durée 2h00, aucun document autorisé

Exercice 1 chiffrement à clefs publiques

On appelle E un procédé de chiffrement à clef publique et D le procédé de déchiffrement associé. On suppose qu'il existe un procédé de signature associé à E que l 'on notera S. On notera VS le procédé de vérification de signature associé.

On suppose que toutes les personnes intervenant dans cet exercice ont chacune un couple (clef privée, clef publique) correspondant aux procédés cités ci-dessus. Par souci de simplification, on supposera que le même couple peut servir indifféremment aux opérations de chiffrement ou de signature.

Répondez aux questions suivantes directement sur l'énoncé :

Question 1: Alice veut envoyer un message chiffré à Bob, avec quelle clef doit-elle le chiffrer ?

A l'arrivée, quelle clef, Bob doit-il utiliser pour déchiffrer le message ?

Question 2: Alice veut envoyer un message signé à Bob, avec quelle clef doit-elle le signer ?

A l'arrivée, quelle clef, Bob doit-il utiliser pour vérifier la signature du message ?

Question 3: Alice veut envoyer un message chiffré et signé à Bob, avec quelle clef doit-elle le chiffrer ?

Le signer ?

A l'arrivée, quelle clef, Bob doit-il utiliser pour déchiffrer le message ?

Pour vérifier la signature ?

Question 4: Alice veut envoyer un message chiffré et signé à Bob, Gérard, Jackie, Ahmed, ... (25 destinataires) avec quelle clef doit-elle le chiffrer ? Le signer ?

Exercice 2 Diffie-Hellman

question 1

Quel est le but de l'algorithme de Diffie-Hellman ?

question 2

Les données suivantes sont publiques :

- p un grand nombre premier,
- G un groupe multiplicatif de cardinal p-1
- g un générateur de G

Décrivez l'algorithme de Diffie Hellman (description détaillée de chacune des étapes).

question 3

Alice et Bob veulent communiquer de façon sûre à travers un réseau non sûr. Ils décident d'utiliser Diffie-Hellman. L'espion possède un contrôle total du réseau : il peut lire et modifier tout ce qui est y passe. Expliquez quelles sont les faiblesses de Diffie-Hellman dans ce contexte.

Exercice 3 hachage cryptographique

question 1

Quelles sont les propriétés que doivent vérifier les fonctions de hachage cryptographiques pour pouvoir être utilisées dans le cadre d'applications cryptographiques ?

Donnez deux exemples d'applications utilisant les fonctions de hachage à sens unique dans lesquelles il est important que ces propriétés soient vérifiées. Pour chaque application, vous expliquerez en quoi ces propriétés interviennent

question 2

Exercice 4 hachage cryptologique

On considère l'application suivante du hachage cryptographique. On suppose que h est une fonction de hachage cryptographique que tout le monde (Alice, le serveur, l'espion, …) connaissent.

- Alice choisit un nombre g
- elle calcule h(g), h(h(g)), …, h(h(h(h(h(h(h(h(h(g))))))))))=h^{11}(g)
- elle transmet[1] h^{11}(g) au serveur qui le stocke tel quel.

L'espion, de son côté, peut espionner tout ce qui passe sur le réseau. Il souhaite se connecter à distance au serveur en se faisant passer pour Alice.

question 1

Pour s'authentifier à distance, Alice transmet h^{10}(g) au serveur.

- Est-ce un authentification solide ?

Pour s'authentifier à distance une deuxième fois, Alice transmet h^{10}(g) au serveur.

- Est-ce un authentification solide ?

question 2

Proposez un algorithme d'authentification sûr qui s'appuie sur les résultats de la question 1 et qui résiste à un espion qui peut lire[2] tout ce qui passe sur le réseau. La seule fonction cryptographique que votre algorithme est autorisé à utiliser est le fonction h.

Exercice 5 hachage cryptologique

Dans cet exercice nous étudions des fonctions de hachage à sens unique. Pour simplifier la présentation, ainsi que les essais qui sont demandés, nous allons les décrire comme retournant une empreinte sur 8 bits. Il est clair que cela est insuffisant pour des questions de sécurité mais toutes les descriptions qui suivent peuvent être étendues pour générer des empreintes de longueur quelconques.

Si le message M à hacher n'a pas une longueur qui est un multiple de 8 bits, la fonction de hachage commence par le compléter avec un bit 1 suivi d'autant de bits 0 que nécessaire pour que le message ainsi complété contienne un multiple de 8 bits. Notons M' le message ainsi complété après cette étape. M' est alors découpé en blocs de 8 bits consécutifs. Notons B_1,B_2,...,B_q ces q blocs, q⩽1. On peut alors décrire les deux fonctions de hachage étudiées dans cet exercice.

$$\mathbf{h\_1(M) = B\_1 \oplus B\_2 \ ... \ B\_l}$$

avec ⊕ désignant le XOR bit à bit (Rappel~: 0 ⊕ 0= 1 ⊕ 1 =0 et 0 ⊕ 1 = 1 ⊕ 0 = 1).

Pour décrire h_2, notons A = a_1a_2... a_8 = B_1 ⊕ B_2 ⊕ ... ⊕ B_l (où a_i ∈ {0,1 }). Avec ces notations, on a~: h_2(M) = c_1... c_8 avec c_i = a_i ⊕ a_{i+1}, pour i=1,..., 7 et c_8 = a_8 ⊕ 0.

1 Supposons qu'elle tape directement cette valeur sur le clavier du serveur

2 Mais pas modifier

- Répondez aux questions suivantes en justifiant clairement vos calculs. :
  - Soit M=001011100001, calculez h_1(M) et h_2(M).
  - Construisez un message M tel que h_1(M) = 01011001.
  - Construisez un message M tel que h_2(M)=11001011
  - Comment l'espion peut il attaquer h_1 et h_2 ? Que peut il faire ?

Dans votre réponse à la question précédente vous avez peut être utilisé le fait que les empreintes calculées ne sont que sur 8 bits. Qu'en est il de la sécurité de h_1 et h_2 si on les généralise naturellement pour produire des empreintes de 160 bits?

Exercice 6 PKI

A quel problème répond une infrastructure de gestion de clefs (PKI) ?

Décrivez sa structure.