

Aperçu de la Cryptographie Moderne

Le Vote Électronique

Fabrice Mouhartem

École Normale Supérieure de Lyon

Journée Alkindi, 15 Juin 2016



Cryptographie

Chiffrement : Utilisation d'un secret pour rendre un message inintelligible sans ce secret.

NVSRVYCZVYZEOWVSAO

Cryptographie

Chiffrement : Utilisation d'un secret pour rendre un message inintelligible sans ce secret.

NVSRVYCYVYZEOWVSAO
BONJOURTOUTLEMONDE

Cryptographie symétrique :

On partage le même secret
avec son interlocuteur

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
Q	N	K	A	O	I	U	F	D	R	J	E	W	S	V	H	P	C	T	Z	Y	G	B	L	X	M

Cryptographie

Chiffrement : Utilisation d'un secret pour rendre un message inintelligible sans ce secret.

NVSRVYCZVYZEOWVSAO
BONJOURTOUTLEMONDE

Cryptographie symétrique :

On partage le même secret
avec son interlocuteur

ABCDEFGHIJKLMNOPQRSTUVWXYZ
↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓
QNKAOIUFD RJ EWSVHPCTZYGBLXM

Problème

Comment échanger ce secret ?

Échange de Clefs



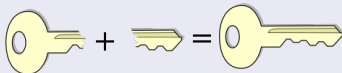
Proposé par W. Diffie et M. Hellman en 1976

Échange de Clefs



Proposé par W. Diffie et M. Hellman en 1976

Idée : Envoyer deux morceaux de la clef et chacun peut ensuite reconstruire la clef de son côté

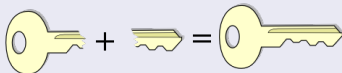


Échange de Clefs



Proposé par W. Diffie et M. Hellman en 1976

Idée : Envoyer deux morceaux de la clef et chacun peut ensuite reconstruire la clef de son côté



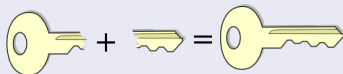
Utilisé dans SSL ([http**s**](https))

Échange de Clefs



Proposé par W. Diffie et M. Hellman en 1976

Idée : Envoyer deux morceaux de la clef et chacun peut ensuite reconstruire la clef de son côté



Utilisé dans SSL ([https](https://))

Autre solution : Le chiffrement à **clef publique** (RSA, El Gamal, ...)

Mais ce n'est pas tout

1976 Échange de Clefs Diffie-Hellman

1977 Chiffrement RSA

Mais ce n'est pas tout

1976 Échange de Clefs Diffie-Hellman

1977 Chiffrement RSA

1978 Utilisation de la malléabilité

1984 Chiffrement fondé sur l'identité (IBE)

1987 Multiparty Computation

Mais ce n'est pas tout

1976 Échange de Clefs Diffie-Hellman

1977 Chiffrement RSA

1978 Utilisation de la malléabilité

1984 Chiffrement fondé sur l'identité (IBE)

1987 Multiparty Computation

1997 Vote électronique

Vote Électronique

On distingue deux types de votes électroniques :

Avec des machines à voter

- Une urne physique
- Vérification habituelle avec la CNI par exemple



Via Internet

- Depuis chez soi
- Avec sa machine personnelle



Vote Électronique

On distingue deux types de votes électroniques :

Avec des machines à voter

- Une urne physique
- Vérification habituelle avec la CNI par exemple



Via Internet

- Depuis chez soi
- Avec sa machine personnelle



On s'intéressera au vote par internet

Primitive Cryptographique

Le **vote électronique** est une **primitive** cryptographique.

Primitive Cryptographique

Le **vote électronique** est une **primitive** cryptographique.

C'est-à-dire la donnée d'**algorithmes** et de **fonctionnalités**

Primitive Cryptographique

Le **vote électronique** est une **primitive** cryptographique.

C'est-à-dire la donnée d'**algorithmes** et de **fonctionnalités**

Exemple de Primitive Cryptographique

Le **chiffrement** est une autre primitive.

C'est la donnée de trois algorithmes :

- **GenClefs** qui prend en entrée une sécurité et renvoie (pk, sk)
- **Chiffrer** qui prend en entrée un **message** et pk et renvoie un **chiffré** C
- **Déchiffrer** qui prend en entrée un **chiffré** et sk et renvoie un **message** M

Primitive Cryptographique

Le **vote électronique** est une **primitive** cryptographique.

C'est-à-dire la donnée d'**algorithmes** et de **fonctionnalités**

Exemple de Primitive Cryptographique

Le **chiffrement** est une autre primitive.

