Cryptographie asymétrique

19 septembre 2023

1 Intro

L'asymétrique ne sert pas à chiffrer mais plutot aux echanges de clés, etc..
Ansi recommande

- Des clés de 80 à 100 bits pour un niveau moyen de sécu (données ne durant pas dans le temps $\sim$ minutes)
- $>$ 100 bits : forts

2 Arithmétique entiers

La complexité est calc en fonction de :

- La taille des données.
- ex : un entier n en représentation binaire est en $log_2(n) = log(n)$.

A regarder : table de soustraction binaire lol.

2.1 multiplication

$$\begin{aligned} 11101 &= a \\ \times 1101 &= b \end{aligned}$$

multiplication naive :

- Taille(b) additions d'elts de taille a.
- Complexité : Taille(a)*Taille(b)
- Memoire : Taille(a*b)=Taille(a)+Taille(b)

Méthode de Karatsuba : $a, b \in \mathbb{N}$ et $k = log(a) = log(b)$. $a = \alpha 2^{k/2} + \beta$, $b = \gamma 2^{k/2} + \delta$. On écrit :

$$ab = \alpha\gamma 2^k + (\alpha\gamma + \beta\delta - (\alpha - \beta)(\gamma - \delta))2^{k/2} + \beta\delta$$

On remarque que ya 3 multiplication d'élts de taille $k/2$ et 6 soustr/add de taille $k/2$.

- Complexité : $T(k)$ est donnée par

$$\begin{aligned}
3T(k/2) + 6O(k/2) &= 3^T(k/4) + 6 * 3O(k/4) + 6O(k/2) \\
&= 3^{log(k)} + 2ck \sum_{i=1}^{log(k)} (3/2)^i \\
&= 3^{log(k)} + 2Ck\frac{(3/2)^{log(k)} - 1}{(3/2) - 1} \\
&= \dots \\
&= O(k^{log(3)})
\end{aligned}$$

2.2 division

Division naive (euclidienne) :

- Taille(a)-Taille(b)+1 soustraction de taille Taille(b).
- Complexité : O((taille(a)taille(b)+1)taille(b)).
- Mémoire : Taille(a)-Taille(b)+1 + taille(b).

2.3 algorithme d'euclide normal/etendu

Lemme 2.3.1. *Avec $a = r_0$, $b = r_1$, $r_i = q_{i+2}r_{i+1} + r_{i+2}$. On a $r_{i+2} < r_i/2$. Sauf pour les derniers i.*

D'ou

- Au plus $log(a)$ divisions : i.e. $\sum_{i=0}^{k-1}(log(r_i)-log(r_{i+1}+1)log(r_i)) \leq log(a)(k+log(a))$
- Complexité en $log(a)^2$

Euclide étendu : $u_0 = 1$, $u_1 = 0$ et $v_0 = 0$, $v_1 = 1$ et on écrit

$$u_{i+2} = u_i - q_i u_{i+1}$$
$$v_{i+2} = v_i - q_i v_{i+1}$$

Pour calculer le pgcd :

- Complexité : $O(log^2(a))$. (exo)

A montrer :

Lemme 2.3.2. *n un entier, calcul de la racine carrée entière de n en*

$$O(log^3 n)$$

2.4 indicatrice d'euler/inversion

Proposition 2.4.1. a^{-1} *mod* n se calcule en

$$O(log^2(n))$$

grace a euclide

Definition 2.4.2. $\phi\ :\ \mathbb{Z}/n\mathbb{Z} \to \#\{0 < i \leq n\}^*$

Proposition 2.4.3. On veut $\phi(1) = 1$ pour la récursion.

