Sécurité des mécanismes cryptographiques

(source : D. Boneh)

Chiffrement symétrique

Définition

Une paire d'algorithmes (E, D) efficaces définis sur $(\mathcal{K}, \mathcal{M}, \mathcal{C})$ où
$E : \mathcal{K} \times \mathcal{M} \to \mathcal{C} \quad \mathcal{D} : \mathcal{K} \times \mathcal{C} \to \mathcal{M}$
tq $\forall m \in \mathcal{M},\ k \in \mathcal{K},$

$$D(k, E(k, m)) = m$$

E est (souvent) randomisée D est déterministe

One time pad (Vernam 1917)

$\mathcal{M} = \mathcal{C} = \mathcal{K} = \{0,1\}^n$

$$c = E(k,m) = k \oplus m \qquad\qquad D(k,c) = k \oplus c$$

Très rapide mais pas pratique car la clé doit être aussi longue que le message

Comment prouver sa sécurité ?

Idée : le chiffré ne doit révéler aucune information sur le texte clair

Notion de sécurité parfaite...

Sécurité parfaite (Perfect secrecy)

Définition

(E, D) sur $(\mathcal{K}, \mathcal{M}, \mathcal{C})$ a une sécurité parfaite si
$\forall m_0, m_1 \in \mathcal{M}, \quad |m_0| = |m_1|, \quad \forall c \in \mathcal{C}$

$$Pr_k[E(k, m_0) = c] = Pr_k[E(k, m_1) = c]$$

où $k \xleftarrow{R} \mathcal{K}$ (k est choisi aléatoirement dans $\mathcal{K}$)

- c donné, impossible de savoir s'il est le chiffré de m_0 ou de m_1
- L'adversaire n'apprend rien du chiffré
- Donc une attaque utilisant seulement le chiffré n'est pas possible

Exercice

Soit m, c, combien de clés permettent de chiffrer m en c par OTP ?

- dépend de m
- une infinité
- aucune
- 1
- 2

Que vaut $\#\mathcal{K}$?

$\forall m \in \mathcal{M}, \quad \forall c \in \mathcal{C}$
Que vaut : $\#\{cles \;\; k \;\; tq \;\; E(k, m) = c\}$?

OTP a une sécurité parfaite

Preuve

$\forall m \in \mathcal{M}, \quad \forall c \in \mathcal{C}$

$$Pr_k[E(k,m) = c] = \frac{\#\{cles \;\; k \;\; tq \;\; E(k,m) = c\}}{|\mathcal{K}|} = 1/2^n$$

Avec OTP, une attaque par chiffrés n'est pas possible
Mais d'autres attaques sont possibles ...
Peut-on garder une sécurité parfaite si on réduit la taille des clés ?

Théorème

Sécu parfaite $\Rightarrow \quad |\mathcal{K}| = |\mathcal{M}|$

OTP non pratique ! !

Stream cipher : rendre pratique OTP

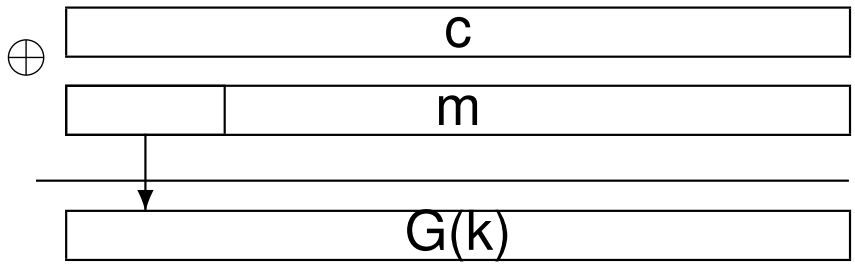
Les Stream Ciphers ne peuvent pas avoir de sécurité parfaite

- Besoin d'une autre définition de sécurité
- La sécurité dépendra du PRG
- Le PRG doit être imprédictible

Supposons PRG prédictible $\quad k =$seed, $G =$PRG

$$\exists i \quad G(k)|_{1,\ldots,i} \xrightarrow{Algo} G(k)|_{i+1,\ldots,n}$$

Alors



Le fait de pouvoir prédire le prochain bit est un problème !

PRG prédictible ou imprédictible

Définition : PRG prédictible

$G : \mathcal{K} \to \{0,1\}^n$ est prédictible si $\exists \mathcal{A}$ Algo efficace et $\exists 1 \leq i \leq n-1$ tq

$$Pr_{k \xleftarrow{R} \mathcal{K}}[\mathcal{A}(G(k))|_{1,\ldots,i} = G(k)|_{i+1}] \geq 1/2 + \epsilon$$

pour ϵ "non négligeable" c.a.d. $\epsilon \geq 1/2^{30}$

Définition : un PRG est imprédictible s'il n'est pas prédictible

$\Rightarrow \forall i$, aucun adversaire ne peut prédire le prochain bit pour un ϵ non négligeable

Exercice : $G : \mathcal{K} \to \{0,1\}^n$ tq $\forall k \in \mathcal{K}, XOR(G(k)) = 1$. G est-il prédictible ?

1. Oui, le 1er bit donné, je peux prédire le 2eme
2. G est imprédictible
3. Ca dépend de n
4. Oui car si je connais les $n-1$ premiers bits, je peux prédire le nième

Que veut dire négligeable ?

