utilisé pour le transfert des données. Celles-ci seront décompressées à la volée par le navigateur de manière transparente pour l'utilisateur. Selon le type de documents, le flux HTTP peut se trouver alors considérablement réduit puisque le contenu du fichier est envoyé dans son format compressé (GZIP par exemple) et c'est le navigateur, utilisant la CPU du poste client, qui se charge de la décompression. On allège ainsi la charge du serveur et du réseau.

La figure 6 montre l'atténuation obtenue avec un taux de compression maximal sur un gros fichier texte.
La première courbe est celle d'un téléchargement classique et la seconde celle du flux générée par le téléchargement du même fichier avec compression du flux.

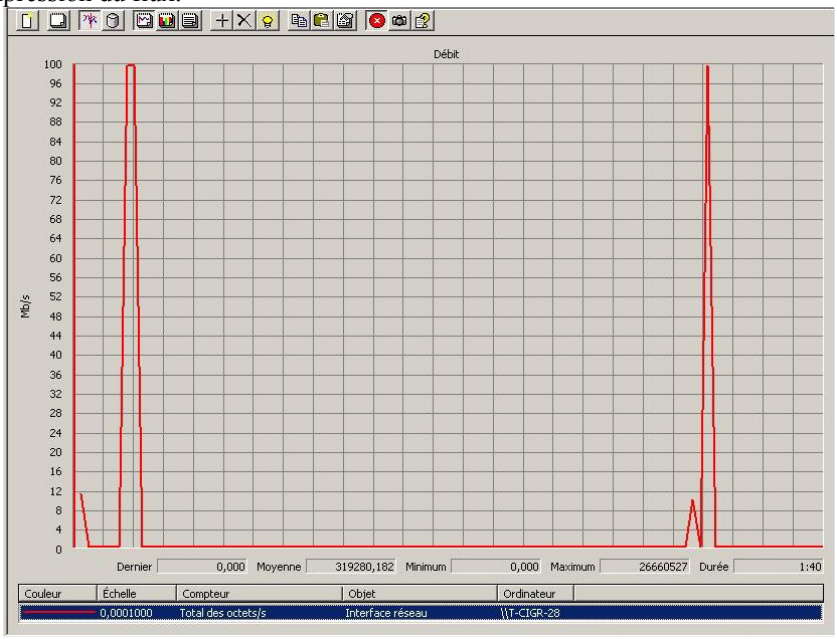

Figure 6 – *Différence de trafic entre un flux http classique et un flux compressé*

Sur le serveur, c'est la directive Apache AddEncoding qui autorise cette décompression "à la volée" par le navigateur. Pour envoyer le contenu de fichiers compressés par la méthode GZIP, le serveur http doit accepter ce mode de compression.

Pour un serveur APACHE, la ligne suivante est requise dans le fichier http.conf:
AddEncoding x-gzip gz
En outre, Apache doit être installé avec le module mod_gzip.
http://www.schroepl.net/projekte/mod_gzip

Malheureusement, tous les navigateurs ne peuvent accepter un flux de données compressées. Il faut donc pouvoir positionner selon le `navigateur la valeur du l'en-tête` Content-Encoding qui sera envoyé juste avant le flux de données.

Un récapitulatif sur les fonctionnalités de décompression GZIP des navigateurs les plus courants est disponible à l'adresse http://www.schroepl.net/projekte/mod_gzip/browser.htm.htm
Parce que le type d'encodage est déterminé en fonction des possibilités du navigateur, il faut préciser lors de l'envoi des en-têtes http que l'en-tête HTTP Content-Encoding est variable. C'est le rôle du message Vary: Accept-Encoding.
Attention, certain navigateur comme par exemple MS IE 4 ne gère pas proprement les en-têtes variables.
Une solution est de forcer les en-têtes à non variable pour les navigateurs n'acceptant pas les en-têtes VARY
BrowserMatch "MSIE 4\.0" force-no-vary
Cette solution ne s'applique que sur les serveurs Apache de version supérieure à 1.3.6.

6 Utilisation de MIME

6.1 Utilité de MIME

MIME, l'extension "Multipurpose Internet Mail Extensions", est une spécification décrivant les formats de messages multimédias sur l'Internet. MIME est disponible et utilisable gratuitement et permet en particulier :
- L'échange de textes écrits dans des jeux de caractères différents,
- L'utilisation de courrier électronique multimédia entre les systèmes informatiques.

