Correction Examen 2011-2012

Exercice 1 :

- Un proxy est un intermédiaire entre 2 hôtes/machines.
- L'utilisation d'un proxy est transparente. Donc je pense que le code des applications n'a pas besoin d'être modifié.

- Un proxy sert d'intermédiaire pour accéder à un autre réseaux.
  Il peut servir à :
 - appliquer un filtrage sur un site web par exemple.
 - activer des options de mémoire cache, compression, filtrage publicités...
 - journalisation des requêtes.
 - sécurité.

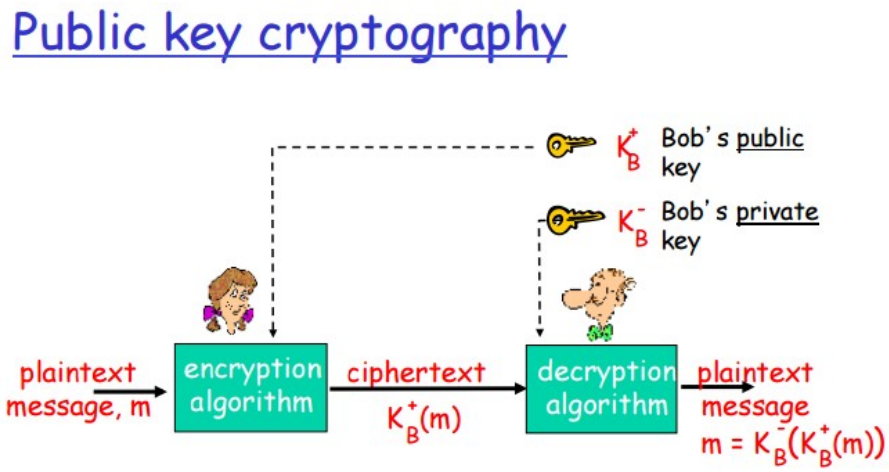
- Différences :
  - Le proxy fournit des services qui ne concernent pas seulement la sécurité comme vu dans le point précédent.

- Ressemblances :
  - Les deux peuvent faire office de contrôle des connexions entrantes et sortantes.
    Par exemple, empêcher les connexions vers un site, ou encore empêcher les connexions venant d'un autre pays.

Exercice 2 :

Question 1 :

- L'utilisation des clés publiques implique deux clés. Cette méthode ne nécessite pas de partager un « secret » au préalable contrairement au cryptage symétrique. La clé publique est connue de tout le monde. La clé privée est connue seulement par le receveur.



Dans ce schéma, pour décrypter le message envoyé par Alice,

Bob applique la formule suivante pour décrypter le message de Alice :

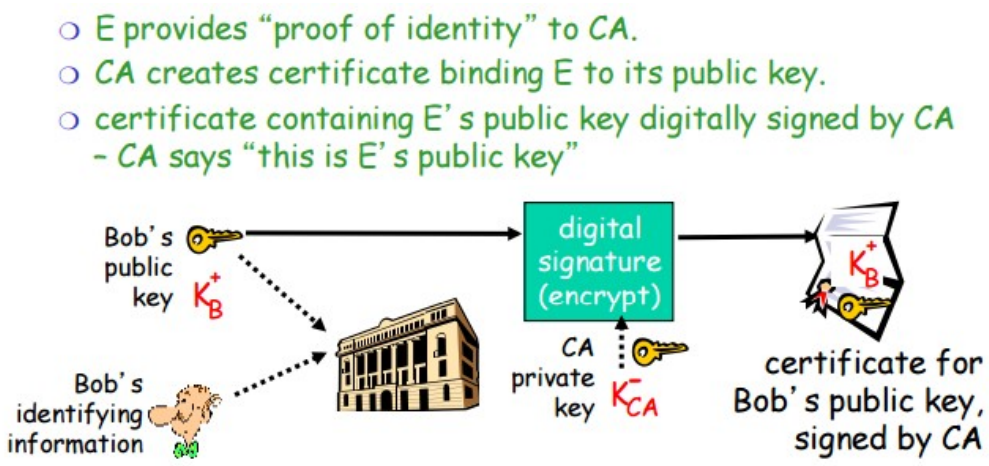
messageDecrypté = ClePrivéeBob (CléPubliqueBob(messageCryptéReçu))
OU
messageDecrypté = ClePubliqueBob (CléPrivéeBob(messageCryptéReçu))
=> Les 2 méthodes de décryptage donnent le même message.