En pratique ϵ est un scalaire

- ϵ non négligeable : $\epsilon \geq 1/2^{30}$ (l'événement peut survenir sur 1GB de données)
- ϵ négligeable : $\epsilon \leq 1/2^{88}$ (l'événement n'arrivera jamais dans toute la vie de la clé)

En théorie ϵ est une fonction

$$\epsilon \ : \ \mathbb{Z}^{\geq 0} \rightarrow \mathbb{R}^{\geq 0}$$

- ϵ non négligeable : $\exists d \ : \ \epsilon(\lambda) \geq 1/\lambda^d$ ($\epsilon \geq 1/Poly$, pour beaucoup de λ)
- ϵ négligeable : $\forall d, \ \exists \lambda_d, \ \lambda \geq \lambda_d \ : \ \epsilon(\lambda) \leq 1/\lambda^d$ ($\epsilon \leq 1/Poly$, pour λ grand)

Exercice : négligeable ou non négligeable ?

- $\epsilon(\lambda) = 1/2^{\lambda}$
- $\epsilon(\lambda) = 1/\lambda^{1000}$
- $\epsilon(\lambda) = \begin{cases} 1/2^{\lambda} \text{ si } \lambda \text{ impair} \\ 1/\lambda^{1000} \text{ si } \lambda \text{ pair} \end{cases}$
- $\epsilon(\lambda) = 1/2^{\lambda} + 1/\lambda^{1000}$
- $\epsilon(\lambda) = 1/1000^{\lambda}$

PRG sûr

Un bon PRG doit se comporter (presque) comme un générateur aléatoire (RG)
Qu'est-ce que cela signifie ?

Soit $G : \mathcal{K} \to \{0,1\}^n$ un PRG,

$$[k \xleftarrow{R} \mathcal{K}, \quad output\ G(k)]$$

doit être "indistinguable" de

$$[r \xleftarrow{R} \{0,1\}^n, \quad output\ r]$$

Remarque : L'espace des outputs de $G()$ est beaucoup plus petit que $\{0,1\}^n$

Tests statistiques (voir NIST)

Un test sur $\{0,1\}^n$ est un algo (distingueur)

$$\{0,1\}^n \to \begin{cases} 0 \text{ l'output n'est pas aléatoire} \\ 1 \text{ le test est passé avec succès} \end{cases}$$

Exemples de tests :

- nombre de 1 dans la séquence
- nombre de runs
- longueur du plus grand run de 1
- ...

Avantage

Soit $G : \mathcal{K} \rightarrow \{0,1\}^n$ un PRG, A un test stat. sur $\{0,1\}^n$

Définition

$$Adv_{PRG}[A, G] := |Pr_{k \xleftarrow{R} \mathcal{K}}[A(G(k)) = 1] - Pr_{r \xleftarrow{R} \{0,1\}^n}[A(r) = 1]|$$

- L'avantage donne une valeur entre 0 et 1
- Adv proche de 1 $\Rightarrow$ A peut distinguer G d'un RG
- Adv proche de 0 $\Rightarrow$ A ne peut pas distinguer G d'un RG

Exemple

Soit $G : \mathcal{K} \to \{0,1\}^n$ un PRG, A un test stat. sur $\{0,1\}^n$
G satisfait $msb(G(k)) = 1$ pour $2/3$ des clés de $\mathcal{K}$

Définissons le test stat A par :
$A(x) = 1$ si $msb(x) = 1$
$A(x) = 0$ si $msb(x) = 0$

Quel est l'avantage de A ?

$$Adv_{PRG}[A, G] := |Pr_{k \xleftarrow{R} \mathcal{K}}[A(G(k)) = 1] - Pr_{r \xleftarrow{R} \{0,1\}^n}[A(r) = 1]| = ?$$

Exemple

Soit $G : \mathcal{K} \to \{0,1\}^n$ un PRG, A un test stat. sur $\{0,1\}^n$
G satisfait $msb(G(k)) = 1$ pour $2/3$ des clés de $\mathcal{K}$

Définissons le test stat A par :
$A(x) = 1$ si $msb(x) = 1$
$A(x) = 0$ si $msb(x) = 0$

Quel est l'avantage de A ?

$$Adv_{PRG}[A, G] := |Pr_{k \xleftarrow{R} \mathcal{K}}[A(G(k)) = 1] - Pr_{r \xleftarrow{R} \{0,1\}^n}[A(r) = 1]| = 1/6$$

L'avantage n'est pas négligeable donc A casse G avec avantage $1/6$

PRG sûr

Définition : PRG sûr

$G : \mathcal{K} \to \{0,1\}^n$ est sûr si pour tout test stat A, $Adv_{PRG}[A, G]$ est négligeable.

