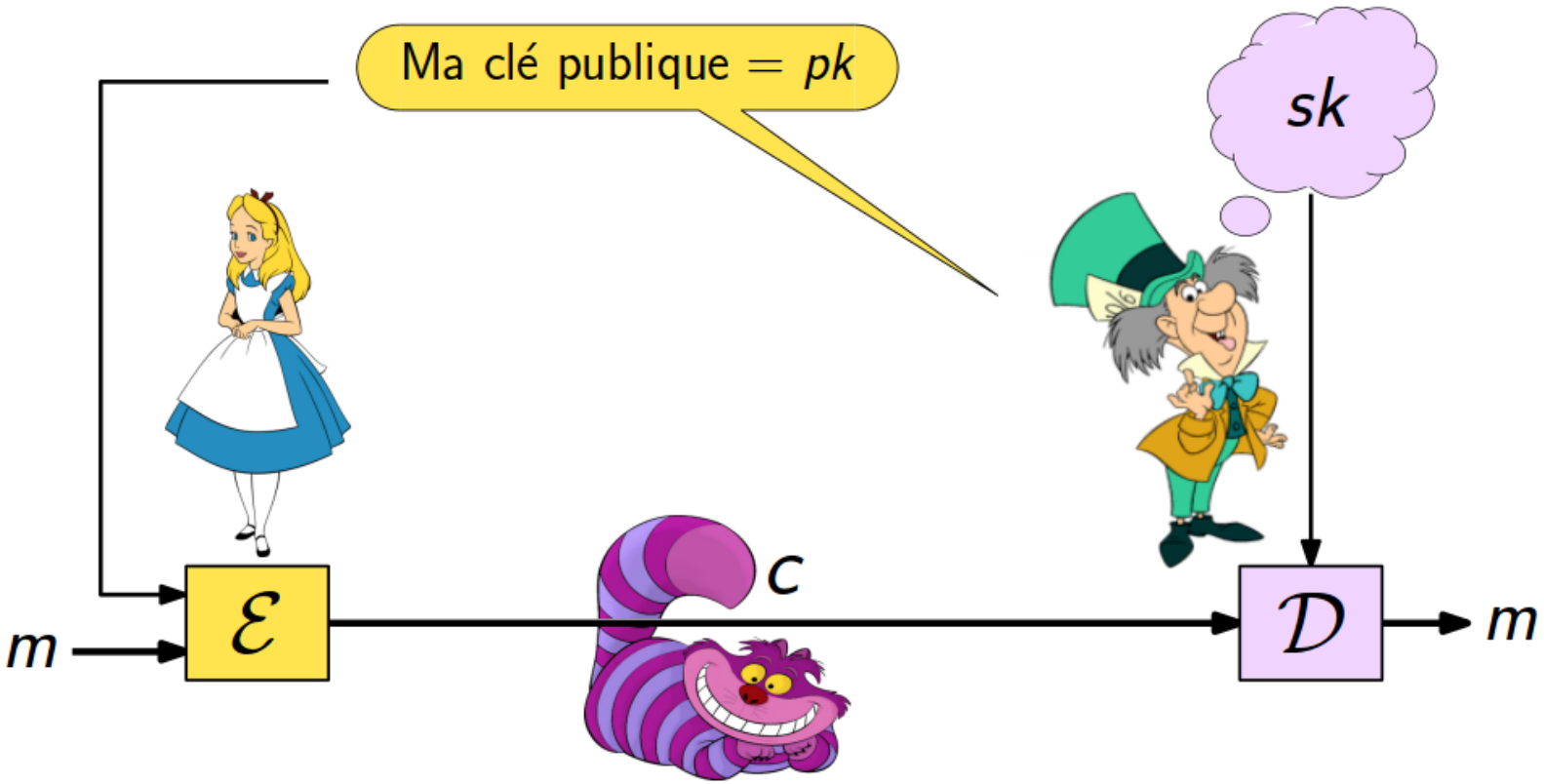
Notions de sécurité – Preuves par réduction

Anca Nitulescu
anca.nitulescu@ens.fr

Ecole Normale Supérieure, Paris

Cours 5

Sécurité des cryptosystèmes



Analiser la sécurité

Sécurité PARFAITE ?
Au sens de Shannon (théorie de l'information), la sécurité parfaite est impossible en clé publique !

Pourquoi ?

La connaissance de la clé publique et du chiffré biaise la distribution de probabilité du message clair

Quel espoir ?

Un attaquant avec une capacité de calcul polynomiale peut-il exploiter ce biais et accéder à une information ?

