TD3 de cryptographie
Correction

Questions de cours :
quels sont les avantages et inconvénients du chiffrement asymétrique par rapport au chiffrement symétrique?
Avantages du chiffrement asymétrique : permet de chiffrer un message pour une entité sans avoir à partager de clé secrète avec cette entité. Personne d'autre que l'entité ne peut déchiffrer le message (en crypto sym, deux entités partagent la même clé). La signature permet donc d'assurer la non répudiation.
Inconvénients du chiffrement asymétrique : très lent, ne permet pas de chiffrer des données mais seulement des clés.

Vous utilisez un chiffrement symétrique avec des clés de 128 bits, quelle longueur de clé devez-vous choisir pour le chiffrement asymétrique (non elliptique)? 4096 RSA
Quelle fonction de hachage correspond à cette même sécurité? 256 bits pour la fonction de hachage
Pourquoi le chiffrement doit-il être probabiliste? Donnez un exemple.
Afin d'éviter que deux clairs donnent le même chiffré. Exemple donnée en TD : Si chaque bloc représente un pixel d'une image, et si le chiffrement n'est pas probabiliste, deux pixels identiques donnent le même chiffré. Ainsi, le chiffré de l'image donne la même image avec des couleurs différentes.
A quoi sert un certificat? À être sûr de la clé publique
Quel est le principal intérêt de la cryptographie elliptique? La taille des clés est plus petite à sécurité égale.

Exercices :

2) Ecrire un protocole qui utilise l'enveloppe digitale pour chiffrer mais qui en plus signe le message

A avec paire de clés (PA/SA) envoie le message m à B avec clés (PB/SB) ; H=fct de hachage
A ->B (E_{PB}(k) || E_{k}(m) || E_{SA}(H(m))
B déchiffre k avec sa clé privée, déchiffre m avec la clé de session k et vérifie la signature avec la clé publique de A

3) Adaptative chosen-ciphertext attack on RSA encryption.
   1. Montrez que le chiffrement du produit de deux clairs est le produit des chiffrés.

   m1^{e}.m2^{e} = (m1.m2)^{e}
   2. Soit m et c resp le clair et le chiffré. Soit A un adversaire. On suppose que A a accès à un oracle de déchiffrement et peut donc déchiffrer tout chiffré différent de c. Comment l'adversaire va-t-il faire pour retrouver le message?

   A choisit x premier avec n, le module RSA , et calcule c . x^e qu'il soumet à l'oracle de déchiffrement. A calcule ensuite x^{-1} qu'il multiplie par l'output de l'oracle.
   On a m = (c . x^e)^d . x^{-1} car e.d=1 mod phi(n).

4) RSA : Small encryption exponent. Supposons que la clé de chiffrement soit e=3 pour tout le monde mais chacun utilise un module différent. Une personne du groupe souhaite chiffrer un même message m pour l'envoyer aux autres (on suppose que les

autres sont trois). Iwan, l'adversaire qui snif le réseau, obtient les trois chiffrés. Comment va-t-il retrouver m?

C1 = m^3 mod n1
C2 = m^3 mod n2
C3 = m^3 mod n3
le théorème du reste chinois permet de résoudre le système
Voyez-vous une solution pour éviter cette attaque tout en gardant la même clé?
Chaque personne pourrait concaténer du sel au message avant le chiffrement.

5) Comparé à RSA , le chiffrement de ElGamal a l'avantage d'être probabiliste. Par contre, quel est son inconvénient si on est limité en mémoire?

Le chiffré comprend deux parties.

3) Alice veut construire un protocole, à base de RSA, qui permette à Bob de signer un message sans que Bob ne puisse connaître le message ni la signature associée à m. L'idée est la suivante : Alice envoie le message camouflé à Bob. Bob le signe et le renvoie à Alice. A partir de cette signature, Alice peut calculer la signature de Bob sur le message m choisi par Alice dès le début. Les paramètres de RSA sont n=pq, la clef publique est (n, e) et la clef privée est d. Alice choisi un entier k premier avec n et considère une fonction qui permet de « camoufler » le message m : $f(m) = m.k^e \bmod n$ et une fonction de « décamouflage » : $g(m) = k^{-1}.m \bmod n$. Essayez d'aider Alice à la mise au point du protocole.

Alice choisit x premier avec n et calcule y=x^{-1} mod n. Elle envoie (x^e . m) à Bob pour signature. Bob calcule la signature SA = (x^e . m)^d = x . m^d avec l'aide de sa clé privée d. Alice récupère SA et calcule S = y . SA = m^d, qui est la signature de m par Bob.

4) RSA est-il ind-cpa? Comment doit-on utiliser RSA en pratique pour obtenir une sécurité satisfaisante?

RSA est déterministe donc il n'est pas ind-cpa.

