Application de la complexité en cryptographie

Judicaël Courant

Lycée La Martinière-Monplaisir, Lyon

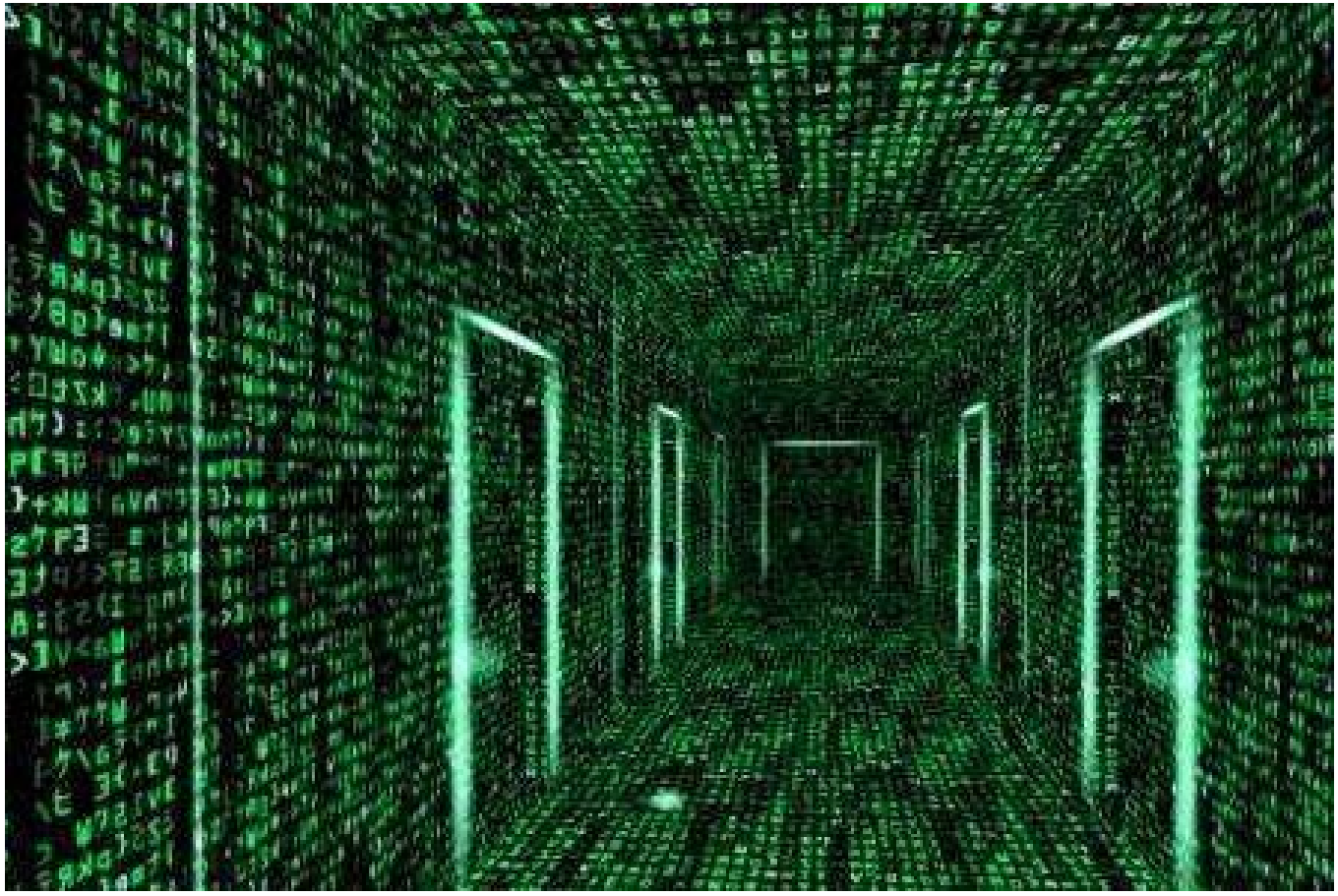
Conférence Algorithmique et Programmation
CIRM
Le 6 mai 2016