Prouver la sécurité

Modèle de sécurité

Pour analyser la sécurité d'un cryptosystème :

1. spécifier les buts de l'attaquant
2. spécifier ses moyens : calculs, accès aux ressources . . .
3. examiner quelles sont les chances pour un attaquant d'atteindre son but avec les moyens spécifiés
   - probabilité de succès
   - "preuve" de sécurité
4. conclure (pertinence du modèle, choix des paramètres, . . .)

L'adversaire

L'attaquant

- le plus **intelligent** possible
  - il peut faire toutes les opérations qu'il souhaite
- il dispose d'un **temps limité**
  - on ne considère pas les attaques faisables en 2^{60} ans
  - force brute : énumérer toutes les clés – temps $2^{\text{taille(clés)}}$

Modéliser l'adversaire

L'algorithme de l'attaquant

Adversaire modélisé par une **machine de Turing** :

- **probabiliste :**
  - il génére des clés
  - il tire au sort certaines étapes de son comportement
- **polynomiale** en la taille des clés :
  - il a un temps *raisonnable* d'exécution

Représenter l'attaquant

L'attaquant $\mathcal{A}$ est un algorithme (machine de Turing) probabiliste et polynomiale

Cryptographie

" $\forall \mathcal{A}$ securité"
- l'attaquant est une boîte noire
(on ignore son code)

Cryptanalyse

"$\exists \mathcal{A}$ victorieux"
- on cherche à exhiber un attaquant
(on construit son code)

Preuves de sécurité

Preuve par réduction

Principe

1. **Hypothèse** :
   Un problème P diffcile = il n'y a pas d'algorithme polynomial
   P= RSA, DLOG, DDH, CDH...
2. **Réduction** :
   - si $\mathcal{A}$ un adversaire (polynomial) casse le schéma de chiffrement,
   - alors $\mathcal{A}$ peut être utilisé pour résoudre P en temps polynomial (ce qui est considéré impossible)
3. **Résultat de sécurité** : il n'existe pas d'adversaire polynomial