Existe-t-il des PRG dont la sécurité est prouvable ? on ne sait pas (P= ?NP)

Un PRG sûr est imprédictible

Preuve : par contraposé

On montre que PRG prédictible $\Rightarrow$ PRG non sûr

Un PRG sûr est imprédictible : preuve

Soit A un algo efficient tq

$$Pr_{k \xleftarrow{R} \mathcal{K}}[A(G(k))|_{1,\ldots,i} = G(k)|_{i+1}] \geq 1/2 + \epsilon$$

pour ϵ "non négligeable" (par exemple $\epsilon = 1/1000$)
Définissons un test stat B :

$$B(x) = \begin{cases} \text{if } A(x)|_{1,\ldots,i} = x_{i+1} \text{ output 1} \\ \text{else output 0} \end{cases}$$

$$r \xleftarrow{R} \{0,1\}^n : \qquad Pr[B(r) = 1] = 1/2$$

$$k \xleftarrow{R} \mathcal{K} : \qquad Pr[B(G(k) = 1)] = 1/2 + \epsilon$$

$$\Rightarrow Adv_{PRG}[B, G] = \epsilon$$

avec ϵ non négligeable

Yao'82 : Un PRG imprédictible est sûr

Théorème

Soit $G : \mathcal{K} \to \{0,1\}^n$ un PRG. Si $\forall i \in \{0,\ldots,n-1\}$, G est imprédictible à la position i, alors G est sûr.

Cela signifie que si les prédicateurs du prochain bit ne peuvent pas distinguer G d'un RG, alors aucun test statistique ne peut le faire.

Exemple

Soit $G : \mathcal{K} \to \{0,1\}^n$ un PRG tq à partir des derniers $n/2$ bits de $G(k)$, il est facile de calculer les $n/2$ premiers bits

G est-il prédictible pour certains $i \in \{0, \ldots, n-1\}$?

Indistinguabilité (calculatoire)

Soit P_1 et P_2 deux distributions sur $\{0,1\}^n$

Définition

On dit que P_1 et P_2 sont calculatoirement indistinguable ($P_1 \approx_p P_2$) si pour tout test statistique A

$$|Pr_{k \xleftarrow{R} P_1}[A(x) = 1] - Pr_{r \xleftarrow{R} P_2}[A(x) = 1]| < negl.$$

Ex : un PRG est sûr si $\{k \xleftarrow{R} \mathcal{K} \ : \ G(k)\} \approx_p uniform(\{0,1\}^n)$

Sécurité sémantique

Qu'est-ce qu'un chiffrement sûr ?

puissance de l'attaquant (pour l'instant) : il connait le chiffré

Possibles exigences de sécurité :

- l'attaquant ne peut pas retrouver la clé
  Exemple : $E(k, m) = m$ le chiffrement n'est pas sûr et pourtant on ne peut pas retrouver la clé
- l'attaquant ne peut retrouver le clair en entier
  Exemple : $E(k, m_0 \| m_1) = m_0 \| E(k, m_1)$
- Shannon : le chiffré ne doit donner aucune info sur le clair
  $H(m|c) = H(m)$ (H est l'entropie = degré d'incertitude)

Sécurité sémantique (suite)

Soit (E, D) un syst. de chiffrement sur $(\mathcal{K}, \mathcal{M}, \mathcal{C})$
Au lieu de considérer la définition

Définition

(E, D) sur $(\mathcal{K}, \mathcal{M}, \mathcal{C})$ a une sécurité parfaite si $\forall m_0, m_1 \in \mathcal{M}, \quad |m_0| = |m_1|$

$$\{E(k, m_0)\} \;=\; \{E(k, m_1)\} \text{ où } k \xleftarrow{R} \mathcal{K}$$

on préfère la définition

Définition

(E, D) sur $(\mathcal{K}, \mathcal{M}, \mathcal{C})$ a une sécurité parfaite si $\forall m_0, m_1 \in \mathcal{M}, \quad |m_0| = |m_1|$

$$\{E(k, m_0)\} \;\approx_p\; \{E(k, m_1)\} \text{ où } k \xleftarrow{R} \mathcal{K}$$

ET l'adversaire doit choisir explicitement m_0 et m_1

Sécurité sémantique (one time key)

$\mathbb{E} = (E, D)$
Un adversaire (attaquant) A, un challenger Chal.
Chal. choisit une clé aléatoire dans $\mathcal{K}$
A choisit m_0 et m_1 dans $\mathcal{M}$ de même taille
Chal. choisit b au hasard dans $\{0, 1\}$ et envoie le chiffré de $m_b = c$ à A
A doit deviner $b \in \{0, 1\}$
Pour $b = 0, 1$ on définit 2 expérimentations

- $EXP(0)$: le challenger a chiffré m_0
- $EXP(1)$: le challenger a chiffré m_1

W_b = événement tq $EXP(b) = 1$ (Chal. a chiffré m_b et A répond 1)

$$Adv_{SS}[A, \mathbb{E}] := |Pr[W_0] - Pr[W_1]| \in [0, 1]$$

Sécurité sémantique

Définition

$\mathbb{E} = (E, D)$ est sémantiquement sûr si pour tout algo A efficient,

$$Adv_{SS}[A, \mathbb{E}]$$

est négligeable

$\Rightarrow$ les distributions des chiffrés $\{E(k, m_0)\}$ et $\{E(k, m_1)\}$ sont indistinguables

Exemple

Soit un algo A efficient qui peut toujours déduire LSB du clair à partir du chiffré
Montrer que $\mathbb{E}$ n'est pas sémantiquement sûr

Preuve

Challenger
$Adv.\ B$
$EXP(0)$, $EXP(1)$
$Pr[EXP(0) = 1]$?, $Pr[EXP(1) = 1]$?

$Adv_{SS}[B, \mathbb{E}]$?

