Signature numérique

1. INTRODUCTION

Une signature numérique d'un message est un nombre dépendant d'un secret connu uniquement du signataire et du contenu du message signé. Les signatures doivent être vérifiables; si un différend survient quant à savoir si une partie a signé un document (causée par un signataire qui tente de répudier la signature qu'il a créé, ou un prestataire frauduleux), un tiers impartial devrait être en mesure de résoudre la question de façon équitable, sans nécessiter l'accès à des informations secrètes du signataire (clé privée). Les signatures numériques ont de nombreuses applications dans la sécurité des informations, y compris l'authentification, l'intégrité des données, et la non-répudiation.

Une des applications les plus importantes de la signature numérique est la certification des clés publiques dans les grands réseaux. La certification est un moyen pour un tiers de confiance de lier l'identité d'un utilisateur à une clé publique. Par la suite, d'autres entités peuvent authentifier une clé publique, sans l'assistance d'un tiers de confiance.

Le concept et l'utilité d'une signature numérique a été reconnu depuis plusieurs années avant que toute réalisation pratique soit disponible. La première méthode découverte fut le schéma de signature RSA, qui reste aujourd'hui l'une des techniques les plus pratiques et polyvalentes disponibles. D'autres schémas de signature ont vu le jour tel signature El Gamal, DSA, ….

2. Définitions

1. Une signature numérique est une donnée associée à un message (sous forme numérique).
2. Un algorithme de génération de signature numérique (ou algorithme de génération de signature) est un procédé pour produire une signature numérique.
3. Un algorithme de vérification de signature numérique (ou un algorithme de vérification) est une méthode pour vérifier que la signature numérique est authentique (elle a bien été créé par l'entité spécifiée).
4. Un schéma de signature numérique (ou le mécanisme) est constitué d'un algorithme de génération de signature et un algorithme de vérification associées.
5. Un processus de signature numérique de signature (ou de la procédure) se compose d'un algorithme de génération de signature numérique (mathématique), ainsi qu'un procédé de mise en forme des données dans les messages qui peuvent être signées.
6. Un processus de vérification de signature numérique (ou de la procédure) se compose d'un algorithme de vérification, avec une méthode de récupération de données à partir du message

Notations

M : un ensemble d'éléments appelé l'espace des messages.
M_S : un ensemble d'éléments appelé l'espace à signer.
S : un ensemble d'éléments appelé l'espace des signatures.
R : relation de $M \rightarrow M_S$ appelé la fonction de redondance.
M_R : l'image de R (à savoir, $M_R = Im(R)$).
R^{-1} : l'inverse de R (à savoir, $R^{-1}: M_R \rightarrow M$).
R : un ensemble d'éléments appelé l'indexation fixée pour la signature.

h : une fonction à sens unique avec domaine M.
M_h : l'image de h (ie, h: $M \rightarrow M_h$); $M_h \subseteq M_S$ appelé espace de valeurs de hachage.

3. CLASSIFICATION DES SCHEMAS DE SIGNATURES NUMERIQUES

Il y a deux classes de schémas de signature numérique:

a. Schémas de signature numérique avec appendice qui exige le message original comme entrée pour la vérification de signature.
b. Schémas de signature numérique avec récupération des messages ne nécessitent pas le message d'origine comme entrée pour l'algorithme de vérification. Dans ce cas, le message d'origine est récupéré de la signature elle-même.

3.1 Algorithme de génération de clés pour les schémas de signature numérique avec appendice

Chaque entité crée une clé privée de signature pour les messages, et une clé publique correspondante pour être utilisée par d'autres entités pour la vérification des signatures.

a. Chaque entité A doit choisir une clé privée qui définit un ensemble $S_A = \{S_{A,k}: k \in R\}$ des transformations. Chaque $S_{A,k}$ est une application M_h à S et appelée transformation de signature.
b. S_A définit une correspondance V_A de M_h x S vers {true, false} de telle sorte que $V_A(\tilde{m};\bar{s})$= vrai, si $S_{A,k}(\tilde{m})=s$,
= faux, sinon;
Pour tous les $\tilde{m} \in M_h$, $\bar{s} \in S$; ici, $\tilde{m} = h(m)$ pour $m \in M$. V_A est appelée une vérification de la transformation et qui est construit de telle sorte qu'elle peut être calculée à l'insu de la clé privée du signataire.
c. La clé publique est A est V_A; la clé privée de A est l'ensemble S_A

3.2 Algorithme de génération de signature et de vérification (Schémas de signature numérique avec appendice)

Une entité produit une signature $s \in S$; relative à un message $m \in M$; qui peut ensuite être vérifiée par une entité B.