Lignes directrices

Lignes directrices

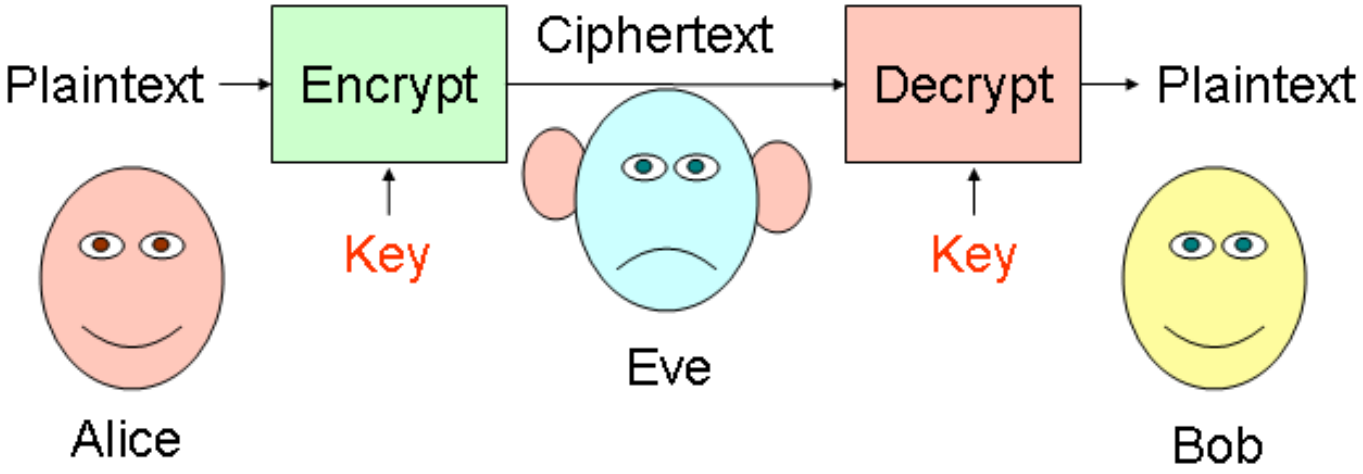
Cryptographie : science (art ?) de la confidentialité

« Tout est nombre » (Pythagore)



Message (de taille bornée) = Suite de bits (finie) = Entier (borné)

