Ensimag 1ère année

TP n°4 – Messagerie : protocoles IMAP, SMTP, et aspects sécurité

Il est recommandé de prendre des notes. Se référer à la version numérique de ce document pour les liens (sur Chamilo).
Une question sur l'effet ou l'utilisation d'une commande ?
man nom_de_la_commande !

1 Partie messagerie

Cette partie de TP illustre quelques protocoles et applications relatifs à la messagerie présentés en cours.
Le TP commence par étudier l'un des protocoles d'accès distant aux boites aux lettres : IMAP. Il considère ensuite SMTP, le protocole de transfert de courrier électronique entre une station cliente et un serveur de messagerie, ou entre deux serveurs de messagerie.

Il aborde enfin la sécurisation des échanges avec le standard S/MIME.

Pour démarrer :

L'ensemble des manipulations de cette partie est à réaliser depuis la session Linux CentOS des machines de TP. Si vous étiez connecté sur une session FreeBSD, redémarrez l'ordinateur et choisissez l'option Linux CentOS.

Les interactions se feront avec votre compte de messagerie `<prénom>.<nom>@grenoble-inp.org`. Vérifiez que vous êtes en mesure d'y accéder depuis Thunderbird, ou depuis Zimbra :
`https://webmail.grenoble-inp.org`

Pour vous assurer de ne pas perdre de données importantes, pensez à archiver vos courriels importants et à conserver une boite de réception propre. Ajoutez des messages de test à votre boite de réception de la façon qui vous conviendra le mieux : envoyez-en vous-même depuis votre adresse ENSIMAG, demandez à un camarade de le faire pour vous, ou utilisez une autre adresse que vous possédez. (Dans ce dernier cas, vous pouvez trouver sur Internet les adresses des différents serveurs IMAP et SMTP : http://www.commentcamarche.net/faq/893-parametres-de-serveurs-pop-imap-et-smtp-des-principaux-fai, par exemple.)

Remarques :

- Lorsqu'un client établit une connexion avec un serveur, il peut le faire en clair. L'ensemble des communications seront visibles sur le réseau (par exemple avec `telnet`). Il peut également former un tunnel sécurisé où tous les messages sont chiffrés de point à point (par exemple avec `openssl`). De nos jours, certaines applications comme Zimbra n'hésitent pas à interdire les communications non sécurisées en bloquant les ports associés.
- Les différents protocoles étudiés ne dépendent pas de la couche de communication : peu importe comment vous communiquez, vous continuerez à respecter les mêmes règles d'interaction. Nous utiliserons ici `openssl`.

1.1 Accès distant aux boites aux lettres avec IMAP

À la différence du protocole POP, le protocole IMAP permet de manipuler les messages directement sur le serveur de messagerie. Il est par exemple possible de créer des dossiers sur le serveur IMAP (des boites aux lettres) et de déplacer des messages d'un dossier à un autre sans en télécharger le contenu.

Chaque commande est précédée d'un index unique sous forme d'une chaine de caractère. Ceci permet dans certaines conditions d'envoyer au serveur une nouvelle commande alors que l'on n'a pas encore reçu la réponse de la précédente.

On utilisera principalement les commandes `LOGIN`, `SELECT`, `FETCH`, `STATUS`, `CREATE`, `COPY` et `LOGOUT`. Vous trouverez leur mode d'emploi en consultant la RFC 3501 disponible à l'adresse suivante : http://tools.ietf.org/html/rfc3501.

Cette documentation se présente de la manière suivante, pour les sections consacrées aux commandes.

`Arguments:` Une liste des différents arguments à utiliser.

`Responses:` Les réponses qui sont données à la requête, s'il y en a.

`Result:` La signification qu'ont les codes d'erreur que l'on trouve en fin de réponse. Suit une description du fonctionnement de la commande.

`Example:` Des exemples d'utilisation sur des cas courants avec le résultat obtenu.

Remarques :

- Dans toute la suite, faites bien attention à respecter la syntaxe du protocole IMAP : `index COMMANDE PARAMETRES`. Il n'est pas nécessaire pour ce TP de choisir un index différent à chaque commande (vous serez le seul à accéder à votre compte de messagerie). Il est d'usage de choisir comme `index` un simple caractère, comme un unique `« . »`
- Votre mot de passe sera à nouveau visible dans votre terminal, nous vous conseillons d'utiliser rapidement la commande de « choix de boite de réception » qui est assez verbeuse : `. SELECT inbox` ou de changer de mot de passe pour la session.

