Le web de Dominique Guebey – Bazar informatique
Page : http://www.dg77.net/tekno/securite/pubkey.htm

A propos de sécurité informatique

Chiffrage à clefs asymétriques privée/publique, RSA

Définition :

Ce système repose sur l'établissement d'une paire de clefs de chiffrage, chacune permettant de déchiffrer ce que l'autre à chiffré. Si l'une des deux, qu'on appellera *clé privée*, est détenue par une seule personne, l'autre étant disponible pour tous les autres (ce sera la *clé publique*), celà permettra à chacun d'envoyer au propriétaire de la clé privée un message crypté que ce dernier sera seul à pouvoir traduire. A l'inverse, il sera le seul à pouvoir générer une *signature* que les autres pourront vérifier à l'aide de la clef publique.

Le principal inconvénient des clefs asymétriques est qu'elles exigent beaucoup plus de temps de traitement qu'une clé symétrique.

Rapide introduction au chiffrage asymétrique :

La méthode utilise les propriétés de factorisation des nombres premiers[1]. Elle est basée sur les principes établis par Ronald Rivest, Adi Shamir et Leonard Adleman, d'où l'acronyme RSA. Dans ce système, on part du produit de deux nombres premiers (p et q). L'une des clefs est un nombre tel que lui et (p-1).(q-1) n'ont que 1 pour Plus Grand Commun Diviseur[2]. L'autre est un nombre tel que son produit avec la première clef est égal à 1(mod((p-1).(q-1)))[3]

RSA applique le théorème d'Euler. Etant donnés p et q deux nombres premiers différents, et a un nombre divisible par aucun des deux, alors a élevé à la puissance (p-1).(q-1) équivaut à 1(mod $p.q$)

Exemple : soit $a = 5$, $p = 2$, $q = 3$

Alors $(p-1).(q-1) = 1*2 = 2$; $p.q = 2*3 = 6$; et $5^2 = 25 = 1(\text{mod}6)$ (parce que 25 = 6*4 + 1).

Formules générales :

Chiffrage : *Ma = N(mod n)*. M est le message de départ, N est le message envoyé.

Déchiffrage : *Nb = W(mod n)*. On doit constater que W est égal à M.

n est une valeur commune aux deux clefs, *a* et *b* peuvent être permutés entre ces deux opérations.

Il est recommandé d'utiliser des clefs asymétriques d'au moins 1024 bits, ce qui est très supérieur aux algorythmes symétriques. A titre d'exemple, un Pentium3 1.2GHz mettra moins de 8 secondes pour trouver que le nombre de 133 bits : 10844374209563071155801253734487122581039 est le produit de 158757429537214379 et 68307821820842960838541.

RSA, illustration très simplifiée :

Tout d'abord calcul de la paire de clefs :

- Choix des facteurs $p = 3$ et $q = 11$. En fait, ils doivent être le plus grand possible, et pas trop proches l'un de l'autre.
- ***n*** = $p.q$ = 3*11 = **33**
- $(p-1).(q-1) = 2*10 = 20$
- Choix d'un nombre dont le PGCD avec 20 soit égal à un , ce qui veut dire qu'on écarte 2, 4, 5, 10, dans ce qui reste on choisit par ex. 7. ***a* = 7**. Ce sera la clé privée.
- Calcul de la clé publique **b** : comme ab = 1(mod ((p-1)(q-1))) ou dans l'exemple : 7*b = 1(mod 20), on devine que **b = 3** car le reste de (3*7)/20 est 1.

Cryptage d'une valeur (par exemple : 5) avec la clé privée (**a** et **n**) : on élève 5 à la puissance *a* (i.e. 7) ce qui donne 78125. Il faut alors calculer le résultat modulo pq (i.e. 33) qui est 14 car 78125=2367*33+14

Déchiffrement : le réceptionnaire dispose de la deuxième clef composée de deux parties : la valeur commune n=33, et b=3. 14 est d'abord élevé à la puissance b, (ici 14 au cube) ce qui donne 2744. Il ne reste plus qu'à calculer 2744 modulo 33, ce qui donne bien 5 (puisque 83*33+5=2744).

RSA, exemple avec valeurs plus réalistes :

Où l'on comprendra au moins que la méthode pouvait bien rester théorique tant qu'on n'avait pas fabriqué des ordinateurs.

Calcul de la paire de clefs :

- Choix des facteurs $p = 79$ et $q = 127$.
- $\boldsymbol{n} = p.q$ = 79*127 = **10033**
- $(p-1).(q-1)$ = 78*126 = 9828
- Choix d'un nombre dont le PGCD avec 9828 soit égal à un : parmi les valeurs possibles on choisit (par exemple) ***a* = 97** (clé privée).
- Calcul de la clé publique **b** : 1(mod 9828) = 2533*97 ; donc ***b* = 2533** .

Chiffrage d'une valeur avec la clé privée (**a** et **n**) :

- Préparation : dans un message RSA les données à chiffrer, dûment numérisées, sont découpées en blocs de 4 chiffres.
- Chiffrement des blocs : on élève chaque nombre de 4 chiffres à la puissance *a* (97), puis on calcule le résultat modulo 10033 à partir du nombre obtenu. Par exemple **2118** à la puissance 97 donnera 9253(mod 10033).

Déchiffrage avec la clef publique (**b** et **n**) :

- Etant donnée l'autre clef ***b* = 2533**, chaque bloc est élevé à la puissance 2533 (i.e. multiplié par lui-même 2533 fois !).
- Puis on fait le calcul modulo 10033 : 9253 puissance 2533 = 2118(mod 10033) On retrouve les **2118**

RSA, exemples en Javascript

Les amateurs y sont allé de leur script, quelques exemples de démonstrations en ligne :

RSA en Javascript par Dave Shapiro Geek repenti [http://www.ohdave.com/rsa/]
Autre page utilisant le BigInt.js de D. Shapiro [http://www.leemon.com/crypto/BigInt.html]
Cryptography home (designed by Cary Sullivan and Rummy Makmur.) [http://www.cs.pitt.edu/~kirk/cs1501/notes/rsademo/]
RSA, RC4, AES, MD5… par Michiel van Everdingen [http://home.zonnet.nl/MAvanEverdingen/Code/]
Demo RSA en français. [http://cryptosec.lautre.net/articles/rsa.html]

Notes

1. Un **nombre premier** est un entier qui ne peut être divisé que par 1 ou par lui-même. Tous les autres entiers sont le produit de deux nombres premiers.
2. Le **PGCD** est le plus grand de tous les diviseurs entiers communs à deux nombres entiers. Par exemple, le PGCD de 36 et 48 est 12.
3. **mod="modulo"** a=1(mod b) signifie que le reste de la division de a par b est 1 ; autrement dit que un ajouté au produit de b par le résultat entier de a/b, donne a.

Cre : 28 dec 2003 - Maj : 19 aou 2011
À propos de ces pages / *about these pages* : http://www.dg77.net/about.htm