Chiffrement symétrique

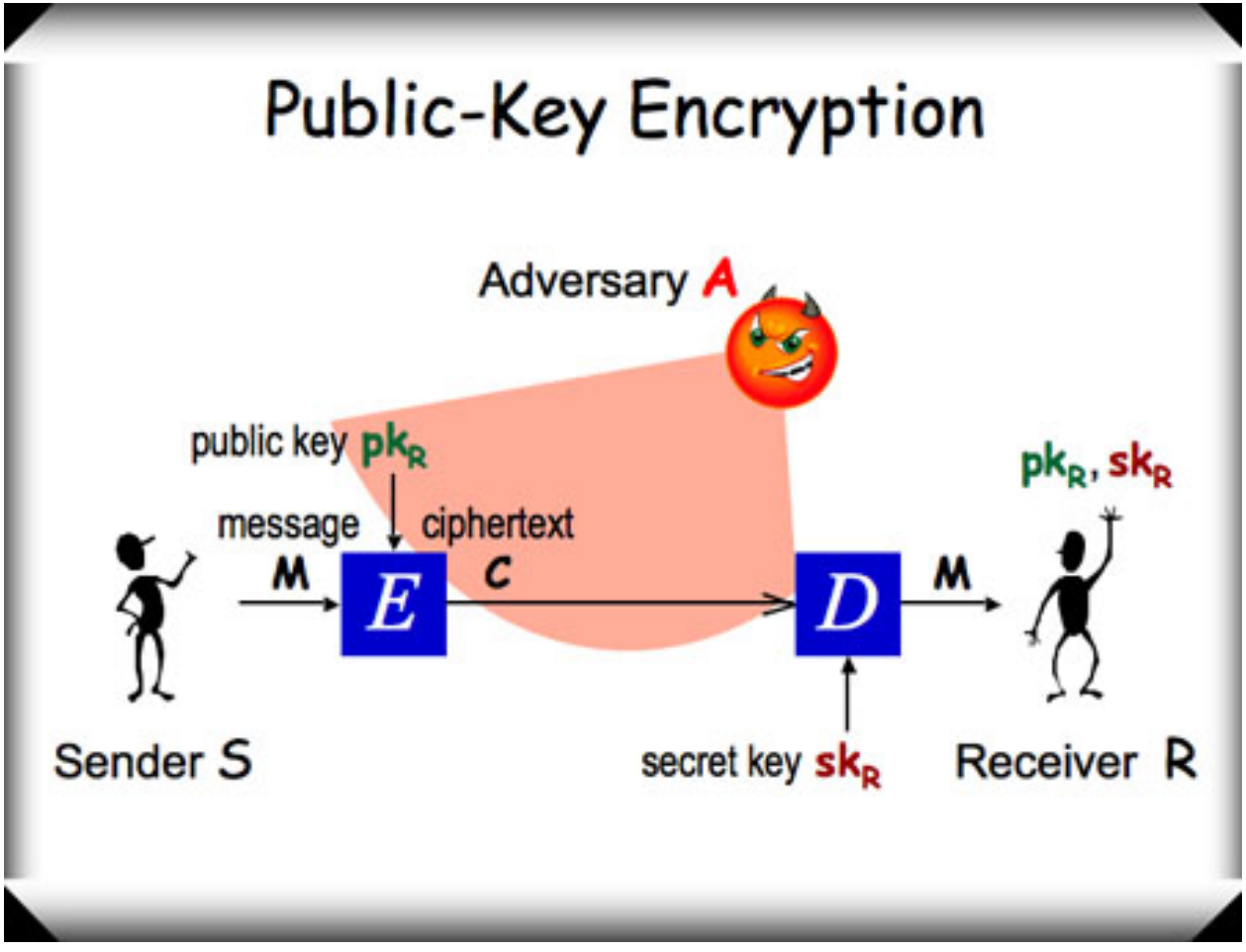


Ciphertext = Encrypt (Plaintext, Key)
Plaintext = Decrypt (Ciphertext, Key)

plaintext : message (en clair)
ciphertext : chiffré

to encrypt : chiffrer
to decrypt : déchiffrer.

Chiffrement asymétrique



Lignes directrices

Exemple : RSA (1978)

Si p, q premiers distincts, et $ed \equiv 1 \mod (p-1)(q-1)$, alors

$$\forall x \in \mathbb{Z}_{pq} \quad (x^e)^d = x$$

Cryptosystème RSA :

- clé publique $\mathbf{pk} = (pq, e)$
- clé secrète $\mathbf{sk} = (pq, d)$
- $\mathsf{Enc}((pq, e), m) = m^e$
- $\mathsf{Dec}((pq, d), c) = c^d$.

Correction

Le déchiffrement d'un chiffré redonne le message en clair :

$$\forall m \in \mathbb{Z}_{pq} \quad \mathsf{Dec}(\mathsf{sk}, \mathsf{Enc}(\mathsf{pk}, m)) = m$$

C'est juste une autre façon d'écrire

$$\forall x \in \mathbb{Z}_{pq} \quad (x^e)^d = x$$

Sécurité (de RSA)

Argument rapide :

- Pour casser RSA, on a besoin de calculer e à partir de d.
- Pour cela il faut connaître $(p-1)(q-1)$.
- Pour cela, on a besoin de factoriser pq.
- Or c'est difficile quand p et q sont grands.
- Donc RSA est sûr.

Sécurité (de RSA)

Reprenons lentement :

Sécurité (de RSA)

Reprenons lentement :

- Si factoriser pq est facile on peut facilement trouver $\mathbf{sk}$ à partir de $\mathbf{pk}$.

Sécurité (de RSA)

Reprenons lentement :

- Si factoriser pq est facile on peut facilement trouver $\mathbf{sk}$ à partir de $\mathbf{pk}$.
- Or factoriser n'est pas facile.

Sécurité (de RSA)

Reprenons lentement :

- Si factoriser pq est facile on peut facilement trouver $\mathbf{sk}$ à partir de $\mathbf{pk}$.
- Or factoriser n'est pas facile.
- Donc trouver $\mathbf{sk}$ à partir de $\mathbf{pk}$ n'est pas facile.

Sécurité (de RSA)

Reprenons lentement :

- Si factoriser pq est facile on peut facilement trouver $\mathbf{sk}$ à partir de $\mathbf{pk}$.
- Or factoriser n'est pas facile.
- Donc trouver $\mathbf{sk}$ à partir de $\mathbf{pk}$ n'est pas facile.

Syllogisme grec bien connu :

Sécurité (de RSA)

Reprenons lentement :

- Si factoriser pq est facile on peut facilement trouver $\mathbf{sk}$ à partir de $\mathbf{pk}$.
- Or factoriser n'est pas facile.
- Donc trouver $\mathbf{sk}$ à partir de $\mathbf{pk}$ n'est pas facile.

Syllogisme grec bien connu :

- Si un être est humain, alors il est mortel.