1. Génération de signature. L'entité A doit faire ce qui suit:
   a. Sélectionnez un élément $k \in R$.
   b. Calculer $\tilde{m} = h(m)$ et $s^* = S_{A;k}(\tilde{m})$.
   c. La signature de A pour m est s^*. Les deux valeurs m et s^* sont mises à la disposition des entités qui souhaiteront vérifier la signature.
2. Vérification. L'entité B doit effectuer les opérations suivantes:
   a. Un authentique Obtenir la clé publique de V_A.
   b. Calculer $\tilde{m} = h(m)$ et $u = VA(\tilde{m}; s^*)$.
   c. Accepter la signature si et seulement si u = true.

3.3 Schémas de signature numérique avec récupération de message

Les schémas de signature numérique décrits dans cette section ont la particularité que le message signé peut être récupéré à partir de la signature elle-même. Dans la pratique, cette fonctionnalité est d'utiliser pour les messages courts.

3.3.1 Définition

Un schéma de signature numérique avec récupération de message est un schéma de signature numérique pour laquelle une connaissance à priori du message n'est pas nécessaire pour l'algorithme de vérification.
Exemples de mécanismes fournissant des signatures numériques avec récupération de message sont RSA, Rabin, des schémas de signature à clé publique.

3.3.2 Algorithme génération de clés pour les schémas de signature numérique avec récupération message

Chaque entité crée une clé privée à utiliser pour les messages de signature, et une clé publique correspondante à utiliser par d'autres entités pour la vérification des signatures.

1. Chaque entité A doit choisir un ensemble $S_A = \{S_{A;\,k} : k \in R\}$ des transformations. Chaque $S_{A;k}$ est une application unaire de M_S vers S et est appelée une transformation de signature.
2. S_A définit une correspondance V_A avec la propriété que $V_A \circ S_{A;\,K}$ est l'identité sur M_S pour tout $k \in R$. V_A est appelée une transformation de vérification et est construit de telle sorte qu'elle peut être calculée sans la connaissance de la clé privée du signataire.
3. La clé publique de A est V_A; la clé privée de A est l'ensemble S_A.

3.3.3 Algorithme de génération et de vérification de signature pour les schémas de récupération des messages

Une entité A produit une signature $s \in S$ pour un message $m \in M$, elle peut être vérifiée par une entité B ultérieurement. Le message m est récupéré à partir de s

1. Génération de Signature. L'entité A doit faire ce qui suit:
   a. Sélectionnez un élément $k \in R$.
   b. Calculer $\tilde{m} = R(m)$ et $s^* = S_{A;\,k}(\tilde{m})$. (R est une fonction de redondance)
   c. La signature de A pour m est s^* ; m est s^* sont mis à la disposition des entités qui désirent vérifier la signature et récupérer m.
2. Vérification. L'entité B doit effectuer les opérations suivantes:
   a. Obtenir la clé publique authentique de A : V_A.
   b. Calculer $\tilde{m}$ $V_A(s^*)$.
   c. Vérifiez que $\tilde{m} \in M_R$. (Si $m \notin M_R$, alors rejeter la signature.)
   d. Récupérer m à partir de $\tilde{m}$ en calculant $R^{-1}(\tilde{m})$.