1. Le serveur IMAP de Zimbra se trouve sur la même machine `webmail.grenoble-inp.org`. Par défaut, le serveur écoute en clair sur le port 143 et en chiffré sur le port 993.
   Créez une connexion interactive sécurisée entre votre machine et le serveur via la commande `openssl s_client -connect webmail.grenoble-inp.org:993`.
2. Identifiez-vous sur le serveur au moyen de votre adresse électronique et de votre mot de passe. Vous devrez pour cela utiliser la commande IMAP `LOGIN`.
3. Identifiez le nombre de messages contenus dans la boite aux lettres `inbox`, à l'aide de la commande `SELECT`. *Combien d'entre eux sont dans l'état non-lus ?*
4. Lisez le contenu de quelques messages de votre `inbox` à l'aide de la commande `FETCH` et repérez l'identifiant d'un de vos courriels de tests.
5. Supprimez un courriel de test en lui ajoutant le flag `\Deleted` à l'aide de la commande `STORE`. *Le message a-t-il été supprimé du serveur ?* Si non, trouvez une commande permettant la suppression définitive de ce message.
6. Utilisez la commande `CREATE` pour créer un dossier intitulé `Archives`. Déplacez-y quelques messages depuis votre boite de réception. Marquez-les comme comme lus.
7. Déconnectez-vous enfin du serveur IMAP.

Informations :

— IMAP accède à la boite de réception (mailbox) mais également aux différents dossiers d'archives.
— L'intérêt principal de IMAP est de pouvoir accéder à ses courriels depuis différents clients de messagerie. Les données étant stockées sur le serveur, l'UA agit uniquement comme interface pour afficher les messages.
— Le principal inconvénient est que IMAP nécessite une connexion permanente pour interagir avec les données sur le serveur distant.

1.2 Transfert de courrier électronique avec SMTP

Le protocole SMTP sert à transférer du courrier électronique entre une station cliente et un serveur de courriel ou entre deux serveurs de courriel. La RFC 2821 pour ce protocole est disponible à cette adresse : http://tools.ietf.org/html/rfc2821. Pour obtenir les informations sur l'utilisation des différentes commandes, vous pouvez vous référer à cette RFC ou bien aux transparents du cours.

1. Trouvez les serveurs de messagerie associés au nom de domaine `grenoble-inp.org`, en utilisant une requête DNS.
2. *Quel(s) port(s) (par défaut) utilise un serveur SMTP ?*
3. Dans les questions suivantes, vous allez effectuer un envoi de courriel à vous-même sous une identité arbitraire, à l'aide du serveur SMTP.
   De façon générale, avez-vous le droit de réaliser une telle manipulation ?
   Attention : Faites bien attention à envoyer ce message à vous-même. L'utilisation d'une autre adresse de destination constituerait un délit, et engagerait votre responsabilité (LOPPSI du 14 Mars 2011) :
   « Le fait d'usurper l'identité d'un tiers ou de faire usage d'une ou plusieurs données de toute nature permettant de l'identifier en vue de troubler sa tranquillité ou celle d'autrui, ou de porter atteinte à son honneur ou à sa considération, est puni d'un an d'emprisonnement et de 15 000 € d'amende. » « Cette infraction est punie des mêmes peines lorsqu'elle est commise sur un réseau de communication au public en ligne ».
   En cas de doute, demandez à vos enseignants avant d'envoyer le message.
4. Créez une connexion interactive non sécurisée entre votre machine et un des serveurs de messagerie de `grenoble-inp.org`
5. Saluez le serveur, puis envoyez un message à vous-même paraissant venir de James Bond et dont l'objet serait « Compte rendu de mission ». Pour cela, servez-vous seulement des commandes SMTP offertes par le serveur.
   Commencez par préciser l'expéditeur (`MAIL FROM`), le destinataire (`RCPT TO`) et enfin le corps du courriel (DATA).
   Assurez-vous que l'adresse d'expéditeur dans `MAIL FROM` utilise un nom de domaine existant, autrement votre email ne sera pas transmis.
   Vous pouvez trouver l'ensemble des commandes à partir des transparents du cours, sur Internet, ou grâce à la RFC2821 mentionnée en introduction.
6. Lancez une capture de paquets avec Wireshark et envoyez un nouveau mail avec la même technique que précédemment. Analysez le résultat de la capture, en utilisant par exemple le filtre « smtp ».
   Que pouvez-vous dire sur le niveau de sécurité de la communication ?
   Il est possible de demander à utiliser une session sécurisé avec le serveur SMTP, grâce à la commande `STARTTLS`. Pour cela, on peut de nouveau utiliser `openssl`, mais en précisant qu'on souhaite établir une session sécurisée avec `STARTTLS` :

```
openssl s_client -starttls smtp -connect <hôte>:<port>
```