Sécurité (de RSA)

Reprenons lentement :

- Si factoriser pq est facile on peut facilement trouver $\mathbf{sk}$ à partir de $\mathbf{pk}$.
- Or factoriser n'est pas facile.
- Donc trouver $\mathbf{sk}$ à partir de $\mathbf{pk}$ n'est pas facile.

Syllogisme grec bien connu :

- Si un être est humain, alors il est mortel.
- Or mon chat n'est pas humain.

Sécurité (de RSA)

Reprenons lentement :

- Si factoriser pq est facile on peut facilement trouver $\mathbf{sk}$ à partir de $\mathbf{pk}$.
- Or factoriser n'est pas facile.
- Donc trouver $\mathbf{sk}$ à partir de $\mathbf{pk}$ n'est pas facile.

Syllogisme grec bien connu :

- Si un être est humain, alors il est mortel.
- Or mon chat n'est pas humain.
- Donc mon chat n'est pas mortel.

C'est bien la peine de faire des maths...

Lignes directrices

Conclusion provisoire

- Il y a besoin d'une vraie modélisation du problème
- Utiliser RSA sans comprendre, c'est prendre des risques...

Lignes directrices

Casser le chiffrement

Faisable en principe :

1. intercepter la clé publique (facile)
2. intercepter le chiffré c (demander à PRISM ou ECHELON)
3. énumérer tous les messages possibles, calculer leurs chiffrés c'
4. s'arrêter quand $c' = c$.

Infaisable en pratique : trop long (2^N messages de longueur N)

Première idée

Idée 1 :

$$\text{Problème de sécurité} \quad = \quad \text{Problème de complexité}$$

Un chiffrement sera solide s'il n'existe pas d'algorithme *efficace* pour le casser.

Modèle de complexité

Paramètre de sécurité Un entier N représentant la taille des clés (nombre de chiffres) et des messages.

Étude asymptotique Efficace = complexité polynomiale.

Propriétés à étudier

Alice envoie un message m chiffré en c à Bob. Peut-on :

- Trouver m à partir de c dans le cas général ? (notion de fonction à sens unique)
- Trouver m si on sait que le message était choisi parmi un ensemble restreint de possibilité, par exemple 0, ..., 9 ? (sécurité sémantique)
- Construire le chiffré de $2m$ à partir de c sans connaître m ? (non-malléabilité)

RSA : bilan

RSA : bilan

Fonction à sens unique Problème ouvert pour RSA.

RSA : bilan

Fonction à sens unique Problème ouvert pour RSA.

Malléabilité Oui.

$$\mathsf{Enc}(\mathsf{pk}, m \times m') = \mathsf{Enc}(\mathsf{pk}, m) \times \mathsf{Enc}(\mathsf{pk}, m').$$

RSA : bilan

Fonction à sens unique	Problème ouvert pour RSA.
Malléabilité	Oui. $\mathsf{Enc}(\mathsf{pk}, m \times m') = \mathsf{Enc}(\mathsf{pk}, m) \times \mathsf{Enc}(\mathsf{pk}, m')$.
Sécurité sémantique	Non. Calculer $\mathsf{Enc}(\mathsf{pk}, i)$ pour $i = 0 \dots 9$.

RSA : bilan

Fonction à sens unique	Problème ouvert pour RSA.
Malléabilité	Oui. $\mathsf{Enc}(\mathsf{pk}, m \times m') = \mathsf{Enc}(\mathsf{pk}, m) \times \mathsf{Enc}(\mathsf{pk}, m')$.
Sécurité sémantique	Non. Calculer $\mathsf{Enc}(\mathsf{pk}, i)$ pour $i = 0 \dots 9$.
Et en pratique ?	GPG *utilise* RSA mais GPG $\neq$ RSA.

Lignes directrices

Remarque sur la sécurité sémantique

- RSA n'a pas la propriété de sécurité sémantique car on peut calculer $\mathsf{Enc}(\mathsf{pk}, i)$ pour $i = 0 \dots 9$
- C'est vrai pour tout algorithme de chiffrement à clé publique
- ...**à condition** qu'en chiffrant deux fois un même message, on obtienne le même résultat.
- Donc les algorithmes de chiffrement sûrs ne sont pas déterministes.

Au fait

- Si l'adversaire ne peut pas gagner à tous les coups mais seulement une fois sur 10, c'est gênant ?
- Commerce électronique mondial : 10^{12} € par an.

Cadre probabiliste

Idée 2 : **Cadre probabiliste**

- Algorithme de chiffrement **probabiliste.**
- L'adversaire peut jouer aux dés.
- On veut que la probabilité de succès de l'adversaire soit une fonction *négligeable* du paramètre de sécurité N.

