Numérique et science informatique
Classe de terminale

IV Architecture et réseaux
B Réseaux et internet
3. Sécurisation des communications

I Le problème à résoudre et notions fondamentales

I.1 Le problème à résoudre

• Un problème à résoudre : la sécurisation des communications

I.2 Notions fondamentales

• coder et décoder

• chiffrer et déchiffrer (on proscrira l'anglicisme « crypter »)

• qu'est-ce que décrypter ?

• cryptologie, cryptographie et cryptanalyse

II Méthodes de chiffrement symétrique

Un texte, si on le voit d'abord comme des chaînes de caractères, peut toujours être représenté par un nombre entier positif. En effet, tout texte pourra être représenté comme une succession de bits qui à son tour peut constituer l'écriture binaire d'un nombre entier positif. Nous pouvons donc traiter toutes les informations (message, message chiffré, clé, etc.) comme des nombres entiers.

II.1 Le principe de la méthode de chiffrement symétrique

• Dans une méthode de chiffrement symétrique, on dispose de deux fonctions C et D :
  * qui prennent comme argument deux nombres ;
  * et dont la valeur de sortie est un nombre.

• On suppose de plus que ces deux fonctions C et D possèdent entre elles la relation suivante :

Si le nombre s est l'image des nombres m et k par la fonction C alors m est l'image de s et k par la fonction D. Avec des notations mathématiques classiques, soient les nombres entiers m, k et s

$$\text{Si } C(m,\ k) = s \text{ alors } D(s,\ \ k) = m$$

• Dans cette formalisation,
  * la fonction C représente la **méthode de chiffrement** ;
  * la fonction D représente la **méthode de déchiffrement** ;
  * Le nombre m représente le **message « en clair »** ;
  * le nombre s représente le **message chiffré** ;
  * Le nombre k représente la **clé de chiffrement**, c'est-à-dire un paramètre supplémentaire qui conditionne la manière dont on va mettre en œuvre la méthode de chiffrement.

• Dans une méthode de chiffrement symétrique, on a la propriété suivante.

Le *même paramètre* k (la même clé) qui est utilisé pour chiffrer le message en clair m (avec la fonction C) et pour déchiffrer le message chiffré s (avec la fonction D).

• On part du principe que la méthode de chiffrement utilisée (c'est-à-dire le couple formé par les fonctions C et D) est bien connue des attaquants. Dans une méthode de chiffrement symétrique, il est donc vital que la clé k qui sert à chiffrer et à déchiffrer les messages :

* soit *connues à la fois* de l'expéditeur et du récepteur du message ;
* qu'elle soit *inconnue* de toute autre personne qui pourrait souhaiter espionner la communication ;
* et qu'elle soit très difficile à retrouver par « essai et erreur ».

• Les deux interlocuteurs de la communication doivent donc convenir ensemble de la clé qui sera utilisée lors de la communication tout en faisant en sorte qu'elle soit ignorée de tout autre personne. On parle donc aussi de **méthode à clé secrète**.

II.2 Chiffrement par décalage (ou de César)

• Cette méthode est très ancienne puisqu'elle était déjà utilisée par l'empereur romain Jules César pendant l'Antiquité.

• On applique à chaque lettre du texte un décalage de n positions des lettres de l'alphabet (où n est un entier compris entre 1 et 25) de façon cyclique (c'est-à-dire que l'on revient au début si l'on dépasse la fin de l'alphabet). Par exemple si $n = 5$ alors les '`a`' deviendront des '`f`', les '`u`' deviendront des '`z`' et les '`y`' deviendront des '`d`', etc..

• Cette méthode de chiffrement est une méthode symétrique puisque la seule connaissance du nombre de décalage n permet de chiffrer mais aussi de déchiffrer le message. Ce nombre n est donc la clé de chiffrement qui doit rester secrète.

• Il s'agit d'une méthode très rudimentaire et très facile à décrypter puisque, si on suppose que la méthode de chiffrement utilisée sur un message donné est la méthode de décalage et la clé clé est n c'est-à-dire le nombre de décalage qui ne peut prendre que 25 valeurs possibles. Il est donc facile de les essayer une par une sur un message chiffré donné.