7. Trouvez les serveurs de messagerie du domaine `univ-grenoble-alpes.fr`. Lancez une capture de paquets avec Wireshark, et connectez-vous en `STARTTLS` sur le port 25 à un de ces serveurs de messagerie. Analysez ensuite la capture.
   La communication avec le serveur est-elle entièrement chiffrée ? Comment fonctionne ***`STARTTLS`*** *?*
8. Testez avec les serveurs de messagerie associés à `grenoble-inp.org`.
   Que se passe-t-il ?
9. En utilisant un client de messagerie plus évolué, comme la messagerie Zimbra ou Thunderbird, envoyez-vous un courriel contenant une image GIF récupérée sur Internet.
   Consultez le message en vous connectant directement au serveur IMAP.
   Comment l'image est-elle véhiculée ?

Informations :

— SMTP spécifie l'expéditeur ainsi que le destinataire d'un message, vérifie les noms de domaines puis envoie le message. La transmission du message depuis le serveur de messagerie expéditeur jusqu'au serveur de messagerie destinataire se fait grâce aux enregistrements MX des serveurs DNS.

— Le protocole peut être mis en place de manière peu sécurisée. Il faut être conscient que votre message peut circuler en clair sur le réseau.

1.3 Sécurisation des échanges de bout en bout avec S/MIME

À travers l'étude des protocoles SMTP et IMAP, nous avons montré que le service de messagerie standardisé sur Internet présente de nombreuses lacunes en matière de sécurité (confidentialité, authentification, intégrité). L'utilisation de tunnels TLS ou SSH offre une solution partielle à cette problématique dans la mesure où ils ne garantissent pas la sécurité du contenu, mais seulement celle de certaines connexions.

Dans le but d'assurer une sécurité des échanges de messagerie de bout-en-bout, c'est-à-dire entre un émetteur et un destinataire, nous allons utiliser le standard S/MIME. Il s'agit d'une extension du format de message qui permet d'introduire des procédés cryptographiques tels que le chiffrement ou la signature. S/MIME repose sur l'utilisation de chiffrement asymétrique de sorte que chaque utilisateur dispose d'une clé privée et d'une clé publique. Cette dernière est certifiée par un tiers de confiance, appelée autorité de certification (AC).

En utilisant ce système de sécurisation, on obtient, entre autres, les propriétés suivantes :

— Signature digitale : en utilisant votre clé privée, vous pouvez signer vos messages et donc certifier que vous en êtes l'auteur.

— Chiffrement par clé : en chiffrant le message avec la clé publique de votre destinataire, il est le seul à pouvoir déchiffrer et lire le message.

Pour cet exercice, vous allez travailler en binôme pour échanger des courriels.

Remarque :

— Si vous n'avez pas encore mis en place Thunderbird sur votre Linux CentOS, faites-le rapidement en suivant les instructions du wiki : http://ensiwiki.ensimag.fr/index.php/Thunderbird

— À la fin de la séance, pensez à supprimer les clés que vous avez ajoutées à Thunderbird et à révoquer la clé demandée sur Comodo (vous pouvez toujours en demander une nouvelle à l'avenir).

1.3.1 Obtention d'un trousseau de clés

Comme pour la sécurisation de site internet avec HTTPS, la sécurisation d'échanges de courriels avec S/MIME repose sur une hiérarchie de certificats avec les racines considérées comme valides par la plupart des applications. Chaque utilisateur d'une organisation se voit attribué un couple de clé asymétriques et une certification signée par le certificat de l'oganisation. L'ensemble est appelé un « porte-clés » numérique.