Définition (Négligeable)

Est un $O_{N\to+\infty}(N^{-k})$ pour tout $k \in \mathbb{N}$.

Lignes directrices

Modélisation de l'adversaire

Pour cet exposé l'adversaire est :

- un algorithme
- probabiliste
- polynomial
- omniscient sur le réseau (accès aux messages entre Alice à Bob).
- passif (à la différence de la NSA)

Et si on jouait ?

On peut modéliser la sécurité comme un **jeu** à deux joueurs :

- Le challenger (algorithme modélisant Alice et Bob)
- L'adversaire (algorithme modélisant Ève)

Question : quelles chances de succès pour l'adversaire ?

Le jeu de la sécurité sémantique

- On donne à l'adversaire la clé publique utilisée. L'adversaire choisit deux messages m_0 et m_1 et les donne au challenger.
- Le challenger :
  - choisit $b \leftarrow U(\{0,1\})$
  - calcule le chiffré $c_b \leftarrow \mathsf{Enc}(\mathsf{pk}, m_b)$
  - donne c_b à l'adversaire
- L'adversaire calcule une valeur b'.
- Si $b' = b$, l'adversaire a gagné.

Quantifier la dangerosité de l'adversaire

- Si $\Pr[b' = 1 | b = 1] = \Pr[b' = 1 | b = 0]$, l'adversaire n'a aucun succès.
- Un adversaire est dangereux si $\Pr[b' = 1 | b = 1]$ et $\Pr[b' = 1 | b = 0]$ sont *significativement* différents.

Indistinguabilité

Autre présentation du jeu :

- $\mathsf{Enc}(\mathsf{pk}, m_b)$ peut être vu comme une loi de probabilité $\mathcal{D}_b$.
- L'adversaire doit tenter de *distinguer* $\mathcal{D}_0$ et $\mathcal{D}_1$.

Définition (Indistinguabilité de distributions de probabilité)

$\mathcal{D}_0$ et $\mathcal{D}_1$ indistinguables (noté $\mathcal{D}_0 \approx \mathcal{D}_1$) ssi pour tout algorithme polynomial probabiliste F à valeur dans $\{0, 1\}$ et tout k fixé

$$\left|\Pr[F(x) = 1 \,|\, x \leftarrow \mathcal{D}_0] - \Pr[F(x) = 1 \,|\, x \leftarrow \mathcal{D}_1]\right|$$

est une fonction négligeable du paramètre de sécurité N.

Relation d'indistinguabilité

L'indistinguabilité :

- est une relation d'équivalence
- passe au contexte : $\mathcal{D}_0 \approx \mathcal{D}_1 \Rightarrow f(\mathcal{D}_0) \approx f(\mathcal{D}_1)$ pour tout algorithme (probabiliste) f polynomial.

Idée 3

Idée 3 :

$$\text{Sécurité sémantique} = \text{indistinguabilité de distributions}$$

Définition (Sécurité sémantique)

Le chiffrement possède la propriété de sécurité sémantique ssi pour tous messages m_0 et m_1

$$(\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, m_0)) \approx (\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, m_1))$$

Lignes directrices

Premiers sûrs

Définition (Premier sûr)

Nombre premier de la forme $2q + 1$ avec q premier.

Dans la suite :

- p premier sûr
- $q = (p - 1)/2$ (et q premier)
- p possède N chiffres (en binaire)

Résidus quadratiques

Définition (Sous-groupe des résidus quadratiques modulo p)

Sous-groupe des carrés de $\mathbb{Z}_p^\times$:

$$G_p = \{x^2 | x \in \mathbb{Z}_p^\times\}$$

Logarithme discret

Remarque

G_p est d'ordre $q = (p-1)/2$ (avec q premier).

Corollaire

Pour tout $g \in G_p \setminus \{1\}$ fixé, tout élément de G_p s'écrit sous la forme g^x, avec $x \in \mathbb{Z}$ et x est unique modulo q.

Définition (Log discret en base g)

$DL_g(g^x) = x$ pour $g \in G_p \setminus \{1\}$ et $x \in [\![0, q[\![$.

Complexité temporelle des calculs dans G_p

N : nombre de bits de p (paramètre de sécurité).