Rappel sur la sécurité d'un cryptosystème

En cryptographie moderne, un système de chiffrement est traditionnellement étudié dans le modèle dit en boîte noire. Dans ce modèle, le cryptosystème est vu comme un oracle répondant à des requêtes de chiffrement (et/ou déchiffrement) de messages à partir d'une valeur secrète : la clé. La sécurité du cryptosystème est alors définie suivant un simple jeu. Un adversaire interroge l'oracle sur le chiffrement (et/ou le déchiffrement) de messages de son choix et, selon les réponses obtenues, tente de déterminer la valeur de la clé secrète (ou encore de chiffrer/déchiffrer un message pour lequel il n'a pas questionné l'oracle). Si, tout en suivant une stratégie optimale, l'adversaire n'a qu'une chance de gain négligeable, la sécurité est alors établie. Plusieurs cryptosystèmes existants ont été prouvés sûrs dans le modèle en boîte noire. Cependant, ce modèle n'est pas toujours suffisant pour établir la sécurité d'un cryptosystème en pratique. Prenons l'exemple de la carte à puce qui est utilisée comme support pour le cryptosystème dans de nombreuses applications telles le bancaire, le contrôle d'accès, la téléphonie mobile, la télévision à péage ou encore le passeport électronique. De par la nature de ces applications, un cryptosystème implanté sur carte à puce est physiquement accessible à de potentiels attaquants. Cet accès physique invalide l'abstraction du cryptosystème par un oracle de chiffrement car il permet à

l'adversaire d'en observer et/ou d'en perturber le comportement physique. De nouvelles attaques cryptanalytiques deviennent alors possibles se regroupant sous le terme de cryptanalyse physique.

La cryptanalyse physique se compose essentiellement de deux familles principales d'attaques : les attaques par canaux auxiliaires (side channel) et les attaques par fautes. L'objet des attaques par canaux auxiliaires est l'analyse des différentes fuites physiques d'une implémentation cryptographique durant ses calculs. On compte parmi ces fuites le temps d'exécution, la consommation électrique ainsi que les émanations d'ondes électromagnétiques. L'observation de ces dits canaux auxiliaires fournit de l'information sensible sur le calcul cryptographique. La valeur de la clé peut alors facilement être déterminée par traitement statistique bien que le cryptosystème soit sûr dans le modèle en boîte noire. L'accès à une implémentation cryptographique permet plus qu'une simple observation passive de son comportement physique ; il devient également possible d'en perturber le calcul. Partant de ce principe, les attaques par fautes consistent en la corruption de calculs cryptographiques en vue de l'obtention de résultats erronés. De manière tout à fait surprenante, ces derniers peuvent alors être traités afin d'en extraire de l'information sur la clé secrète.

Revenons à la sécurité des cryptosystèmes de chiffrement dans le modèle en boîte noire. L'indistinguabilité est une propriété importante des systèmes. Intuitivement, un cryptosystème possède cette propriété si un adversaire ne peut pas distinguer deux chiffrés correspondants à deux textes clairs connus. Plus précisément, il existe plusieurs niveaux d'indistinguabilité. Considérons le jeu suivant avec comme acteurs un challenger et un adversaire :

- l'adversaire a à sa disposition un oracle de chiffrement, lui permettant de chiffrer tout message. Il soumet deux messages m1 et m2 distincts au challenger.
- Le challenger choisit au hasard un des messages et envoie à l'adversaire le chiffré de ce message.
- L'adversaire peut faire autant de calculs qu'il veut et peut utiliser l'oracle à volonté. Finalement, il lui est demandé de dire quel message a été chiffré (m1 ou m2).

Un système est ind-cpa (chosen ciphertext attack) si la probabilité que l'adversaire trouve le bon message est ½ (autant de chance de se tromper que de trouver le bon message). Bien sûr, si le chiffrement est déterministe, le système ne peut pas être ind-cpa puisque l'adversaire peut chiffrer les deux messages grâce à l'oracle de chiffrement et comparer avec le chiffré que lui donne le challenger.
Ind-cca (adaptative and non adaptative chosen ciphertext attack) utilise le même jeu mais l'adversaire a plus de pouvoir.

- Ind-cca : L'adversaire a maintenant accès à un oracle de déchiffrement (en plus de l'oracle de chiffrement). Il peut donc chiffrer n'importe quel clair jusqu'à ce que le challenger lui envoie le chiffré. Lorsqu'il reçoit le chiffré, il n'a plus le droit d'utiliser l'oracle de déchiffrement.
- Ind-cca2 : L'adversaire a maintenant accès à un oracle de déchiffrement (en plus de l'oracle de chiffrement) et il peut l'utiliser même après avoir reçu le chiffré du challenger. Seule contrainte, il ne peut pas soumettre à l'oracle le chiffré reçu du challenger, mais il peut soumettre n'importe quel autre chiffré.

Un système ind-cca2 est plus robuste qu'un système ind-cca qui est lui même plus robuste qu'un système ind-cpa. En effet, plus l'adversaire a du pouvoir, plus le système doit être robuste pour lui résister.