OTP est sémantiquement sûr

Preuve

Challenger
Adv. A
$Pr[A(k \oplus m_0) = 1]$?, $Pr[A(k \oplus m_1) = 1]$?
Rappel : les distributions de $\{k \oplus m_0\}$ et de $\{k \oplus m_1\}$ sont identiques

$Adv_{SS}[A, \mathbb{E}]$?

OTP est sémantiquement sûr contre n'importe quel attaquant car les distributions des chiffrés sont égales (pas possible de les distinguer)

Sécurité lorsque la clé est réutilisée

Exemple : Système de fichiers : plusieurs fichiers chiffrés par la même clé AES
$\rightarrow$ l'adversaire peut obtenir plusieurs chiffrés d'une même clé

Puissance de l'attaquant : chosen-plaintext attack (CPA)
il peut obtenir le chiffré de n'importes quels clairs
But de l'adversaire : casser la sécurité sémantique

Le jeu est identique au précédent mais l'attaquant peut répéter le jeu plusieurs fois

CPA $\Rightarrow$ si l'attaquant veut connaitre m tq $c = E(k, m)$, requête avec $m_0 = m_1$

$\mathbb{E}$ est sémantiquement sûr sous CPA

si pour tout algo A

$$Adv_{CPA}[A, \mathbb{E}] = |Pr[EXP(0) = 1] - Pr[EXP(1) = 1]|$$

est négligeable

Sous CPA, un chiffrement déterministe n'est pas sûr

Preuve en exercice

Chiffrement authentifié

Définition

Un système de chiffrement authentifié (E, D) est tq

$$E : \mathcal{K} \times \mathcal{M} \times N \to \mathcal{C}$$

$$D : \mathcal{K} \times \mathcal{C} \times N \to \mathcal{M} \cup \{\perp\}$$

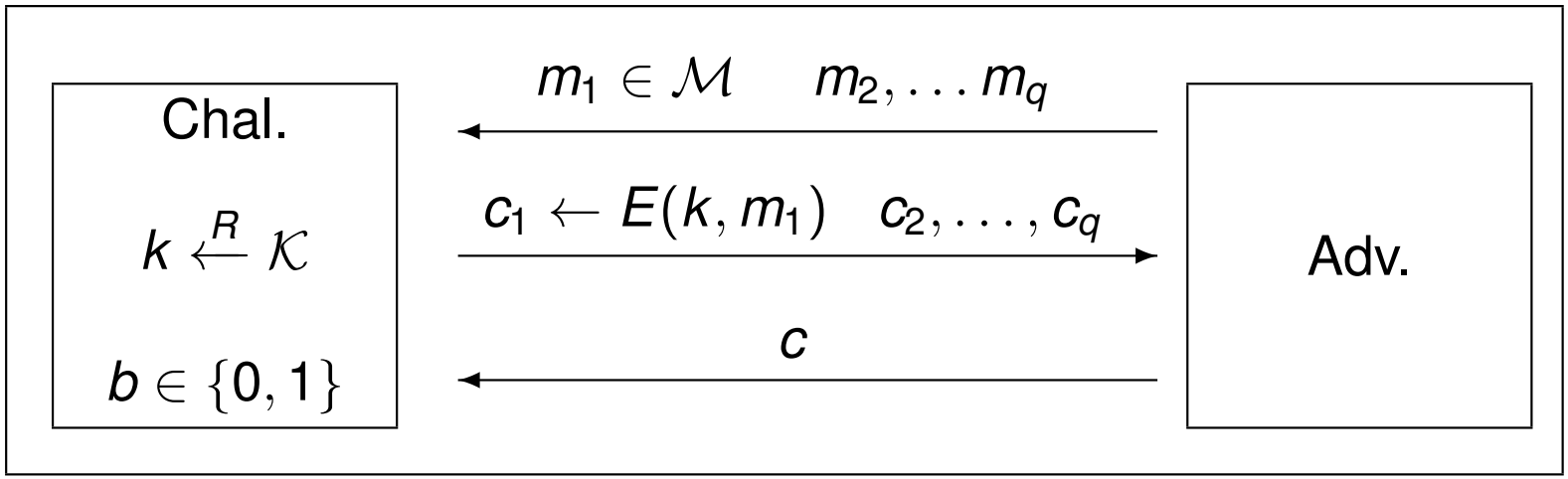
où N est l'ensemble des nonces pour un chiffrement non déterministe
($\perp$ = chiffré rejeté)

Sécurité : le système doit assurer

- la sécurité sémantique sous CPA
- l'intégrité du chiffré (l'attaquant ne peut pas créer de nouveaux chiffrés qui permettent le déchiffrement)

Intégrité du chiffré (ciphertext integrity)

Soit (E, D) un système de chiffrement et $\mathcal{M}$ l'espace des messages



$b = 1$ si $D(k, c) \neq \perp$ et $c \notin \{c_1, \ldots, c_q\}$
$b = 0$ sinon

(E, D) a "l'intégrité du chiffré" si pour tout algo efficace A

$$Adv_{CI}[A, E] = Pr[Chal.\ outputs\ 1] \text{ est négligeable}$$

Chiffrement authentifié

Définition

Un système de chiffrement (E, D) assure le chiffrement authentifié (AE) si

1. il est sémantiquement sûr sous CPA
2. il a la propriété de l'intégrité du chiffré (ciphertext integrity)

Exemple : CBC avec random IV assure-t-il AE ?