Complexité des opérations usuelles dans G_p :

- Somme : $O(N)$.
- Produit : $O(N^2)$ (en pratique $O(N^{1,59})$ et en théorie $O(N \log N \log\log N)$).
- Calculer g^x avec $x \in [\![0, q[\![$: $O(N^3)$ (car $O(\log x)$ multiplications).
- Calculer $g^{-1} = g^{q-1}$: $O(N^3)$.
- Calculer $DL_g(x)$: $e^{\sqrt{2 \ln 2 \cdot N} + o(\sqrt{N})}$ (sous-exponentiel mais pas polynomial).

Problème du log discret

Définition (Problème du log discret)

Existe-t-il un algorithme (probabiliste) efficace prenant en entrée p, g et x ayant une probabilité non-négligeable de calculer $DL_g(x)$?

Conjecture (DL)

Non.

Lignes directrices

Le protocole (1976)

Alice et Bob veulent convenir d'un secret mais sont écoutés par Ève.

1. Convention publique : p premier fort, $g \in G_p \setminus \{1\}$.
2. Alice : $x \leftarrow U(\llbracket 0, q \llbracket)$.
3. Alice $\xrightarrow{g^x}$ Bob.
4. Bob : $y \leftarrow U(\llbracket 0, q \llbracket)$.
5. Bob $\xrightarrow{g^y}$ Alice.
6. Alice calcule $g^{xy} = (g^y)^x$.
7. Bob calcule $g^{xy} = (g^x)^y$.

Computational Diffie-Hellman Assumption

Ève n'arrivera pas à calculer g^{xy} à partir de p, g, g^x et g^y :

Conjecture (CDH)

Tout algorithme efficace prenant en entrée p, g, g^x et g^y a une probabilité négligeable de calculer g^{xy}.

Decisional Diffie-Hellman Assumption

Ève ne sait même pas faire la différence entre g^{xy} et un élément de G_p quelconque :

Conjecture (DDH)

Les deux distributions suivantes sont indistinguables :

$$\mathcal{D}\left(\left(g^x, g^y, g^{xy}\right) \mid x \leftarrow U\left(\llbracket 0, q \rrbracket\right) ; y \leftarrow U\left(\llbracket 0, q \rrbracket\right)\right)$$
$$\mathcal{D}\left(\left(g^x, g^y, g'\right) \mid x \leftarrow U\left(\llbracket 0, q \rrbracket\right) ; y \leftarrow U\left(\llbracket 0, q \rrbracket\right) ; g' \leftarrow U\left(G_p\right)\right)$$

Remarques

- Difficultés relatives des conjectures

Remarques

- Difficultés relatives des conjectures

$$DDH \Rightarrow CDH$$

Remarques

- Difficultés relatives des conjectures

$$DDH \Rightarrow CDH \Rightarrow DL$$

Remarques

- Difficultés relatives des conjectures

$$DDH \Rightarrow CDH \Rightarrow DL \Rightarrow P \neq NP$$

Remarques

- Difficultés relatives des conjectures

$$DDH \Rightarrow CDH \Rightarrow DL \Rightarrow P \neq NP$$

- Démonstration de DDH probablement non triviale...

Remarques

- Difficultés relatives des conjectures

$$DDH \Rightarrow CDH \Rightarrow DL \Rightarrow P \neq NP$$

- Démonstration de DDH probablement non triviale...
- Conjecture DDH non réfutée depuis 40 ans.

Lignes directrices

Protocole

- On fixe : p premier sûr de taille N, $g \in G_p \setminus \{1\}$.
- Clé privée : $\mathrm{sk} \leftarrow U(\llbracket 0, q \llbracket)$.
- Clé publique $\mathrm{pk} := g^{\mathrm{sk}}$.
- Messages : éléments de G_p.
- $\mathrm{Enc}(\mathrm{pk}, m) := (g^y, \mathrm{pk}^y \cdot m)$ où $y \leftarrow U(\llbracket 0, q \llbracket)$.
- $\mathrm{Dec}(\mathrm{sk}, (\alpha, \beta)) := \alpha^{-\mathrm{sk}} \cdot \beta$

Correction

$$\mathsf{Dec}(\mathsf{sk}, \mathsf{Enc}(\mathsf{pk}, m)) = \mathsf{Dec}(\mathsf{sk}, (g^y, \mathsf{pk}^y \cdot m))$$

Correction

$$\begin{aligned}\mathsf{Dec}(\mathrm{sk}, \mathsf{Enc}(\mathrm{pk}, m)) &= \mathsf{Dec}(\mathrm{sk}, (g^y, \mathrm{pk}^y \cdot m)) \\ &= (g^y)^{-\mathrm{sk}} \cdot \mathrm{pk}^y \cdot m\end{aligned}$$