II.3 Chiffrement par XOR

a) L'opérateur binaire XOR

• L'opérateur binaire $\oplus$, parfois aussi appelé XOR, est le « ou exclusif ». Sa table de vérité est la suivante :

A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

• Qu'est-ce qu'**appliquer l'opérateur $\oplus$ bit à bit à** deux successions de bits de même longueur ? Un exemple :

$$\begin{array}{r} 01101110 \\ 00110010 \\ \hline 01011100 \end{array}$$

Donc, $01101110 \oplus 00110010 = 01011100$

• L'opérateur $\oplus$ possède une propriété intéressante : si A, B et C sont une succession de bits de même longueur tels que $A \oplus B = C$ alors $C \oplus B = A$. Autrement dit, on a :

$$(A \oplus B) \oplus B = A$$

Appliquer deux fois l'opération ...$\oplus B$ ramène à la valeur initial. On dit que cette opération est **involutive**.

a) La méthode de chiffrement XOR

• La *méthode de chiffrement* par XOR suit les étapes suivantes :

* On définit une clé k directement comme une succession de bits ou bien comme un texte que l'on convertit ensuite en binaire. La clé k est donc une succession de n_k octets
* On convertit le message à chiffrer m en binaire : m est alors lui aussi représenté par une succession de n_m octets.
* On ajuste la longueur de la clé à la longueur du message de la façon suivante :
  - Si $n_m \leq n_k$ on tronque la clé de façon à ne garder que les n_m premiers octets ;
  - Si $n_m > n_k$ on duplique la clé de façon à obtenir n_m octets.
* On applique l'opérateur $\oplus$ sur m et k de façon à obtenir le message crypté s :

$$m \oplus k = s$$

• La *méthode de déchiffrement* consiste alors simplement à appliquer à nouveau l'opérateur $\oplus$ à s et k de façon à obtenir le message initial m :

$$s \oplus k = m$$

m se présente alors sous la forme d'une succession d'octets qui peut ensuite être facilement convertie en un texte.

• Contrairement à la méthode par décalage, la clé de chiffrement peut prendre un très grand nombre de valeur. Si elle est suffisamment longue, il est très difficile de la trouver par « essai et erreur ».

III Méthodes de chiffrement asymétriques

• Les méthodes de chiffrement asymétriques peuvent être illustrées par les méthodes de chiffrement à clé publique et privée.

• Ces méthodes reposent sur l'idée que l'on peut concevoir des techniques de chiffrement utilisant une paire de clés $(k_1 ; k_2)$. Ces clés sont différentes et jouent des rôles complémentaires :

* la clé k_1 permet de *chiffrer*, c'est-à-dire d'obtenir le message chiffré s à partir du message en clair m :

$$s = C(m , k_1)$$

* la clé k_2 sert à *déchiffrer*, c'est-à-dire à retrouver le message en clair m à partir du message chiffré s :

$$m = D(s , k_2)$$

• Pour que cette technique fonctionne, il est nécessaire que :

* la connaissance du message chiffré s et de la clé k_1 ne permet pas de retrouver le message initial m ;
* la connaissance de la clé k_1 ne permet pas de retrouver la clé k_2 qui lui est associée.

• Puisque seule la clé , k_2 permet de déchiffrer le message, la clé k_1 n'a pas besoin d'être protégée et elle peut alors être connue de tout le monde. On dit donc que k_1 est la **clé publique**.

• La clé k_2 doit rester secrète mais l'expéditeur du message n'a pas besoin de la connaître. Il est en effet suffisant que le *destinataire* du message chiffré connaisse k_2 puisque lui seul doit déchiffrer le message. On dit alors que k_2 est la **clé privée**.

• On peut se représenter la clé publique comme un cadenas ouvert et la clé privée comme la clé du cadenas. Le cadenas permet d'enfermer des objets dans des boîtes mais non d'ouvrir les boîtes fermées avec un cadenas identique.

• Illustrons le processus suivi dans le contexte d'un échange de communication entre deux correspondants : Alice et Bernard.

* Alice génère une paire de clés $(k_1^A\ ,\ k_2^A)$ et envoie à Bernard sa clé publique k_1^A (par une voie de communication non sécurisée) ;
* Bernard génère à son tour sa propre paire de clés $(k_1^B\ ,\ k_2^B)$ et envoie à Alice sa clé publique k_1^B ;
* pour envoyer des messages à Alice, Bernard doit d'abord les chiffrer avec la clé publique d'Alice k_1^A et Alice les déchiffrera avec sa clé privée k_2^A ;
* symétriquement, pour envoyer des messages à Bernard, Alice doit d'abord les chiffrer avec la clé publique de Bernard k_1^B et Bernard les déchiffrera avec sa clé privée k_2^B.

• Cette technique a été décrite publiquement pour la première fois en 1976 par W. Diffie et M. Hellman. Une réalisation effective de cette idée a été décrite précisément (avec une définition des fonctions mathématiques de chiffrement et de déchiffrement) en 1978 par les cryptologues R. Rivest, A. Shamir et L. Adleman. Cette méthode particulière porte le nom de **méthode RSA** et elle est encore la méthode de chiffrement asymétrique la plus utilisée aujourd'hui.