C'est la donnée de trois algorithmes :

- **GenClefs** qui prend en entrée une sécurité et renvoie (pk, sk)
- **Chiffrer** qui prend en entrée un **message** et pk et renvoie un **chiffré** C
- **Déchiffrer** qui prend en entrée un **chiffré** et sk et renvoie un **message** M

Et de deux fonctionnalités :

- **Correction** $\text{Chiffrer}(\text{Déchiffrer}(M)) = M$
- **Sécurité** on ne peut **rien** dire sur le message

Vote Électronique : Définition

Le vote électronique est défini par trois algorithmes :

- **GenClefs** où on génère les clefs des votants
- **Vote** où un votant envoie son vote
- **Ouvrir** qui clôt le vote et renvoie le nombre de voix

Vote Électronique : Définition

Le vote électronique est défini par trois algorithmes :

- **GenClefs** où on génère les clefs des votants
- **Vote** où un votant envoie son vote
- **Ouvrir** qui clôt le vote et renvoie le nombre de voix

Et des fonctionnalités :

- **Confidentialité du vote** personne ne connaît mon vote
- **Résistant au chantage** personne ne connaît mon vote. . . même si je veux le dire
- **Vérifiabilité**
 - ▶ *Individuelle* : Je peux vérifier que j'ai correctement voté
 - ▶ *Universelle* : Tout le monde peut vérifier que le résultat final correspond bien au contenu de l'urne
 - ▶ *Éligibilité* : Tout le monde peut vérifier que les bulletins ont bien été émis par un votant légitime

Vote Électronique

Législation

Il y a des pays qui essayent le vote électronique : Australie, France, Estonie, Suisse, Canada, ...

Et des pays qui l'ont interdit : Pays-Bas, Allemagne, Norvège, ...

Vote Électronique

Législation

Il y a des pays qui essayent le vote électronique : Australie, France, Estonie, Suisse, Canada, ...

Et des pays qui l'ont interdit : Pays-Bas, Allemagne, Norvège, ...

Il y a eu des attaques sur les tentatives de mise en place sur les systèmes physiques : Inde, ÉUA, ...

Vote Électronique

Législation

Il y a des pays qui essaient le vote électronique : Australie, France, Estonie, Suisse, Canada, ...

Et des pays qui l'ont interdit : Pays-Bas, Allemagne, Norvège, ...

Il y a eu des attaques sur les tentatives de mise en place sur les systèmes physiques : Inde, ÉUA, ...

Utilisé dans les entreprises (élections CA, décisions,...) : EDF, RATP, Éducation Nationale, ...



Plan

Introduction

Outils

Construction

Conclusion

Sécurité

Pour établir la sécurité d'un *schéma* (par exemple RSA) on utilise des **preuves de sécurité**.

Sécurité

Pour établir la sécurité d'un *schéma* (par exemple RSA) on utilise des **preuves de sécurité**.

Ces preuves reposent sur des problèmes supposés difficiles (factorisation, logarithme discret, . . .)

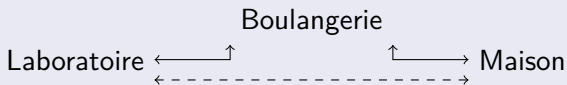
Sécurité

Pour établir la sécurité d'un *schéma* (par exemple RSA) on utilise des **preuves de sécurité**.

Ces preuves reposent sur des problèmes supposés difficiles (factorisation, logarithme discret, ...)

Réduction

Les preuves cryptographiques reposent sur le principe de **réduction** :



Je peux aller à la boulangerie en allant au laboratoire.

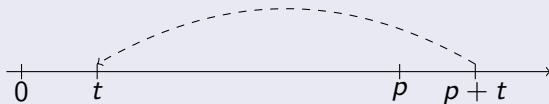
Aller à la boulangerie **est plus simple** que d'aller au laboratoire.

↪ **Résoudre** mon problème difficile **est plus facile** que de **casser** la sécurité du schéma

Notre Problème Difficile : le Logarithme Discret

$\mathbb{Z}_p$

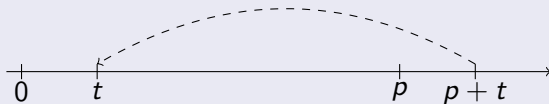
On ne va plus travailler sur les entiers, mais on va les « *rabattre* » :



Notre Problème Difficile : le Logarithme Discret

$\mathbb{Z}_p$

On ne va plus travailler sur les entiers, mais on va les « *rabattre* » :

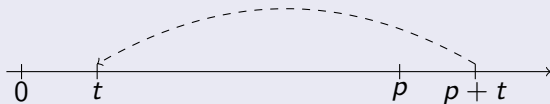


On notera $x \bmod p$ pour désigner un élément de $\mathbb{Z}_p$

Notre Problème Difficile : le Logarithme Discret

$\mathbb{Z}_p$

On ne va plus travailler sur les entiers, mais on va les « *rabattre* » :



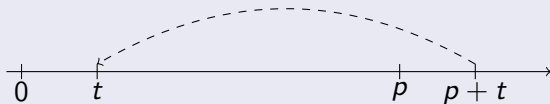
On notera $x \bmod p$ pour désigner un élément de $\mathbb{Z}_p$

Idée En faisant ça on perd la structure de *droite* des entiers.
Et on a alors que $x \mapsto a^x \bmod p$ devient dur à prédire.

