Cryptographie asymétrique: RSA, Diffie-Hellman

Cours de sécurité: DUT S4

2013-2014

1 RSA

Rappel RSA au tableau

1.1 Codage et décodage RSA

On considère la clef publique RSA (11, 319), c'est-à-dire pour $n = 319$ et $e = 11$. **On donne** $p = 11$ **et** $q = 29$.

1. Calculez le chiffré du message $m = 100$
2. Le message $m = 635$ peut-il résulter d'un codage avec la clé publique ? Pourquoi ?
3. Calculez la fonction indicatrice d'Euler $\varphi(n)$
4. Calculez la clé privée d, telle que $ed = 1 \ mod \ \varphi(n)$. Un peu d'aide :
   (a) Les coefficients de bezout de deux entiers a et b sont les entiers x et y tels que:

   $$x * a + b * y = pgcd(a, b)$$

   On suppose que l'on possède une fonction $bezout(a, b)$ qui renvoit pour tout couple (a, b) les coefficients de bezout (x, y) associés. Sachant que e et n sont premiers entre eux (i.e $pgcd(e, \varphi(n)) = 1$), montrer comment obtenir d à partir de e et $\varphi(n)$
   (b) Le calcul des coefficient de bezout peut se faire à l'aide de l'algorithme d'Euclide étendu.
   - Description de l'algorithme: http://fr.wikipedia.org/wiki/Algorithme_d%27Euclide_%C3%A9tendu
   - Une version de l'algorithme facile à faire tourner à la main : http://marauder.millersville.edu/~bikenaga/absalg/exteuc/exteucex.html
5. Déchiffrez le message $m' = 133$

1.2 Attaques sur l'implémentation

1.2.1 Algorithme de Naggle

Alice et Bob communiquent en utilisant RSA. Leur clé publique est incassable (1024 bits). La liaison entre la machine d'Alice et celle de Bob est très rapide, si bien que les caractères saisis au clavier par Alice sont envoyés un par un à Bob.

1. Comment Oscar peut-il déchiffrer les messages d'Alice ?
2. Comment résoudre ce problème ?
3. L'amélioration du protocole que vous avez proposée permet-elle d'empêcher Oscar de distinguer deux messages identiques d'Alice ? Permet-elle d'empêcher Oscar de ré-envoyer plus tard un message qu'il aura intercepté (sans pouvoir le comprendre) ?

1.2.2 Nombres p et q proches

Un programmeur Toto décide de dévier du protocole RSA en choisissant non pas deux grands nombres premiers p et q **aléatoires** mais deux grands nombres premiers p et q **très proches**, avec $p > q$. Ce nouveau protocole sera appelé `RSAnaze`.

1. En posant $n = p * q$, $s = \frac{p-q}{2}$ et $t = \frac{p+q}{2}$, démontrez que $n = t^2 - s^2$
   (a) Que peut ont dire de s (et donc de s^2) ?
   (b) En déduire une approximation de t
2. Déduire de la question précédente une attaque de `RSAnaze`. On précisera:
   (a) Le problème résolu
   (b) L'algorithme le resolvant

1.3 Attaque par modulo commun

Alice envoie le même message clair m (chiffré via RSA) à Bob et Bart. Tous deux utilisent le même modulo n et des exposants publics e_1 et e_2 qui sont premiers entre-eux. Alice publie donc vers Bob et Bart respectivement $m_1 = m^{e_1} \ mod \ n$ et $m_2 = m^{e_2} \ mod \ n$.