• Le grand avantage des méthodes de chiffrement asymétrique est que les correspondants n'ont pas besoin de se communiquer une clé de chiffrement-déchiffrement qu'ils doivent connaître tous les deux et qui doit demeurer inconnue de toute autre personne. Il leur suffit de se communiquer des clés publiques par des voies de communication non sécurisées et de conserver pour eux les clés privées qui permettent de déchiffrer les messages qu'ils reçoivent.

• Un inconvénient important de ces méthodes est qu'elles nécessitent des calculs importants qui peuvent être couteux en temps. Elles ne sont donc pas adaptées pour des échanges d'information très volumineux ou pour lesquels le temps de transmission doit être réduit (par exemple pour la transmission de sons ou de vidéos).

• Une utilisation très courante des méthodes de chiffrement asymétrique est de permettre un premier échange entre correspondants afin de convenir une clé partagée. Cette clé permettra dans un second temps une communication chiffrée par une méthode symétrique.

IV L'attaque de l'homme du milieu et un procédé d'authentification

IV.1 L'attaque de l'homme du milieu

• L'attaque dite de « l'homme du milieu » (MITM pour *man in the middle*) :

• Le chiffrement des communications n'est d'aucune aide pour ce type d'attaque.

• Le problème : Alice se trompe d'interlocuteur. Il est nécessaire **d'authentifier le site** avec lequel on est en communication : c'est-à-dire de disposer d'un moyen de vérifier qu'il est bien ce qu'il prétend être.

• Peu de personnes lisent attentivement les URL (pour *Uniform Ressource Locator* ou « adresse web ») sur lesquelles elles cliquent. Il est possible de créer une imitation parfaite d'un site familier (comme celui d'une banque) et d'apporter des changements subtils au nom de domaine (changer ou ajouter une lettre, etc.)

IV.2 Un procédé d'authentification avec RSA

• Il est nécessaire de pouvoir se tourner vers un **tiers de confiance** qui garantisse qu'un site est géré par des personnes dont il a pu vérifier la bonne foi et donc qu'il est bien ce qu'il prétend être. Exemple de la carte d'identité.

• La procédure de certification suppose la création d'organismes (publiques ou privés) suffisamment reconnus pour assurer cette fonction de délivrer des certificats.

• Chaque site web qui veut pouvoir communiquer selon un protocole sécurisé doit faire une démarche d'un de ces organismes pour obtenir un tel certificat. L'organisme délivre ce certificat après avoir procédé à certaines vérifications. Le responsable du site doit par exemple délivrer une preuve d'achat du nom de domaine utilisé.

• Lors de la communication selon un protocole sécurisé avec un client, il devra délivrer ce certificat pour attester qu'il est authentique.

• Mais comment le client A peut-il vérifier que le certificat numérique fourni par le site B a réellement été délivré par un organisme d'authentification T et qu'il n'a pas été fabriqué par le site B lui-même pour tromper ces interlocuteurs ?

• On va ici faire un nouvel usage de la méthode de chiffrement RSA :
  * il ne s'agit plus de chiffrer un message pour faire en sorte qu'il soit illisible à toute autre personne que le destinataire ;
  * mais de faire en sorte que tout le monde puisse vérifier que seul un certain organisme (le tiers de confiance) a pu produire un certain message (le certificat d'authentification).

• Pour cela, la méthode de chiffrement RSA possède de nouvelles propriétés intéressantes :
  * D'abord, la fonction de chiffrement et la fonction de déchiffrement sont identiques. Autrement dit, le calcul effectué à partir du message en clair m et de la clé k_1 pour produire le message chiffré s est identique au traitement effectué sur le message chiffré s et la clé k_2 pour produire le message en clair m.