Notre Problème Difficile : le Logarithme Discret

$\mathbb{Z}_p$

On ne va plus travailler sur les entiers, mais on va les « *rabattre* » :



On notera $x \bmod p$ pour désigner un élément de $\mathbb{Z}_p$

Idée En faisant ça on perd la structure de *droite* des entiers.
Et on a alors que $x \mapsto a^x \bmod p$ devient dur à prédire.

Exemple :

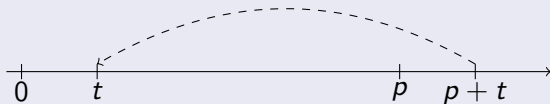
x	0	1	2	3	4	5	6
5^x	1	5	4	6	2	3	1

mod 7

Notre Problème Difficile : le Logarithme Discret

 $\mathbb{Z}_p$

On ne va plus travailler sur les entiers, mais on va les « *rabattre* » :



On notera $x \bmod p$ pour désigner un élément de $\mathbb{Z}_p$

Idée En faisant ça on perd la structure de *droite* des entiers.
Et on a alors que $x \mapsto a^x \bmod p$ devient dur à prédire.

Exemple :

x	0	1	2	3	4	5	6
5^x	1	5	4	6	2	3	1

mod 7

Par conséquent, retrouver x à partir de a^x est difficile, c'est le problème du logarithme discret

Chiffrement El Gamal

Il existe un chiffrement à partir du problème du logarithme discret :
le chiffrement d'El Gamal.

Chiffrement El Gamal

Il existe un chiffrement à partir du problème du logarithme discret :
le chiffrement d'El Gamal.

- **GenClefs** $pk = g^a$, $sk = a$ pour $a \leftarrow \mathbb{Z}_{p-1}$ et g un générateur de $\mathbb{Z}_p^*$.
- **Chiffrer** $C = (g^r, M \times pk^r) = (C_1, C_2)$
- **Déchiffrer** $M = \frac{C_2}{C_1^a}$

Chiffrement El Gamal

Il existe un chiffrement à partir du problème du logarithme discret :
le chiffrement d'El Gamal.

- **GenClefs** $pk = g^a$, $sk = a$ pour $a \leftarrow \mathbb{Z}_{p-1}$ et g un générateur de $\mathbb{Z}_p^*$.
- **Chiffrer** $C = (g^r, M \times pk^r) = (C_1, C_2)$
- **Déchiffrer** $M = \frac{C_2}{C_1^a}$

Correction :

$$\frac{C_2}{C_1^a} = \frac{M \times pk^r}{(g^r)^a} = \frac{M \times pk^r}{(g^a)^r} = \frac{M \times pk^r}{(pk)^r} = M$$

Chiffrement El Gamal

Il existe un chiffrement à partir du problème du logarithme discret :
le chiffrement d'El Gamal.

- **GenClefs** $pk = g^a$, $sk = a$ pour $a \leftarrow \mathbb{Z}_{p-1}$ et g un générateur de $\mathbb{Z}_p^*$.
- **Chiffrer** $C = (g^r, M \times pk^r) = (C_1, C_2)$
- **Déchiffrer** $M = \frac{C_2}{C_1^a}$

Correction :

$$\frac{C_2}{C_1^a} = \frac{M \times pk^r}{(g^r)^a} = \frac{M \times pk^r}{(g^a)^r} = \frac{M \times pk^r}{(pk)^r} = M$$

Sécurité : En ayant connaissance $pk = g^a$, retrouver a est difficile.

Homomorphisme

Rappel

- **GenClefs** $pk = g^a$, $sk = a$ pour $a \leftarrow \mathbb{Z}_p$ et g un élément de $\mathbb{Z}_p$
- **Chiffrer** $C = (g^r, M \times pk^r) = (C_1, C_2)$
- **Déchiffrer** $M = \frac{C_2}{C_1^a}$

Homomorphisme

Rappel

- **GenClefs** $pk = g^a$, $sk = a$ pour $a \leftarrow \mathbb{Z}_p$ et g un élément de $\mathbb{Z}_p$
- **Chiffrer** $C = (g^r, M \times pk^r) = (C_1, C_2)$
- **Déchiffrer** $M = \frac{C_2}{C_1^a}$

Si $M = g^{m'}$ pour m' pas trop grand ($<$ quelques millions), alors, étant donné deux messages $C = (C_1, C_2) = \text{Chiffrer}(g^m)$ et $C^* = (C_1^*, C_2^*) = \text{Chiffrer}(g^{m^*})$, on a

$$(C_1 \cdot C_1^*, C_2 \cdot C_2^*) = \text{Chiffrer}(g^{m+m^*})$$

Ce qui permet de compter les voix à la fin.