Historiquement, la messagerie Internet n'utilise que l'ASCII US (un jeu de caractères codé sur 7 bits) dans les échanges.
Les extensions MIME introduisent deux nouveaux codages pour les données autres que l'ASCII US, appelés Quoted-Printable (QP) et base64. Le premier est destiné à permettre de coder tout alphabet nécessitant plus de 7 bits, le second étant préféré pour les fichiers binaires à transmettre sous forme de pièces jointes ou "attachements" inclus dans les messages. QP code tout jeu de caractères sur 7 bits. Ces codages ont pour principale fonction d'éviter la troncature du 8ème bit par certains serveurs ou routeurs intermédiaires et sont implantés sur toutes les plates-formes existantes. Les deux codages MIME ne sont pas propriétaires (contrairement à BinHex pour la plate-forme Macintosh) et sont universels (à la différence d'uuencode et de ses nombreuses variantes). Si QP peut être endommagé par la traversée de certaines passerelles de courrier utilisant le jeu de caractères EBCDIC, base64 est plus robuste que "uuencode". Les deux codages définis par MIME permettent entre autre de résoudre les problèmes suivants lors du transport du message sur l'Internet :
Conversion des séquences CRLF
Transmission des caractères NULs (US-ASCII 0)
Mauvaise interprétation des tabulations
Troncature des lignes au delà du 76ème caractère
Espaces blancs (tabulation et espaces) excédentaires
Variabilité de l'US-ASCII (base64 n'utilise que 73 caractères fixes)
Corruption des messages due à certaines littérales

6.2 MIME dans le courrier électronique

A l'origine le courrier électronique était prévu pour ne transporter que des textes ASCII, sans aucun accent ni enrichissement, codés sur 7 bits. Pour coder un texte écrit en français, l'ASCII est insuffisant puisqu'il n'autorise la définition que de 33 caractères de contrôle, 52 lettres (minuscules et majuscules), 10 chiffres, les ponctuations et l'espace. Pas de place pour les cédilles, les accents graves ou aigus, les circonflexes et bien d'autres choses encore. Le français, et les langues latines de façon générale, nécessitent un espace de 8 bits pour être codées. Cet "alphabet latin" à été normalisé sous ISO 8859-1 ou ISO-Latin-1.
MIME permet d'utiliser ces alphabets étendus sur 8 bits (et même plus), ouvrant ainsi les possibilités d'envoyer des messages par courrier électronique dans un grand nombre de langues. Pour réaliser cette intégration des jeux de caractères 8 bits, MIME rajoute trois lignes dans l'en-tête de tout message échangé sur l'Internet. L'une décrit le contenu du message (le jeu de caractères ISO 8859-1 par exemple), une autre indique le codage utilisé pour transporter le contenu du message (comme Quoted-Printable) et une dernière signale que MIME est utilisé (Mime-Version: 1.0).

6.3 Conformité MIME

MIME ne restreint pas son champ d'action aux deux formats de codage (Quoted-Printable et base64) qu'il introduit. Un logiciel supportant MIME est capable d'interpréter des messages dont le contenu n'a pas été codé, comme ceux véhiculés en "7 bit", "binary" ou "8 bit". Parmi ceux ci, les deux derniers peuvent être la source de problèmes s'ils traversent des dispositifs du réseau qui tronquent les messages au delà du 8ème bit. MIME recommande QP et base64.

La conformité MIME est un label garantissant que le logiciel est capable de garantir l'interopérabilité des messages MIME. Cela implique que le dit logiciel :
1) Génère toujours le champ d'en-tête "MIME-Version: 1.0" dans tous les messages qu'il crée
2) Reconnaît le champ "Content-Transfer-Encoding" et décode toutes les données reçues lorsqu'elles sont codées soit avec QP ou base64. Toute donnée sur plus de 7 bits, expédiée sans codage QP ou base64, doit être correctement labellée avec un champ content-transfer-encoding à "8bit" ou "binary". Si la couche de transport des données ne supporte pas 8bit ou binary, l'expéditeur du message doit encoder le message et le labeller correctement sous Quoted-Printable ou base64.

3) Reconnaît et interprète le champ d'en-tête "Content-Type". Cela implique que le logiciel évite d'afficher les données brutes lorsque le champ Content-Type contient autre chose que "text". Le logiciel doit au moins être capable d'expédier le texte en spécifiant le type de caractères utilisés dans le message, si ce n'est pas de l'US-ASCII.
4) Manipule explicitement les types de données suivants : Texte (Text), Image, audio et vidéo, Application, Multipart et Message.
5) Traite tout message dont le champ Content-Type est inconnu comme un message de type "application/octet-stream".

Lorsque ces cinq critères sont remplis, l'agent utilisateur ou logiciel client est dit "conforme à MIME".
Bien entendu, pour que MIME soit d'une utilité quelconque, il doit être implanté dans les logiciels utilisés de part et d'autres d'un canal de communication. Ainsi, si deux correspondants, échangeant des messages électroniques souhaitent s'affranchir définitivement des problèmes d'accents dans leurs textes, ils doivent tous deux utiliser un logiciel conforme à MIME.