- La non répudiation :
  Wikipédia : « la non-répudiation signifie la possibilité de vérifier que l'envoyeur et le destinataire sont bien les parties qui disent avoir respectivement envoyé ou reçu le message. Autrement dit, la non-répudiation de l'origine prouve que les données ont été envoyées, et la non-répudiation de l'arrivée prouve qu'elles ont été reçues. »

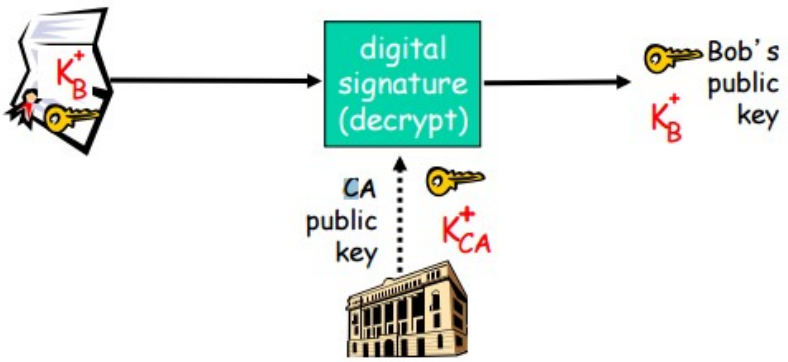
- RSA assure ces propriétés.
  Détails sur la création des clés publiques et privées : cours 7 – page 34.

Question 2 :

Un CA (Certification authority) lie une clé publique à une certaine entité E.
Cette entité enregistre sa clé publique avec le CA.



Si Alice veut la clé publique de Bob. Elle obtient le certificat de Bob (par Bob ou d'un autre moyen). Elle applique la clé publique délivrée par l'autorité de certification sur le certificat de Bob, ce qui donne la clé publique de Bob l'éponge.



Un certificat contient :
- Le nom de l'émetteur.
- Nom, adresse, domaine… de l'entité.
- La clé publique de l'entité.
- La signature digitale qui a été signée avec la clé privée de l'émetteur.

Question 3 :

Je trouve pas de réponses dans le cours,. Peut-être cette partie n'est plus dans le cours depuis 2011 !

Question 4 :

Je suis pas sûr !
Oui en utilisant la méthode « Message Authentication Code (MAC) ».
Cela permet de vérifier l'intégrité des messages envoyés mais aussi d'authentifier l'expéditeur.

Question 5 :

- Le codage symétrique reste utile malgré le système de clé publique car il est beaucoup plus rapide que ce dernier. Par exemple DES (un codage symétrique) est 100 fois plus rapide que RSA (un code avec clé publique). Cela est dû aux calculs plus complexes effectués par RSA. Bien sûr RSA est beaucoup plus sécurisé (infaillible à ce jour).
- Pour allier rapidité et sécurité, on utilise souvent le système suivant.
Alice et Bob se mettent d'accord sur une clé symétrique à l'aide du protocole RSA (ou Diffie-Hellman qui est un autre protocole de clé publique). Puis ils utilisent cette clé symétrique pour crypter leurs échanges.

Exercice 3 :

- Modèle client-serveur :

- Le serveur est toujours en route. Il attend les requêtes des clients.
Wikipédia :
> « il attend une connexion entrante sur un ou plusieurs ports réseaux locaux ;
> à la connexion d'un client sur le port en écoute, il ouvre un socket local au système d'exploitation;
> suite à la connexion, le processus serveur communique avec le client suivant le protocole prévu par la couche application du modèle OSI. »

- Le client envoi des requêtes au serveur afin que ce dernier les traite, envoi une réponse…
Wikipédia :
> « il établit la connexion au serveur à destination d'un ou plusieurs ports réseaux ;
> lorsque la connexion est acceptée par le serveur, il communique comme le prévoit la couche applicative du modèle OSI. »

- Pair-à-Pair :

Chaque client joue le rôle à la fois du serveur et du client comme on les a définis dans le modèle client-serveur.
Ainsi un client est à la fois en attente des requêtes d'autres clients afin d'y répondre (mode serveur). Mais il peut lui aussi faire une requête auprès d'un autre client (mode client).

- Connexions avec sockets :

Oui cela a du sens.

- 1er exemple :
  En utilisant le protocole Gnutella permettant d'échanger des fichiers.
Cette méthode est complètement décentralisée, il n'y a pas de serveur principal. Les clients utilisent des connexions TCP avec leurs voisins (ou bien avec leur « leader » avec la méthode KaZaA).

- 2e exemple :
  Modèle centralisé. Les clients communiquent avec des sockets TCP vers le serveur pour obtenir les informations disponibles sur les autres clients.

Exercice 4 :

Question 1 :

DNS (Domain Name System) permet de traduire un nom de domaine en adresse(s) IP.
Par exemple www.UnBigMacPourSandrine.com ---> DNS ---> 33.35.185.172

C'est une base de données distribuée et hiérarchique.

Exemple :
Si un client veut récupérer l'adresse ci-dessus, il procède de la façon suivante.
1 - Il contacte un des serveurs root de DNS afin de trouver le .com DNS serveur.
2 – Le client contacte le .com DNS serveur afin de trouver le DNS serveur de l'adresse « UnBigMacPourSandrine.com »
3 – Le client contacte le DNS serveur de « UnBigMacPourSandrine.com » qui lui renvoi l'IP de cette adresse.

Sécurité : A venir...

Question 2 :

Question 3 :