4. Le schéma de signature RSA

L'espace réservé au message chiffré et un espace pour le système de chiffrement RSA à clé publique sont à la fois $Z_n = \{0;\ 1,\ 2,\ldots;\ n-1\}$ où $n = pq$ est le produit de deux des nombres premiers distincts (choisis au hasard).
A partir du fait que la transformation de chiffrement est une bijection, les signatures numériques peuvent être créées en inversant les rôles de cryptage et de décryptage. Le schéma de signature RSA est un schéma de signature numérique déterministe qui permet la récupération des messages. L'espace de signature M_S et l'espace des signatures S sont les deux Z_n. Une fonction de redondance $R: M \rightarrow Zn$ est choisie et elle est publique.
4.1 Algorithme de génération de clés pour le schéma de signature RSA
Chaque entité crée une clé publique RSA et une clé privée correspondante.

Chaque entité A doit faire ce qui suit:

1. Générer deux grands nombres premiers distincts p et q au hasard, ayant les deux la même taille.
2. Calculer n = pq et $\varphi(n) = (p - 1)(q - 1)$.
3. Sélectionnez un e entier aléatoire, $1 < e < \varphi(n)$, tels que pgcd $(e; \varphi(n)) = 1$.
4. Utiliser l'algorithme d'Euclide étendu pour calculer l'unique entier d, $1 < d < \varphi(n)$, tels que $ed \equiv 1 \pmod{\varphi(n)}$.
5. Une clé publique est (n; e); clé privée est d.

4.2 Algorithme de génération et de vérification de signature RSA

Une entité A signe un message $m \in M$. Toute entité B peut vérifier la signature A et récupérer le message m à partir de la signature.

1. génération de signature. L'entité A doit faire ce qui suit:
    a. Calculer $\tilde{m} = R(m)$, un nombre entier dans l'intervalle [0; n - 1].
    b. Calculer $s = \tilde{m}^d \bmod n$.
    c. La signature de m pour A est s.
2. Vérification. Pour vérifier la signature de A et de récupérer le message m, B doit:
    a. obtenir la clé publique authentique de A : (n, e).
    b. Calculer $\tilde{m} = s^d \bmod n$.
    c. Vérifiez que $\tilde{m} \in M_R$, sinon, rejeter la signature.
    d. Récupérer $m = R^{-1}(\tilde{m})$.

Preuve que la vérification de signature fonctionne. Si s est une signature pour un message m, alors $s \equiv \tilde{m}^d \bmod n$ où $\tilde{m} = R(m)$. Du fait que $ed \equiv 1 \pmod{\varphi(n)}$, $s^e \equiv \tilde{m}^{ed} \equiv \tilde{m} \pmod n$. Enfin, $R^{-1}(\tilde{m}) = R^{-1}(R(m)) = m$

5. Le schéma de signature ElGamal

Le schéma de signature ElGamal est un mécanisme de signature aléatoire. Il génère des signatures numériques avec appendice sur les messages binaires de longueur arbitraire, et requiert une fonction de hachage $h: \{0; 1\}^* \rightarrow Zp$ où p est un grand nombre premier.

5.1 Algorithme de génération de clés pour le schéma de signature ElGamal

Chaque entité crée une clé publique et clé privée correspondante.
Chaque entité A doit faire ce qui suit:

1. Générer un grand nombre aléatoire p premier et un générateur g du groupe multiplicatif Zp*.
2. Sélectionner une entier aléatoire a, $1 \leq a \leq p - 2$.
3. Calculer $y = g^a \bmod p$
4. La clé publique de A est (p;g; y); la clé privée est a

5.2 Algorithme de génération et de vérification de signature ElGamal

Une entité A signe un message m binaire de longueur arbitraire. Toute entité B peut vérifier cette signature en utilisant la clé publique de A.

1. Génération de signature. L'entité A doit faire ce qui suit:
    a. Sélectionner un k entier aléatoire secret, $1 \leq k \leq p - 2$, avec pgcd $(k, p - 1) = 1$.
    b. Calculer $r = g^k \bmod p$.

c. Calculer k^{-1} mod (p - 1).
d. Calculer $s = k^{-1}\{h(m) - ar\} \bmod (p - 1)$.
e. La signature de A pour m est le couple (r, s).