6.4 Les "Types MIME"

Il existe des types de données MIME officiels et non officiels, ce qui signifie qu'ils reposent ou non sur des accords préalables et privés entre les expéditeurs et les destinataires des données.

Les principaux types de codage sont :
7BIT
BINARY
8BIT
QUOTED-PRINTABLE
BASE64
Autel utilise le codage BASE64 pour crypter l'ensemble du message pour le transport via SMTP.
Cet encodage rend les données illisibles sans décodage et augmente la taille du message d'environ un tiers. L'algorithme convertit un groupe de trois caractères sur 8 bits en un groupe de quatre caractères sur 6 bits. Tous les caractères sont transformés en caractères de la table commune à US-ASCII et ISO 646. Ces caractères sont connus sous la désignation d'alphabet Base64.

7 Conclusion et bilan du maquettage

La mise en œuvre du concept d'A.UT.E.L. a permis de mettre en évidence les subtilités des protocoles SMTP et HTTP qui passent souvent aux yeux des utilisateurs comme des mécanismes simplistes.

La maquette actuellement en test au sein de la faculté de pharmacie permet l'échange inter UFR de documents de taille importante. (Au delà de 100MB)

L'envoi par paquets via connexion persistante offre une certaine fiabilité des transferts qui s'écoulent en arrière plan sans perturber l'utilisation du poste client.

Pour l'instant, seul le mode GZIP a été testé pour la compression du flux. Celui-ci ne s'avère efficace que pour quelques types de documents tels que les fichiers au format texte (Par exemple les fichiers de log système), la compression de fichiers binaires n'apportant évidement presque pas de réduction de trafic.

Un travail de sécurisation du code applicatif et de la plateforme est à entreprendre avant toute mise en production.
Il faudra notamment :

- Veiller à désactiver le mode Register Global et activer le Safe mode de PHP.
- Sécuriser Apache

Un travail de tests approfondis est à réaliser pour vérifier la compatibilité avec l'ensemble des navigateurs ne supportant pas HTTP 1.1.

Annexe

Quelques RFC

[RFC 821] Simple Mail Transfer Protocol

[RFC 822] Standard for the format of ARP Internet Text Messages
[RFC 1867] Form-based File Upload in HTML
[RFC 1869] Simple Mail Transfer Protocol (SMTP) Service Extensions
[RFC 2616]Hypertext Transfer Protocol - HTTP/1.1
[RFC 2617] HTTP Authentication: Basic and Digest Access Authentication
[RFC 1521] MIME (Multipurpose Internet Mail Extensions) Part One :
<< Mechanisms for Specifying and Describing the Format of Internet Message Bodies >>
[RFC 2047] Multipurpose Internet Mail Extensions (MIME) Part Three:
<< Message Header Extensions for Non-ASCII Text>>

Références

[1] Olivier Perret, *GroupWare à l'Institut Pasteur*. Dans *Actes du congrès JRES2001*, pages 5, Décembre 2001
[2] Jirung Albert Shih, *WebFTP: Un client Web sécurisé pour FTP*. Dans *Actes du congrès JRES2001*, pages XX, Lyon, Décembre 2001
[3] Dilip C. Naik, *INTERNET - Standards and Protocols*. Microsoft Press, 1998
[4] Leon Atkinson, Core *PHP Programming*. Prentice Hall, 2001

Glossaire

Base64
Standard de codage défini par MIME. Il permet de transporter les données binaires et les textes exploitants plus de 7 bits sur Internet. Base64 est standardisé contrairement à BinHex ou uuencode, et Base64 résiste mieux aux "déformations" au cours de son transport sur le réseau Internet.

MDA
Mail Delivery Agent
Cet agent joue le rôle de centre de tri et centre de distribution du courrier dans un service de messagerie. Il est prévu pour être relayable.
Exemple de MDA : mail, deliver, procmail, etc.…

MTA
Mail Transport Agent
Agent de transport de courrier. Il réalise l'acheminement et la distribution du courrier
Exemples de MTA : sendmail, Postfix etc.…

MUA
Mail User Agent
Agent de courrier utilisateur. Il s'agit du programme avec lequel l'utilisateur consulte, lit, gère, compose son courrier électronique.
Exemples de MUA : ELM, /bin/mail, Unix Pine, Eudora, Outlook etc.…

Quoted Printable
Codage standardisé dans MIME permettant de coder simplement les textes émis par des agents de courrier MIME. L'imprimable guillemeté ou "Quoted Printable" est une transformation de tous les caractères non ASCII (7 bits) en des "représentations" uniformes (un signe = suivi d'un code correspondant au caractère codé).

RFC
Acronyme de "Request For Comments". Les RFCs sont des documents décrivant les standards pratiqués sur Internet (protocoles, extensions, etc.)

URL
Les repères localisant des ressources, utilisés dans le World Wide Web. Ils se présentent sous la forme méthode-d'accès://hôte:port/chemin