- AE $\Rightarrow$ authenticité (mais attaques par rejeu possibles)

Attaques par chiffrés choisis (CCA)

Dans certaines situations, l'adversaire arrive à connaitre le clair de certains chiffrés. Cela peut l'aider à décrypter son message

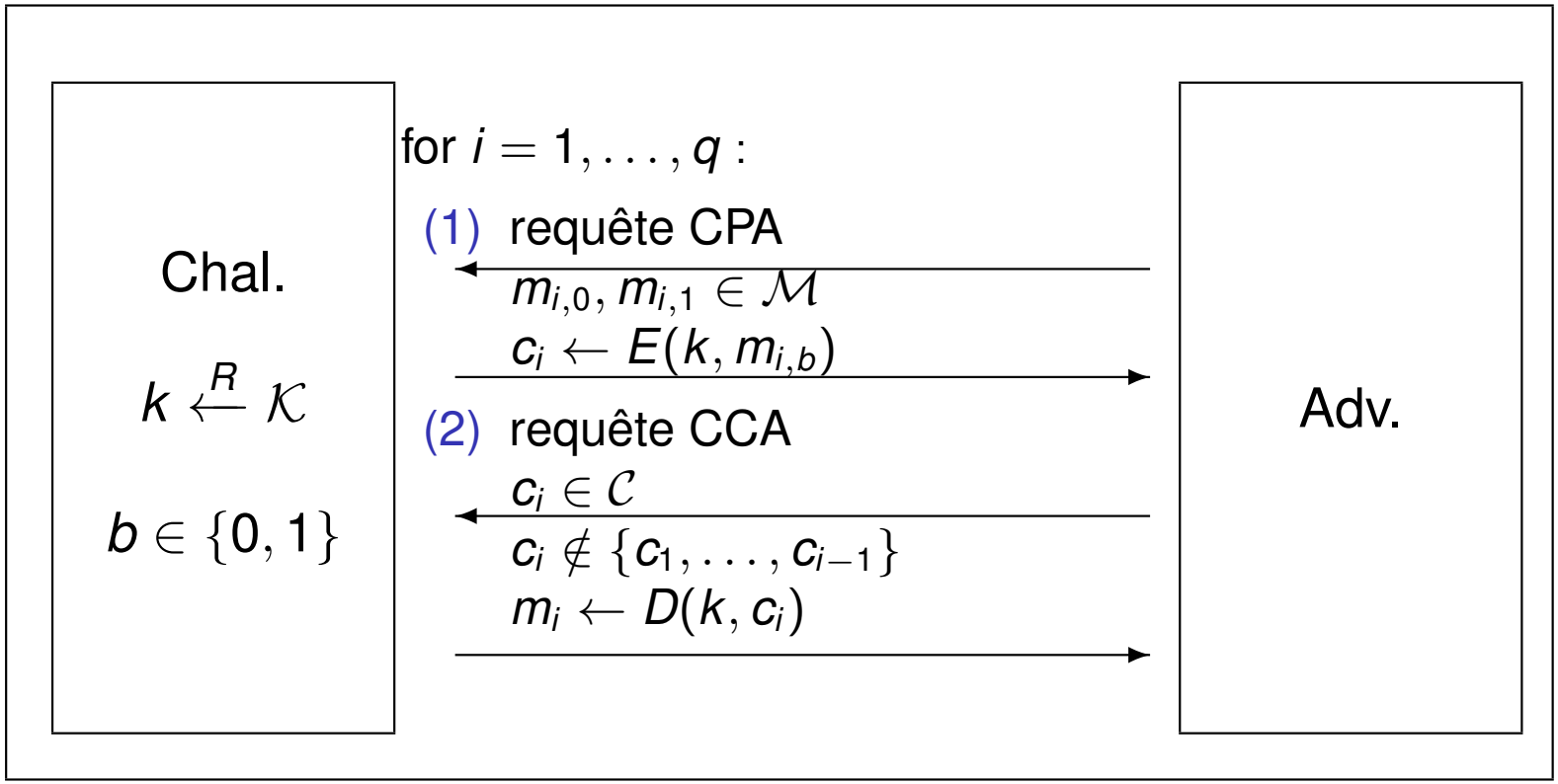
Puissance de l'adversaire : CPA et CCA

- Obtenir le chiffrement des messages de son choix
- Obtenir le déchiffrement des chiffrés de son choix (autre que le challenge)

But de l'adversaire : casser la sécurité sémantique

Modèle de sécurité : Chosen ciphertext security

Soit $\mathbb{E} = (E, D)$ un syst. de chiffrement sur $(\mathcal{K}, \mathcal{M}, \mathcal{C})$



Chosen ciphertext security

$\mathbb{E}$ est sûr si pour tout algo A efficient :