Si on désigne par F la fonction utilisée pour chiffrer comme pour déchiffrer les messages, on a alors :

$$\text{Si } F(m, k_1) = s \text{ alors } F(s, k_2) = m$$
$$\text{Ce qui peut aussi s'écrire } F(F(m, k_1), k_2) = m$$

* De plus, les clés k_1 et k_2 y jouent des rôles symétriques, c'est-à-dire que l'on peut utiliser la clé privée pour chiffrer et la clé publique k_1 pour déchiffrer :

$$\text{Si } F(m, k_2) = s \text{ alors } F(s, k_1) = m$$
$$\text{Ce qui peut aussi s'écrire } F(F(m, k_2), k_1) = m$$

• Ce sont ces propriétés de la méthode RSA qui vont permettre à un client A de vérifier que le certificat exhibé par un site B a bien été délivré par un tiers de confiance T. La délivrance et l'utilisation d'un certificat suit les étapes suivantes :

* T produit un couple de clés (k_1^T, k_2^T) pour garantir qu'il est bien l'auteur des certificats qu'il délivre et diffuse sa clé publique k_1^T auprès de toute personne souhaitant utiliser ses certifications ;

* Le site B produit un couple de clés (k_1^B, k_2^B) pour la communication avec ses clients et envoie sa clé publique k_1^B à l'organisme T ;

* l'organisme T produit un certificat c_B^T pour le site B en chiffrant la clé publique k_1^B *avec sa clé privée* k_2^T :

$$c_B^T = F(k_1^B, k_2^T)$$

et envoie ce certificat au site B : on dit que l'organisme T **a signé** la clé k_1^B ;

* à l'initialisation d'une communication avec un client, avec A par exemple, le site B lui envoie sa clé publique k_1^B et le certification c_B^T que lui a fourni l'organisme T ;

* le client A réceptionne la clé k_1^B et le certificat c_B^T et procède à la vérification : il déchiffre le certificat c_B^T avec la *clé publique* diffusé par T k_2^T et obtient la clé publique du site B :

$$F(c_B^T, k_1^T) = k_1^B$$

Il peut alors constater que la clé publique k_1^B que lui a envoyé le site B est identique à celle obtenu en déchiffrant le certificat c_B^T.

• Puisque l'organisme T est le seul à détenir la clé privée k_2^T et puisque le client A a déchiffré le certificat c_B^T avec la clé publique k_1^T fournie par l'organisme T, il peut être certain que c'est bien cet organisme qui avait chiffré (« signé ») le certificat c_B^T et que c'est donc lui qui a délivré ce certificat. Bien entendu, cette certitude repose sur le fait que l'on peut faire confiance à l'organisme T et qu'il ne divulgue pas sa clé privée.

• Il faut bien insister sur le fait que dans le processus de certification, contrairement au chiffrement dans le cadre d'une communication,

* le tiers de confiance *chiffre avec sa clé privée* ;

* le client A *déchiffre avec la clé publique*.

L'objectif visé n'est pas ici le secret de la communication mais de pouvoir vérifier que c'est le tiers T qui est à l'origine du certificat.

V Le protocole HTTPS

• HTTPS signifie « *hypertext transfer protocol secure* ». Il s'agit de produire une couche de sécurisation de la communication en dessous du protocole HTTP. On a visé à *ajouter de nouvelles opérations* au protocole HTTP et non pas réinventer un nouveau protocole.

• Les objectifs de sécurité à remplir comportent deux aspects :

* garantir l'authentification du serveur (le site web) avec lequel le client (le navigateur web) échange des données et donc éviter les attaques de l'homme du milieu.
*assurer le chiffrement et le déchiffrement des données échangées par le client et le serveur afin qu'elles ne soient pas compréhensibles et donc utilisables par une personne qui parviendrait à intercepter la communication.

- On a visé également :
  * la modularité et les compatibilités futures : pouvoir faire évoluer certains paramètres du protocole de façon à pouvoir augmenter la difficulté du problème pour répondre à une plus grande capacité de calcul des attaquants ;
  * tout en restant performant, c'est-à-dire en ne ralentissant pas significativement les communications.

- Le protocole HTTPS suit les étapes suivantes :
  (1) le client A envoie au serveur B un message « *hello* » accompagné d'options (notamment la liste des algorithmes de chiffrement qu'il peut utiliser) ;
  (2) le serveur B répond au client A en indiquant l'algorithme de chiffrement qu'il a choisi dans la liste et un certificat c_B^T, contenant notamment sa clé publique k_1^B, qui a été signé par un organisme tiers T ;
  (3) A valide le certificat c_B^T par la méthode RSA en utilisant la clé publique de l'organisme tiers k_1^T qu'il possède par ailleurs ;
  (4) le client A définit ensuite une clé de session symétrique k_S et envoie cette clé k_S au serveur B. Cette réponse du client est quant à elle chiffrée avec la clé publique k_1^B envoyée par le site B ;
  (5) après cette phase d'authentification et de définition de la clé de session k_S, la communication entre A et B, chiffrée selon l'algorithme *symétrique* choisi et en utilisant la clé partagée k_S, peut commencer. Cette clé ne sera utilisée que pendant cet échange. On dit que c'est une **clé de session**.