Plan

Introduction

Outils

Construction

Conclusion

Rappel

- **GenClefs** où on génère les clefs des votants
- **Vote** où un votant envoie son vote
- **Ouvrir** qui clôt le vote et renvoie le nombre de voix

GenClefs : On crée une pair (pk, sk) pour **El Gamal**, et on distribue à chaque votant un certificat

Rappel

- **GenClefs** où on génère les clefs des votants
- **Vote** où un votant envoie son vote
- **Ouvrir** qui clôt le vote et renvoie le nombre de voies

GenClefs : On crée une pair (pk, sk) pour **El Gamal**, et on distribue à chaque votant un certificat

Vote : Chaque votant chiffre g^0 ou g^1 par El Gamal, et envoie son vote en utilisant son certificat

Rappel

- **GenClefs** où on génère les clefs des votants
- **Vote** où un votant envoie son vote
- **Ouvrir** qui clôt le vote et renvoie le nombre de voix

GenClefs : On crée une pair (pk, sk) pour **El Gamal**, et on distribue à chaque votant un certificat

Vote : Chaque votant chiffre g^0 ou g^1 par El Gamal, et envoie son vote en utilisant son certificat

Ouvrir : On somme toutes les voix et on déchiffre g^n d'où on retrouve n

Rappel

- **GenClefs** où on génère les clefs des votants
- **Vote** où un votant envoie son vote
- **Ouvrir** qui clôt le vote et renvoie le nombre de voix

GenClefs : On crée une pair (pk, sk) pour **El Gamal**, et on distribue à chaque votant un certificat

Vote : Chaque votant chiffre g^0 ou g^1 par El Gamal, et envoie son vote en utilisant son certificat

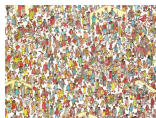
Ouvrir : On somme toutes les voix et on déchiffre g^n d'où on retrouve n

Problème Comment garantir que l'on a bien envoyé g^0 ou g^1 ?

Preuves à Divulgateur Nulle de Connaissances

Preuves ZK

Il existe un moyen de prouver que l'on sait quelque chose, sans préciser quoi.

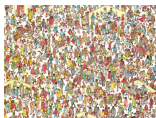


Exemple :

Preuves à Divulgation Nulle de Connaissances

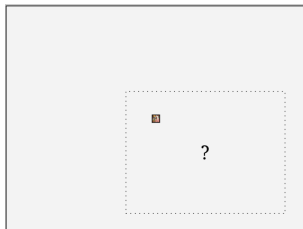
Preuves ZK

Il existe un moyen de prouver que l'on sait quelque chose, sans préciser quoi.



Exemple :

Idée : Je place un carton quatre fois plus grand que la feuille et je montre Charlie par un trou au milieu de la feuille.

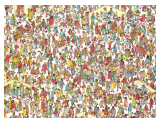


On peut utiliser cela pour prouver ce qu'on voulait.

Preuves à Divulgation Nulle de Connaissances

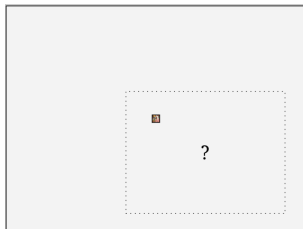
Preuves ZK

Il existe un moyen de prouver que l'on sait quelque chose, sans préciser quoi.



Exemple :

Idée : Je place un carton quatre fois plus grand que la feuille et je montre Charlie par un trou au milieu de la feuille.



On peut utiliser cela pour prouver ce qu'on voulait. (sans Charlie)

Conclusion

- Le vote électronique est une primitive qui cherche à répondre à des besoins précis
- On a vu comment construire un schéma de vote en utilisant des constructions plus élémentaires
- La cryptographie a encore bien d'autres constructions de ce type là à développer
- Il existe encore d'autres domaines de la cryptographie à développer : le logarithme discret ne résistera pas à un ordinateur quantique. . .

Conclusion

- Le vote électronique est une primitive qui cherche à répondre à des besoins précis
- On a vu comment construire un schéma de vote en utilisant des constructions plus élémentaires
- La cryptographie a encore bien d'autres constructions de ce type là à développer
- Il existe encore d'autres domaines de la cryptographie à développer : le logarithme discret ne résistera pas à un ordinateur quantique. . .

Merci de votre attention.
Des questions ?