$Adv_{CPA}[A, \mathbb{E}] = |Pr[EXP(0) = 1] - Pr[EXP(1) = 1]|$ est négligeable

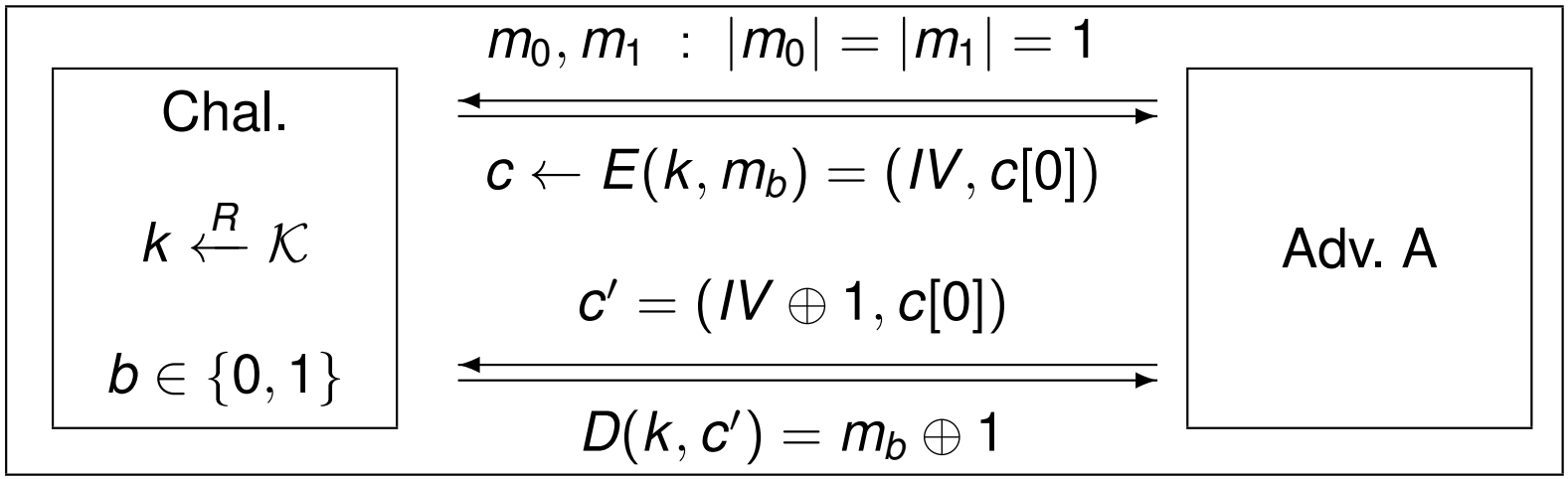
Exemple : CBC avec IV aléatoire n'est pas CCA-sûr

Chosen ciphertext security

$\mathbb{E}$ est sûr si pour tout algo A efficient :

$Adv_{CPA}[A, \mathbb{E}] = |Pr[EXP(0) = 1] - Pr[EXP(1) = 1]|$ est négligeable

Exemple : CBC avec IV aléatoire n'est pas CCA-sûr



Chiffrement authentifié $\Rightarrow$ CCA-sûr

Théorème

Soit (E, D) un système de chiffrement qui assure AE, alors (E, D) est CCA-sûr

AE assure la confidentialité contre des adversaires qui peuvent décrypter des chiffrés
Mais inefficace contre les attaques par rejeu

Sécurité pour le chiffrement à clé publique

Sécurité contre l'espionnage
Exemple : Alice génère une paire de clés, envoie à Bob sa clé publique et Bob utilise cette clé pour envoyer un message chiffré à Alice

Définition

Un système de chiffrement à clés publiques est un triplet d'algos (G, E, D)

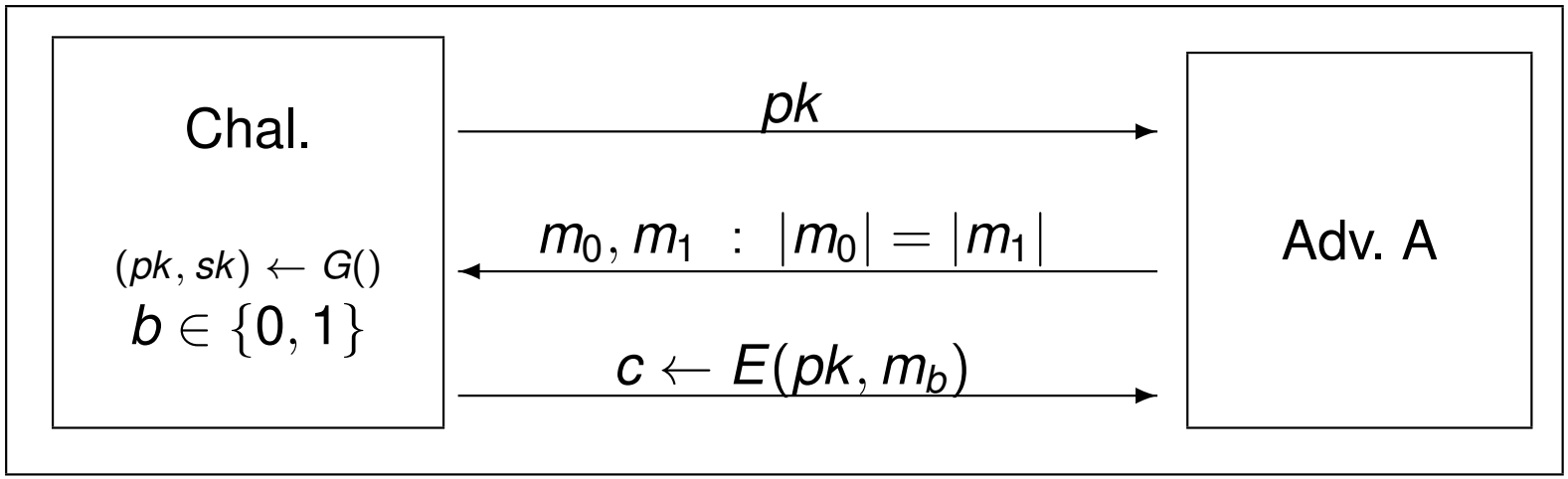
- $G()$: Algo randomisé, outputs (pk, sk)
- $E(pk, m)$: algo randomisé, outputs $c \in \mathcal{C}$
- $D(sk, m)$: algo déterministe outputs $m \in \mathcal{M}$ ou $\perp$