2. Vérification. Pour vérifier la signature (r, s) de A sur m, B doit effectuer les opérations suivantes:

a. Obtenir la clé publique authentique de A (p;g; y).
b. Vérifier que $1 \leq r \leq p - 1$; sinon, rejeter la signature.
c. Calculer $v_1 = y^r r^s \bmod p$.
d. Calculer h(m) et $v_2 = g^{h(m)} \bmod p$.
e. accepte la signature si et seulement si $v_1 = v_2$.

Preuve que la vérification de signature fonctionne. Si la signature a été générée par A, alors $s \equiv k^{-1}\{h(m)-ar\} \pmod{p-1}$. En multipliant les deux côtés par k on aura $ks \equiv h(m)-ar \pmod{p-1}$, et la réorganisation donne $h(m) \equiv ar + ks \pmod{p - 1}$. Cela implique $g^{h(m)} \equiv g^{ar + ks} \equiv (g^a)^r r^s \pmod p$. Ainsi, $v_1 = v_2$.

6. L'algorithme de signature numérique (DSA)

L'algorithme est une variante du schéma ElGamal, c'est une signature numérique avec appendice. Il est appelé aussi DSS.

Le mécanisme de signature nécessite une fonction de hachage h: $\{0; 1\} \rightarrow Zq$ pour un certain entier q. Le DSS exige explicitement l'utilisation de l'algorithme de hachage sécurisé (SHA-1),

6.1 Algorithme de génération de clés pour DSA

Chaque entité crée une clé publique et clé privée correspondante.

Chaque entité A doit faire ce qui suit:

1. Sélectionnez un nombre q premier tel que $2^{159} < q < 2^{160}$.

2. Choisissez t tel que $0 \leq t \leq 8$, et sélectionnez un nombre premier p où $2^{511+ 64t} < p < 2^{512+ 64 t}$, avec la propriété que q divise (p - 1).

3. (Choisissez un générateur α du groupe cyclique unique d'ordre q dans Zp*.)

3.1 Choisir un élément $g \in Zp^*$ et calculer $\alpha = g^{(p-1)/q} \bmod p$.

3.2 Si $\alpha = 1$ alors passez à l'étape 3.1.

4. Sélectionnez un nombre entier aléatoire a tel que $1 \leq a \leq q - 1$.

5. Calculer $y = \alpha^a \bmod p$.

6. Une clé publique est $(p, q; \alpha; y)$; clé privée est a.

6.2 Algorithme de génération et de vérification de signature DSA

Une entité A signe un message binaire m de longueur arbitraire. Toute entité B peut vérifier cette signature en utilisant la clé publique de A.

1. Génération de signature. L'entité A doit faire ce qui suit:

a. Sélectionner un entier aléatoire k secret; $0 < k < q$.
b. Calculer $r = (\alpha^k \bmod p) \bmod q$
c. Calculer $k^{-1} \bmod q$
d. Calculer $s = k^{-1}\{h(m) + ar\} \bmod q$.
e. La signature de m pour A est le couple (r, s)

2. Vérification. Pour vérifier la signature de A (r, s) sur m, B doit effectuer les opérations suivantes:

a. Obtenir la clé publique authentique (p, q;α; y).
b. Vérifier que 0 <r <q et 0 <s <q; sinon, rejeter la signature.
c. Calculer w = s^{-1} mod q et h (m).
d. Calculer u_1 = w. h (m) mod q et u_2 = rw mod q.
e. Calculer v = ($\alpha^{u1}y^{u2}$ mod p) mod q.
f. Accepter la signature si et seulement si v = r.

Preuve que la vérification de signature fonctionne. Si (r, s) est une signature légitime de l'entité A sur le message m, alors h(m)$\equiv$ -ar + ks (mod q). En multipliant les deux côtés de cette congruence par w et en réarrangeant on a : w. h(m) + arw$\equiv$ k (mod q). Mais ceci est tout simplement u_1 + a$u_2\equiv$ k (mod q). L'élévation à la puissance des deux côtés de cette équation donne (α^{u1yu2} mod p) mod q = (α^{k} mod p) mod q. Par conséquent, r = v,