Dans le cas d'un élève du groupe grenoble-inp, il est possible de demander une telle certification auprès de votre école mais cette démarche peut être longue. Pour ce TP, nous allons obtenir un « porte-clés » gratuitement à partir d'un organisme privé.

1. Rendez-vous sur le site de Comodo et remplissez le formulaire pour obtenir un trousseau de clés. Vous devez y renseigner vos nom et prénom, votre courriel, votre pays et un mot de passe pour révoquer votre clé. N'oubliez pas de décocher le « Comodo Newsletter ».
   https://secure.comodo.com/products/frontpage?area=SecureEmailCertificate
   Au terme de ces opérations, vous recevrez un courriel de confirmation (environ 5 minutes).

2. Récupérez votre trousseau en ouvrant le lien du courriel de confirmation de Comodo depuis Firefox. Il est alors ajouté à votre navigateur. Récupérez le sur votre machine en suivant ces étapes :
   Dans Firefox, cliquez sur le bouton en haut à droite pour ouvrir le menu, choisissez `Préférences`. Ouvrez le maitre onglet `Avancé`, puis le sous-onglet `Certificats`. Cliquez sur `Afficher les certificats`. Dans la nouvelle fenêtre, choisissez `Vos Certificats`, sélectionnez le certificat dans la liste et faites `Sauvegarder`. Sauvegardez le dans un nouveau dossier `tp4` de votre `home` avec pour nom `trousseau.p12` (choisir un mot de passe simple). Supprimez-le ensuite de votre navigateur avec l'option `Supprimer`.
3. Un trousseau de clés sous forme de fichier `.p12` est une archive sécurisée contenant la clé publique, la clé privé et le certificat de l'utilisateur. Parfois le certificat de l'autorité de certification se trouve dans l'archive.
   Extraire de `trousseau.p12` le `certificat.crt` avec la commande `openssl` suivante (ne pas mettre de mot de passe à votre certificat) :
   ```
   openssl pkcs12 -in trousseau.p12 -out certificat.crt -clcerts
   ```
4. Consultez le certificat d'utilisateur avec la commande `cat certificat.crt`. *À votre avis, comment les données de votre certificat sont-elles codées ?* Au besoin, faites une recherche sur Internet pour vous renseigner sur les certificats x509.
5. Utilisez la commande `openssl x509 -text -in certificat.crt` pour obtenir le contenu du certificat sous forme textuelle. *Quel est le type du certificat ? Pourra-t-on l'utiliser pour d'autres applications que le chiffrement et la signature de courriels avec S/MIME ? Quelle est sa durée de validité ?*

1.3.2 Mise en place des certificats utilisateur et échanges sécurisés

À présent, les utilisateurs de chaque binôme dispose chacun d'un porte clés (fichier .p12) contenant le certificat de l'autorité de certification. Nous allons utiliser le client de messagerie Thunderbird pour mettre en œuvre les fonctionnalités de chiffrement et signature des messages proposée par S/MIME.

1. Il faut maintenant importer le trousseau dans Thunderbird. Pour cela, effectuez un clic droit sur le compte de messagerie avec lequel on souhaite utiliser S/MIME, puis choisir l'option `Paramètres`

(fig. 1a). Dans le sous-menu `Sécurité`, sélectionnez `Afficher les certificats` (fig. 1b) puis dans l'onglet `Vos certificats`, importez votre certificat à l'aide de votre trousseau (fig. 1c).