Consistance : pour toute paire de clés et tout message

$$D(sk, E(pk, m)) = m$$

Sécurité pour l'espionnage

Pour $b = 0, 1$ définissons $EXP(0)$ et $EXP(1)$



$\mathbb{E} = (G, E, D)$ est sûr sémantiquement (IND-CPA) si pour tout algo A efficient :

$Adv_{CPA}[A, \mathbb{E}] = |Pr[EXP(0) = 1] - Pr[EXP(1) = 1]|$ est négligeable

Relation avec le chiffrement symétrique

Chiffrement symétrique

2 notions de sécurité : one-time security et CPA

One time security $\not\Rightarrow$ CPA

Chiffrement asymétrique

l'attaquant peut chiffrer par lui-même puisqu'il connait la clé de chiffrement

One time security $\Rightarrow$ CPA

Sécurité contre des attaques actives

Un adversaire peut modifier un chiffré

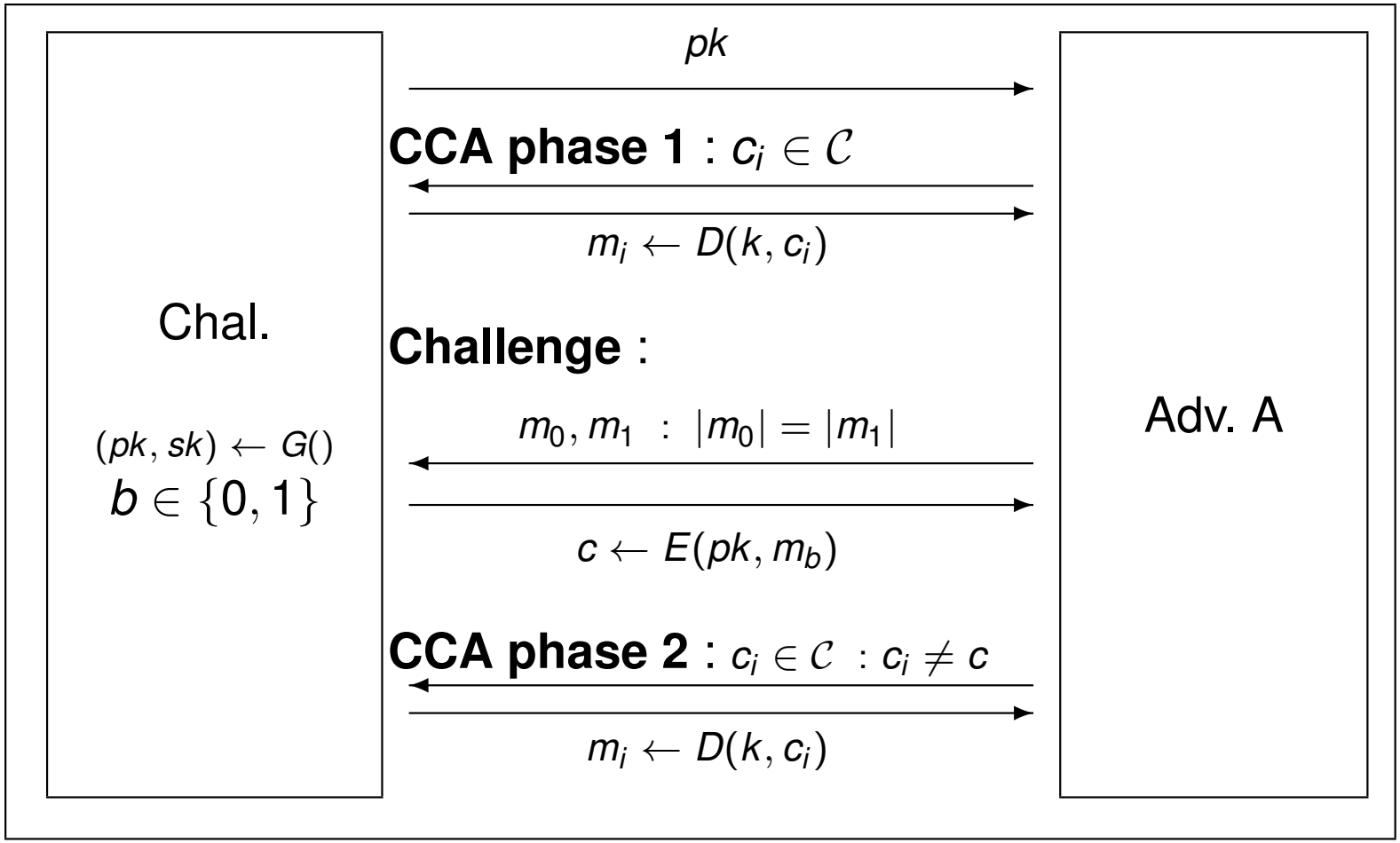
Exemple : il modifie l'entête d'un mail chiffré pour changer le nom du destinataire