Proposition 2.4.4. $\sum_{d|n}\phi(d) = n$

Ca se prouve en posant $sum_{d|n}\phi(d) = f(n)$ alors :

$$f(mn) = \sum_{d|mn}\phi(d) = \sum_{d_1|n}\sum_{d_2|m}\phi(d_1 d_2) = f(m)f(n)$$

. On écrit ducoup $f(n) = f(\prod p_i^{\alpha_i})$ et $f(p^\alpha) = \sum_{k<\alpha}\phi(p^k) = \sum_k p^k - p^{k-1} = p^\alpha$

Proposition 2.4.5. $p \neq q$ deux nombres premiers et $n = pq$. On retrouve p, q en $O(log^3(n))$ avec $n, \phi(n)$.

3 corps finis

$q = p^d$

Proposition 3.0.1.
- Complexité de l'addition/soustraction dans $\mathbb{F}_q$: $O(log(q))$
- Complexité de la mult/l'inverse dans $\mathbb{F}_q$: $O(log^2(q))$

Pour la multiplication : $2d - 2$ calculs des sommes $\sum a_i b_{j-i}$ et d mults a chaque fois puis d additions. A la fin $O(log^2(q))$.

Proposition 3.0.2. $d = gcd(n, q - 1)$ racines n-emes de l'unité dans $\mathbb{F}_q$. $\mathbb{F}_q$ admet une racine primitive ssi $n \mid q - 1$.

Pour le deuxieme truc $(g^j)^n = 1$ ssi $q - 1 \mid nj$ d'ou $q - 1/d \mid j$ et on a d valeurs possibles pour j.

3.1 résidus quadratiques

On prend $p \neq 2$:

Proposition 3.1.1. $x \mapsto \left(x^{p-1/2}\right)$ donne l'indice de $\mathbb{F}_p^{*2}$ et deux non résidus sont des puissances impaires donc le produit est une puissance paire.

Proposition 3.1.2. C'est un morphisme de groupe.

Maintenant on remplace $x \mapsto x^{p-1/2}$ par l'unique caractère abélien dans $\{\pm\}$ (Jacobi).

3.2 Calcul de racine carrée, algo de shanks tonelli

On réduit ca à un calcul de racine 2^α-eme de l'unité !

1. On écrit $p - 1 = 2^\alpha * s$, s impair.
2. $r = a^{(s+1)/2}$
3. on résoud $x^2 a^{-1} \equiv 1 \; mod \; p$
4. En gros : $1 \equiv a^{(p-1)/2} \equiv a^{2^{\alpha-1}s} \equiv (r^2 a^{-1})^{2^{\alpha-1}} \; mod \; p$
5. D'ou on cherche une racine de l'unité, z, alors $z^2 \equiv y$ avec $y = r^2 a^{-1}$.

6. $z^2y \equiv y^{2^{\alpha-1}}\ mod\ p$ d'ou $(z^2y^{1-2^{\alpha-1}})^{2^{\alpha-1}} \equiv z^{2^\alpha}(y^{2^{\alpha-1}})^{1-2^{\alpha-1}} \equiv z^{2^\alpha} \equiv 1\ mod\ p$

7. D'ou il faut trouver une racine 2^α-eme de l'unité.

Determination de la racine 2^α-eme de l'unité :

1. Pour $\left(\frac{n}{p}\right) = -1$ on pose $b = n^s$

2. Alors $|b|^{2^\alpha}$.

On cherche ensuite le b^j tel que $b^{2j}r^2a^{-1} \equiv 1\ mod\ p$, on écrit $j = j_0 + 2j_1 + ... + 2^{\alpha-1}j_{\{}\alpha - 1\}$:

1. $b^{2j}r^2a^{-1} \equiv b^{2j_0+...+2^\alpha j_{\alpha-1}} \equiv b^{2j_0+...+2^{\alpha-1}j_{\alpha-2}}\ mod\ p$

2. On regarde $(b^{2j}r^2a^{-1})^{2^{\alpha-2}} \equiv (b^{2^{\alpha-1}})^{j_0}a^{2^{\alpha-2}s}\ mod\ p$

3. Comme $b^{2^{\alpha-1}} \equiv n^{(p-1)/2} \equiv -1\ mod\ p$

4. Alors pour avoir $(b^{2j}r^2a^{-1})^{2^{\alpha-2}} \equiv 1$ il faut prendre $j_0 = 0$ ssi $(r^2a^{-1})^{2^{\alpha-1}}$