Correction

$$\begin{aligned}\mathsf{Dec}(\mathrm{sk}, \mathsf{Enc}(\mathrm{pk}, m)) &= \mathsf{Dec}(\mathrm{sk}, (g^y, \mathrm{pk}^y \cdot m)) \\ &= (g^y)^{-\mathrm{sk}} \cdot \mathrm{pk}^y \cdot m \\ &= g^{-\mathrm{sk} \cdot y} \cdot \left(g^{\mathrm{sk}}\right)^y \cdot m\end{aligned}$$

Correction

$$\begin{aligned}
\mathsf{Dec}(\mathrm{sk}, \mathsf{Enc}(\mathrm{pk}, m)) &= \mathsf{Dec}(\mathrm{sk}, (g^y, \mathrm{pk}^y \cdot m)) \\
&= (g^y)^{-\mathrm{sk}} \cdot \mathrm{pk}^y \cdot m \\
&= g^{-\mathrm{sk} \cdot y} \cdot \left(g^{\mathrm{sk}}\right)^y \cdot m \\
&= m
\end{aligned}$$

Démo sécurité sémantique

Soit $m_0, m_1 \in G_p$.

- On pose $\mathcal{D}_i = (\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, m_i))$ pour $i = 0, 1$. On veut montrer $\mathcal{D}_0 \approx \mathcal{D}_1$.

Démo sécurité sémantique

Soit $m_0, m_1 \in G_p$.

- On pose $\mathcal{D}_i = (\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, m_i))$ pour $i = 0, 1$. On veut montrer $\mathcal{D}_0 \approx \mathcal{D}_1$.
- $\mathcal{D}_i = \left[\left(g^{\mathrm{sk}}, g^y, \left(g^{\mathrm{sk}}\right)^y \cdot m_i\right) \mid \mathrm{sk} \leftarrow U(\llbracket 0, q \llbracket); y \leftarrow U(\llbracket 0, q \llbracket) \right]$

Démo sécurité sémantique

Soit $m_0, m_1 \in G_p$.

- On pose $\mathcal{D}_i = (\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, m_i))$ pour $i = 0, 1$. On veut montrer $\mathcal{D}_0 \approx \mathcal{D}_1$.
- $\mathcal{D}_i = \left[\left(g^{\mathrm{sk}}, g^y, \left(g^{\mathrm{sk}}\right)^y \cdot m_i\right) \mid \mathrm{sk} \leftarrow U(\llbracket 0, q \llbracket); y \leftarrow U(\llbracket 0, q \llbracket) \right]$
- On pose $\mathcal{D}'_i = \left[\left(g^{\mathrm{sk}}, g^y, g' \cdot m_i\right) \mid \mathrm{sk} \leftarrow U(\llbracket 0, q \llbracket); y \leftarrow U(\llbracket 0, q \llbracket); g' \leftarrow U\left(G_p\right) \right]$, pour $i = 0, 1$.

Démo sécurité sémantique

Soit $m_0, m_1 \in G_p$.

- On pose $\mathcal{D}_i = (\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, m_i))$ pour $i = 0, 1$. On veut montrer $\mathcal{D}_0 \approx \mathcal{D}_1$.
- $\mathcal{D}_i = \left[\left(g^{\mathrm{sk}}, g^y, \left(g^{\mathrm{sk}}\right)^y \cdot m_i\right) \mid \mathrm{sk} \leftarrow U(\llbracket 0, q \llbracket); y \leftarrow U(\llbracket 0, q \llbracket) \right]$
- On pose $\mathcal{D}'_i = \left[\left(g^{\mathrm{sk}}, g^y, g' \cdot m_i\right) \mid \mathrm{sk} \leftarrow U(\llbracket 0, q \llbracket); y \leftarrow U(\llbracket 0, q \llbracket); g' \leftarrow U\left(G_p\right) \right]$, pour $i = 0, 1$.
- Montrons $\mathcal{D}_0 \approx \mathcal{D}'_0 = \mathcal{D}'_1 \approx \mathcal{D}_1$:

Démo sécurité sémantique

Soit $m_0, m_1 \in G_p$.