1. Montrer que si u_1 et u_2 sont tels que $u_1 * e_1 + u_2 * e_2 = 1$ alors $m = (m_1^{u_1} \ mod \ n) * m_2^{u_2} \ mod \ n$.
2. Comment trouver u_1 et u_2 ?
3. Implémentez l'attaque pour $n = 473$, $e_1 = 17$, $e_2 = 5$ $m_1 = 381$ et $m_2 = 252$. Aide:
   - On a forcément u_1 ou u_2 négatif, cela risque de poser problème. Pourquoi ?
   - On pose u_1 négatif. Montrez que $m_1^{u_1} = (m_1^{-1})^{|u_1|}$
   - A l'aide des résultats précédents, montrez que $m = (m_1^{-1} \ mod \ n)^{|u_1|} * m_2^{u_2} \ mod \ n$
   - Calculez $m_1^{-1} \ mod \ n$ (Aide: coefficients de bezout)
   - A l'aide des résultats précédents, retrouvez m, le message en clair original

2 Diffie Hellman

On rappelle tout d'abord le problème `LogarithmeDiscret`:

- Entrées:
  - p un nombre premier
  - α en générateur de Z_p^* (c'est à dire un entier t.q $\{a^i \ modp \ \mid \ i > 0\} = Z_p^*$)
  - un entier $x \in Z_p^*$
- Sortie:
  - b t.q $\alpha^b = x$

2.1 Echange de clés

1. Rappelez le protocole d'échange de clé de `Diffie Hellman`. Sur quels problèmes (difficiles) est-il basé ?
2. Soit k la clé échangée entre Alice et Bob via `Diffie Hellman`. Démontrez que connaître k est aussi facile que
   - le problème `LogarithmeDiscret`
   - le problème `DiffieHellman`
3. Imaginons que la communication entre Alice et Bob ne soit pas sûre, i.e Oscar peut détourner le canal de communication entre eux deux
   (a) Comment peut faire Oscar pour écouter en clair les converstions entre Alice et Bob ?
   (b) Quelle propriété manque dans l'algorithme d'échange de clés de `Diffie Hellman` ?
4. On suppose que chaque participant possède un couple de clés publique et privée. Proposez un nouvel algorithme qui ne souffre pas de ce défaut (plusieurs solutions possibles). Vous préciserez notamment la génération des clés.

2.2 Cryptosystème basé sur Diffie Hellman

1. Bob veut envoyer un message chiffré à Alice. Proposez un cryptosystème à clé publique s'insiprant du problème `Diffie Hellman` (et donc forcément de `LogarithmeDiscret`).
   (a) Quelle est la clé publique de Bob ? Quelle est sa clé privée ?
   (b) Quels sont les échanges entre Alice et Bob ?

3 Mise en pratique

Nous allons maintenant **utiliser** le chiffrement et la signature asymétrique, à travers l'envoi de mails chiffrés et signés. Pour ce faire, nous allons utiliser (sous windows, mais la manip est identique sous linux) le client OpenPGP ainsi que le plugin Thunderbird EnigMail.

3.1 Installation

1. Démarrez sous windows et installez le plugin Thunderbird Enigmail.
2. Téléchargez le client windows GPG à l'adresse `ftp://ftp.gnupg.org/gcrypt/binary/gnupg-w32cli-1.4.0a.zip`. NB: ce n'est pas la dernière version, mais elle est "installable" sans les privilèges administrateur. Si vous voulez l'installer chez vous, utilisez plutôt la dernière version.
3. Dezippez l'archive dans `Z:\gpg`
4. Sous Thunderbird, `OpenPGP->Préférences`:
   - cochez "mode expert"
   - cochez "outrepasser avec" et selectionner l'exécutable `z:\gpg\gpg.exe`
   - dans l'onglet "avancé", ajoutez dans "Parametres supplémentaires pour GnuPG" la ligne:

     ```
     --homedir z:\gpg
     ```

3.2 Utilisation

Organisez-vous en groupe de 3 élèves (A,B et C). Chaque élève doit avoir effectué l'installation.

1. Générez-vous un couple clé privée/clé publique.
   - A votre avis, à quoi sert la phrase secrète ?
   - Exportez ces clés vers un fichier .asc et regardez rapidement leur format
2. Echangez-vous vos clés publiques par mail (OpenPGP->Attacher ma clé publique)
3. Vérifiez que si A envoie un message chiffré à B, C ne peut pas le déchiffrer
4. Vérifiez que si A envoie un message signé en se faisant passer pour B, C le découvrira
5. Bonus $0,5$ points: envoyez-moi un message chiffré (ma clé publique est disponible sur `www.labri.fr/~ramet`)