Maintenant pour les autres coeffs que j_0, on suppose qu'on connait les $l < \alpha - 2$ premiers tq $((b^{j_0+...+2^lj_l})r^2a^{-1})^{2^{\alpha-2-l}}\ mod\ p$ on cherche j_{l+1} tq :

1. $((b^{j_0+...+2^lj_l})r^2a^{-1})^{2^{\alpha-2-l}}\ mod\ p$

2. On a $(b^j)^{2^{\alpha-2-l}}(r^2a^{-1})^{2^{\alpha-2-l-1}} \equiv b^{2^{\alpha-2-l}(j_0+2j_1+...+2^lj_l)}b^{2\alpha-1j_{l+1}}b^{2^\alpha(...)}...\ mod\ p$

3. A nouveau on a $b^{2^{\alpha-1}j_{l+1}} \equiv (-1)^{j_{l+1}}$

4. Et donc on pose $j_{l+1} = 0$ ssi $((b^{j_0+...+2^lj_l})^2r^2a^{-1})^{2^{\alpha-2-l-1}} \equiv 1\ mod\ p$

4 Protocoles de cryptographie à clef publique

Basé sur le principe de Kerkhoff.

Crypto symétrique	**Crypto asymétrique**
+ rapide	+lent
1 clef partagée	2 clefs
$\times$	Mise en reseau facile
Taille de clef petite	Taille de clé grande

Probleme de la crypto sym : nombre quadratique de clé par rapport au nb de personnes face a linéaire pour l'asym. (+ faut pouvoir échanger les clés)

Cryptographie asymétrique :

1. Authentification
2. Echange de clefs
3. Signature

Etant donné une fct de chiffrement asym f :

- $f(m, k_{pub}) = c$
- $f^{-1}(c, k_{priv}) = m$

Authentification par challenge :

- $f(challenge, k_{pub}) \to c$ un challenge est donné et doit être dechiffré
- $challenge = m \leftarrow f^{-1}(c, k_{priv})$

Echange de clefs:

- k la clef de session qu'on veut partager
- $f(k, k_{pub}) \to c$
- $k = f^{-1}(c, k_{priv})$

Signature d'un message :

- $f^{-1}(m, k_{priv}) = sign$

- $f(sign, k_{pub}) = m$

Propriétés d'une signature :

1. Non-répudiable (irrévocable, on peut pas dire qu'on l'a pas signé)
2. Le message est non-modifiable : inaltérable
3. Authentique
4. Non-réutilisable
5. Infalsifiable

4.1 RSA

Décrit ici. On regarde des attaques sur RSA, les p, q doivent être tous achetés !

Definition 4.1.1. Attaque par module commun

Etant donné une communauté de k personnes ayant tous.tes $p * q = n$. Chaque utilisateurs recoit $(N, e_i(publique), d_i(privee))$. Si on connait e_i, d_i alors on sait que $e_i d_i \equiv 1 \ mod \ \phi(n)$ d'ou $e_i d_i = 1 + k\phi(n)$.
On pose $m = e_i d_i - 1 = k\phi(n)$ d'ou $\forall a \in (\mathbb{Z}/N\mathbb{Z})^\times$,

$$a^m \equiv 1 \ mod \ \phi(n)$$

Or $4 \mid \phi(n)$ donc $4 \mid m$.
Donc/etant donné

$$a^m \equiv 1 \ mod \ n$$

. On a $a^{m/2}$ est une racine carrée de 1 mod n(y'en a 4). Si $a^{m/2} \equiv \alpha \neq \pm 1 \ mod \ n$ alors $(\alpha - 1)(\alpha + 1) \equiv 0 \ mod \ N$ et $gcd(\alpha - 1, n) \neq 1$ et $gcd(\alpha - 1, n) = p$. Soit $a \in (\mathbb{Z}/n\mathbb{Z})^\times$. On pose : $m = 2^t s$

- On calc $a^s \ mod \ n$, si $= \pm 1 \ mod \ n$ on change a.
- Sinon on calc successivement $a^{2^i s} \ mod \ n$. Et on s'arrete des qu'on trouve 1.
- Si a l'étape d'avant on change a.
- sinon on a trouvé α.