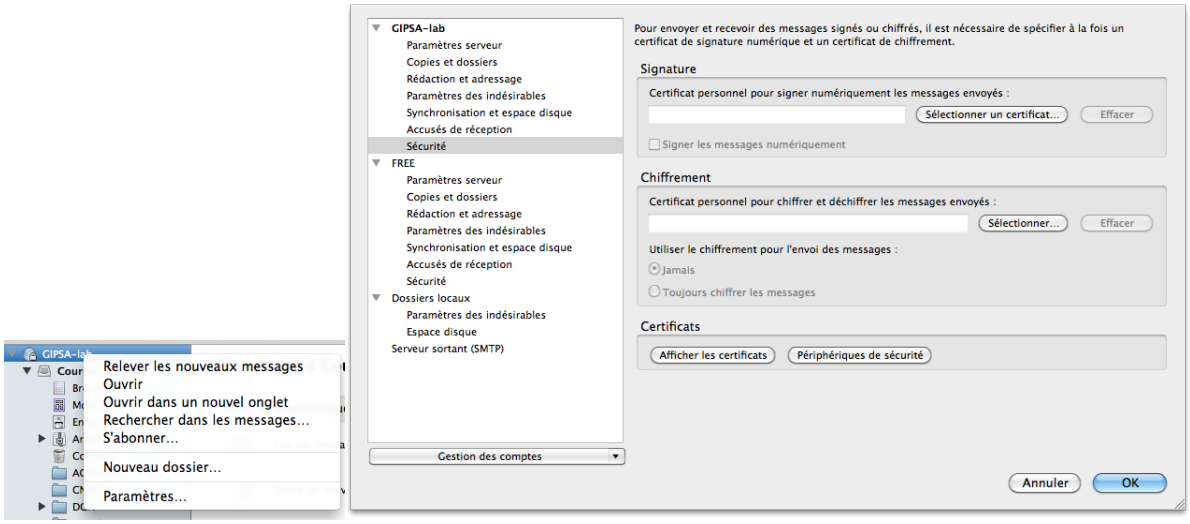
2. Dans les paramètres du compte de messagerie (rubrique Sécurité) avec lequel vous souhaitez utiliser S/MIME, associez votre nouveau certificat aux fonctions de chiffrement et de signature (fig. 1b).
3. Envoyez un premier message signé à votre binôme. Pour cela, rédigez un courriel puis sélectionnez depuis l'interface du courriel `Sécurité` > `Signer numériquement ce message`.
4. Lorsque vous recevez un message signé, il doit apparaitre avec une enveloppe scellée. En cliquant sur cette enveloppe, vous obtenez des informations sur la signature de l'auteur. Affichez le code source du message depuis Thunderbird et retrouvez la signature. *Quel est le type MIME qui lui est associé ?*
5. Envoyez un message chiffré à votre binôme. Si vous ne parvenez pas à chiffrer le message, cela est certainement dû au fait que vous ne possédez pas le certificat de votre binôme (donc sa clé publique). Assurez-vous d'avoir bien reçu un courriel signé venant de lui.
6. Consultez le courriel provenant de votre binôme depuis Thunderbird et assurez vous qu'il est chiffré (une icône en forme de cadenas indique le chiffrement).
   Pouvez-vous lire le contenu ? Pourquoi ?
7. Consultez ce même courriel depuis l'interface de la messagerie de Zimbra.
   Pouvez-vous lire le contenu du courriel ? Pourquoi ?

Informations :

— Utiliser le standard S/MIME nous donne des propriétés très importantes pour la communication entre individus : l'integrité, l'autentification, la non-répudiation et la confidentialité. Il nécessite cependant une autorité de certification (CA) pour délivrer les certificats aux utilisateurs.

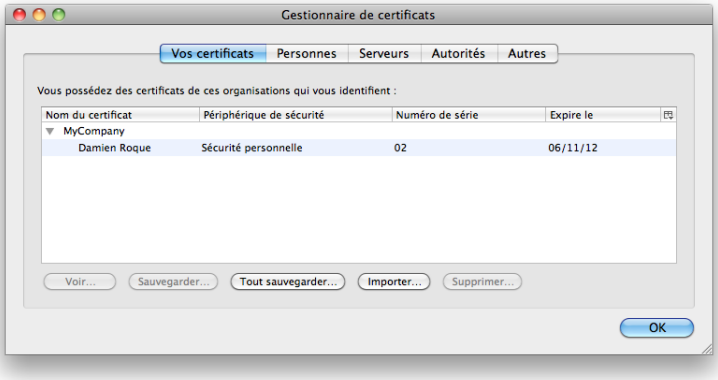
— Il est possible de créer sa propre autorité de certification auto-signée pour générer des trousseaux de clés. Cependant Thunderbird ne la reconnaitra pas par défaut.

— Le standard détermine comment les chiffrements et signatures doivent être traités par les acteurs de la communication. Un certificat est délivré pour une utilisation précise : sécurisation web, sécurisation de communication des courriels...



(a) Accès aux paramètres d'un compte de messagerie.

(b) Paramètres de sécurité d'un compte de messagerie.



(c) Importation d'un certificat d'utilisateur.

FIGURE 1 – Captures d'écran pour la configuration de S/MIME dans Thunderbird.