Nouveau modèle

L'attaquant peut demander le clair de certains chiffrés (autre que le challenge)

Chosen ciphertext security

Soit $\mathbb{E} = (G, E, D)$ un syst de chiffrement à clés publiques. Pour $b = 0, 1$, on définit $EXP(b)$:

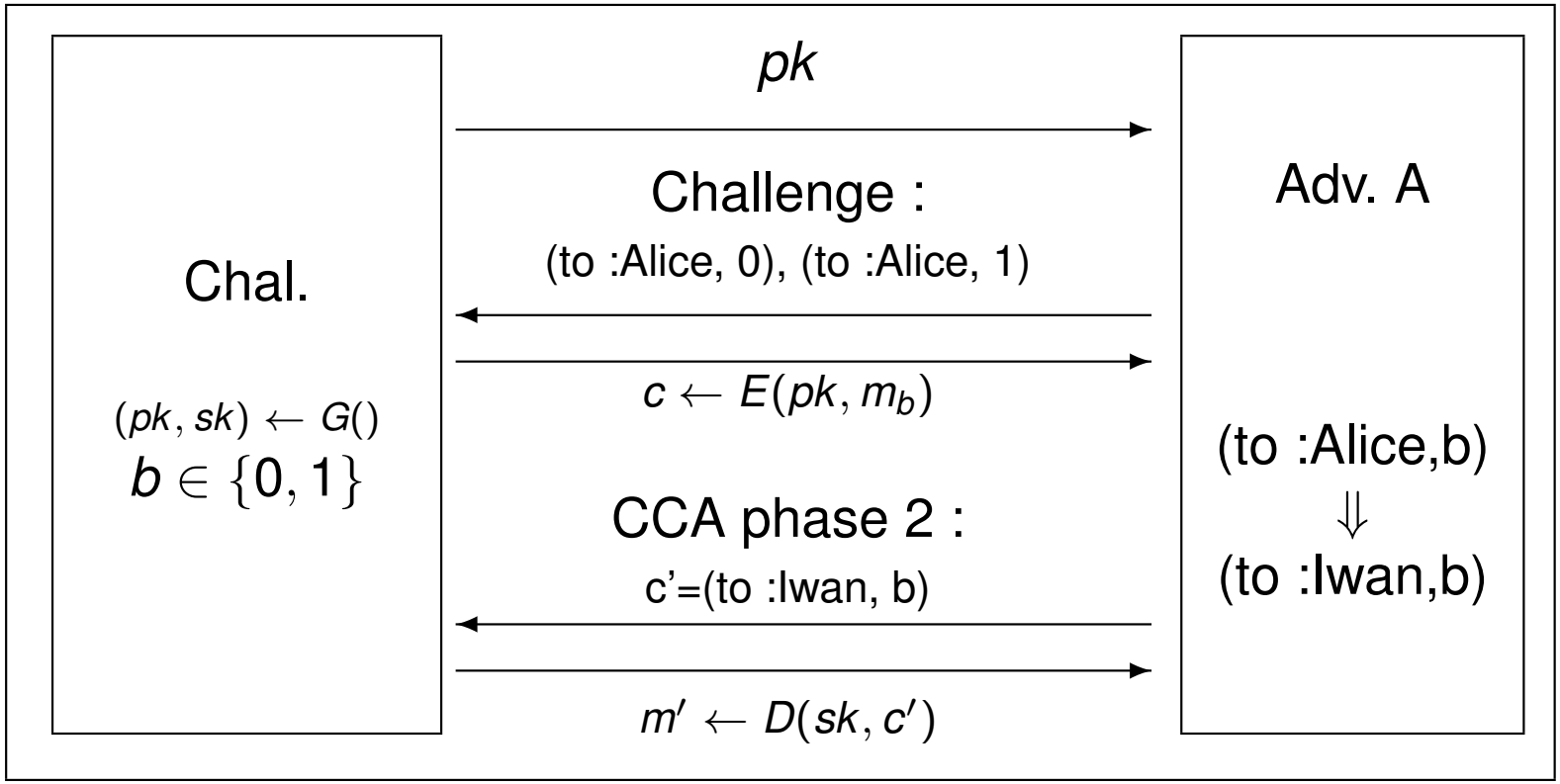


$\mathbb{E} = (G, E, D)$ est sûr sémantiquement (IND-CCA) si pour tout algo A efficient :

$Adv_{CCA}[A, \mathbb{E}] = |Pr[EXP(0) = 1] - Pr[EXP(1) = 1]|$ est négligeable

Exemple : On considère un système de chiffrement $\mathbb{E}$. Supposons qu'il existe un algo A qui permette de modifier le chiffré de (to :Alice , Body) en le chiffré de (to :Iwan , body).

$\mathbb{E}$ est-il CCA-sûr ?



L'adversaire retrouve b !