Autre attaque : Si on chiffre m pour deux destinataire :

- $c_1 \equiv r^{e_1}\ mod\ n$
- $c_2 \equiv r^{e_2}\ mod\ n$

Si $gcd(e_1, e_2) = 1$ alors $\exists u, v \in \mathbb{Z}$ tq $ue_1 + ve_2 = 1$. Donc $c_1^u * c_2^v = m\ mod\ n$.

- Besoin d'une fonction de hachage pour la signature

Alice ne veut pas signer m, Marvin choisit $r \in (\mathbb{Z}/n\mathbb{Z})^\times$ et calcule $m' = m * r^e\ mod\ n$. Alice signe m', donc Marvin obtient $sign(m') \equiv m'^d \equiv (mr^e) \equiv m^d r$.

Nouvelle attaque

Definition 4.1.2. par exposant publique petit :

On propose que tout le monde utilise le même e petit pour accélerer le chiffrement: m est chiffré par k utilisateurs differents: $\begin{cases} c_1 \equiv m^e \quad mod\ n_1 \\ \vdots \\ c_k \equiv m^e \quad mod\ n_k \end{cases}$ Soit les n_i sont premiers entre eux et on fait un lemme chinois, si $e < k$, $m^e < \prod_i n_i$. Si pas premiers entre eux : gros pb.

Definition 4.1.3. Attaque par petit exposant privé, but : améliorer la vitesse de déchiffrement.

Théorème 4.1.4. *Soit $N = pq$ avec $q < p < 2q$ et $d = 1/3\sqrt[4]{n}$. Etant donné le couple (n, e) avec $ed \equiv 1\ mod\ \phi(n)$, on peut retrouver efficacement d.*

Preuve : On pose $ed - k\phi(n) = 1$. D'ou $\frac{e}{\phi(n)} - \frac{k}{d} = \frac{1}{d\phi(n)}$. On approche $\phi(n)$ par n et en utilisant le fait $d < 1/3\sqrt[4]{n}$ on a :

$$|\frac{e}{n} - \frac{k}{d}| < \frac{1}{2d^2}$$

En passant par un dév en fractions continues à la bonne précision on retrouve k/d. □

RSA est pas indistinguable.(exponentiation binaire est rapide)

4.2 Probleme de log discret

Securité dépend du groupe dans lequel on travaille : Si on prend $G = (\mathbb{Z}/p\mathbb{Z}, +)$ et $h = gx \ mod \ p$ alors $x = hg^{-1}$, une étape.

Definition 4.2.1. Problème de Diffie-Hellman(DHP): Etant donnés g, g^a, g^b peut-on trouver g^{ab}.

Definition 4.2.2. Signature d'El Gamal : k doit être secret et d'usage unique.

- k doit être secret : a faire
- k doit être d'usage unique : pareil

5 IGC(infrastucture de gestion de clefs)

Gère les distributions de certificats.

5.1 Création d'un certificat num

2 modes:

- Decentralisé: l'utilisateur crée son bi-clef
- Centralisé: L'autorité de confiance crée le bi-clef

5.2 Révocation d'un certificat num

- Fin de limite de validité
- Révoque avant la date limite(par exemple si oubli du mot de passe)

5.3 Types de certificats

- Signature
- Chiffrement et signature

5.4 Reste

L'IGC est composée de 4 entités obligatoires:

- Autorité de confiance: signe les certifs
- Autorité d'enregistrement:s'assure de l'identité du demandeur de certifs.
- Autorité de dépôt: stocke les certifs et liste de révocations
- Entité finale: celle qui demande le certif

Optionnellement: Autorité de sequestre, conserve les clefs privées