- On pose $\mathcal{D}_i = (\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, m_i))$ pour $i = 0, 1$. On veut montrer $\mathcal{D}_0 \approx \mathcal{D}_1$.
- $\mathcal{D}_i = \left[\left(g^{\mathrm{sk}}, g^y, \left(g^{\mathrm{sk}}\right)^y \cdot m_i\right) \mid \mathrm{sk} \leftarrow U(\llbracket 0, q \llbracket); y \leftarrow U(\llbracket 0, q \llbracket) \right]$
- On pose $\mathcal{D}'_i = \left[\left(g^{\mathrm{sk}}, g^y, g' \cdot m_i\right) \mid \mathrm{sk} \leftarrow U(\llbracket 0, q \llbracket); y \leftarrow U(\llbracket 0, q \llbracket); g' \leftarrow U\left(G_p\right) \right]$, pour $i = 0, 1$.
- Montrons $\mathcal{D}_0 \approx \mathcal{D}'_0 = \mathcal{D}'_1 \approx \mathcal{D}_1$:
  - $DDH \Rightarrow \mathcal{D}_i \approx \mathcal{D}'_i$ (par passage au contexte)

Démo sécurité sémantique

Soit $m_0, m_1 \in G_p$.

- On pose $\mathcal{D}_i = (\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, m_i))$ pour $i = 0, 1$. On veut montrer $\mathcal{D}_0 \approx \mathcal{D}_1$.
- $\mathcal{D}_i = \left[\left(g^{\mathrm{sk}}, g^y, \left(g^{\mathrm{sk}}\right)^y \cdot m_i\right) \mid \mathrm{sk} \leftarrow U(\llbracket 0, q \llbracket); y \leftarrow U(\llbracket 0, q \llbracket) \right]$
- On pose $\mathcal{D}'_i = \left[\left(g^{\mathrm{sk}}, g^y, g' \cdot m_i\right) \mid \mathrm{sk} \leftarrow U(\llbracket 0, q \llbracket); y \leftarrow U(\llbracket 0, q \llbracket); g' \leftarrow U(G_p) \right]$, pour $i = 0, 1$.
- Montrons $\mathcal{D}_0 \approx \mathcal{D}'_0 = \mathcal{D}'_1 \approx \mathcal{D}_1$:
  - $DDH \Rightarrow \mathcal{D}_i \approx \mathcal{D}'_i$ (par passage au contexte)
  - $\mathcal{D}'_0 = \mathcal{D}'_1$, donc $\mathcal{D}'_0$ et $\mathcal{D}'_1$ indistinguables.

Démo sécurité sémantique

Soit $m_0, m_1 \in G_p$.

- On pose $\mathcal{D}_i = (\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, m_i))$ pour $i = 0, 1$. On veut montrer $\mathcal{D}_0 \approx \mathcal{D}_1$.
- $\mathcal{D}_i = \left[\left(g^{\mathrm{sk}}, g^y, \left(g^{\mathrm{sk}}\right)^y \cdot m_i\right) \mid \mathrm{sk} \leftarrow U(\llbracket 0, q \llbracket); y \leftarrow U(\llbracket 0, q \llbracket) \right]$
- On pose $\mathcal{D}'_i = \left[\left(g^{\mathrm{sk}}, g^y, g' \cdot m_i\right) \mid \mathrm{sk} \leftarrow U(\llbracket 0, q \llbracket); y \leftarrow U(\llbracket 0, q \llbracket); g' \leftarrow U\left(G_p\right) \right]$, pour $i = 0, 1$.
- Montrons $\mathcal{D}_0 \approx \mathcal{D}'_0 = \mathcal{D}'_1 \approx \mathcal{D}_1$:
  - $DDH \Rightarrow \mathcal{D}_i \approx \mathcal{D}'_i$ (par passage au contexte)
  - $\mathcal{D}'_0 = \mathcal{D}'_1$, donc $\mathcal{D}'_0$ et $\mathcal{D}'_1$ indistinguables.
- Donc $DDH \Rightarrow \mathcal{D}_0 \approx \mathcal{D}_1$.

Conclusion (provisoire)

Le chiffrement de ElGamal possède la propriété de sécurité sémantique, sous l'hypothèse que DDH est difficile.

Conclusion

- La sécurité des algorithmes de chiffrement peut être modélisée :
  - complexité
  - algorithmes probabilistes
  - notion d'indistinguabilité de distributions de probabilité
- Elle peut être démontrée rigoureusement sous certaines hypothèses (qui impliquent $P \neq NP$).

Pour aller plus loin

- Il y a d'autres propriétés de sécurité à étudier avant de choisir un algorithme de chiffrement :
  - ElGamal est malléable
  - etc.
- Mettre au point un algorithme de chiffrement sûr est délicat.
- Avoir une porte blindée ne sert à rien si on a des murs en carton.

Relativisons l'importance de la crypto

Relativisons l'importance de la crypto