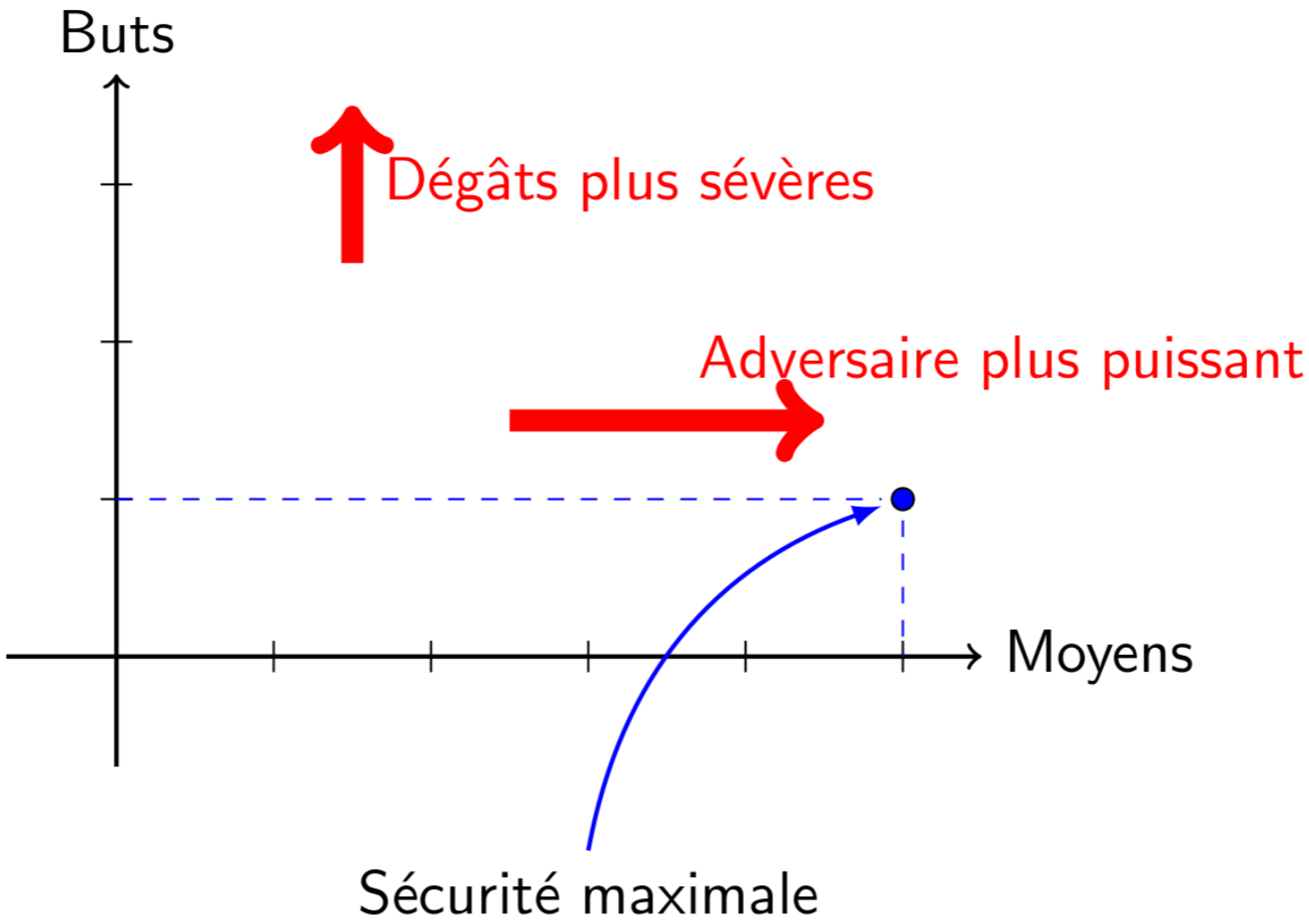
Modèle de sécurité

Deux axes d'analyse : Buts – Moyens

Définir la sécurité d'un algorithme de chiffrement :

- Quels sont les buts de l'adversaire ?
- Quels sont les moyens de l'attaquant ?

Deux axes d'analyse



Buts

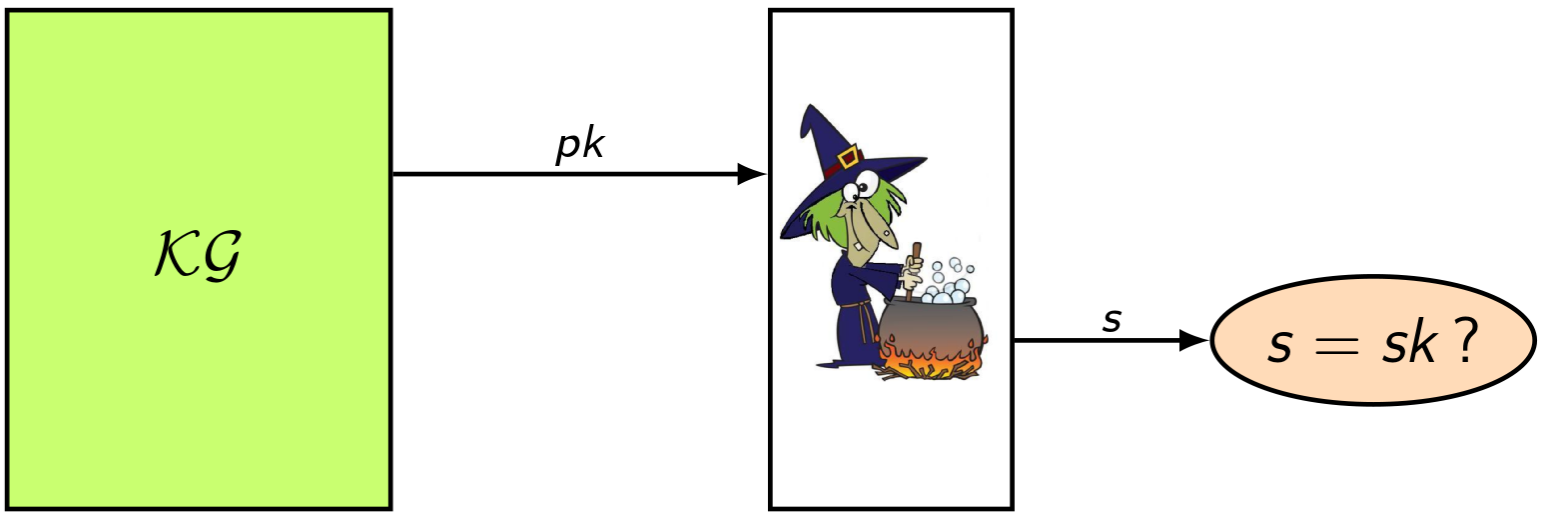
Buts de l'attaquant

1. **BRK – Cassage total** : retrouver la clé de déchiffrement (equivalent à déchiffrer tout message)
2. **OW – Sens unique** : déchiffrer un message "challenge"
3. **IND – Indistinguabilité** : distinguer entre deux chiffrés
   - **Sécurité sémantique** : à partir d'un chiffré, extraire d'information sur le message correspondant
4. **NM – Non malléabilité** : modifier un chiffré pour obtenir un autre chiffré tel que les messages correspondants soient reliés

Rétrouver la clé secrète

BRK – Break the system

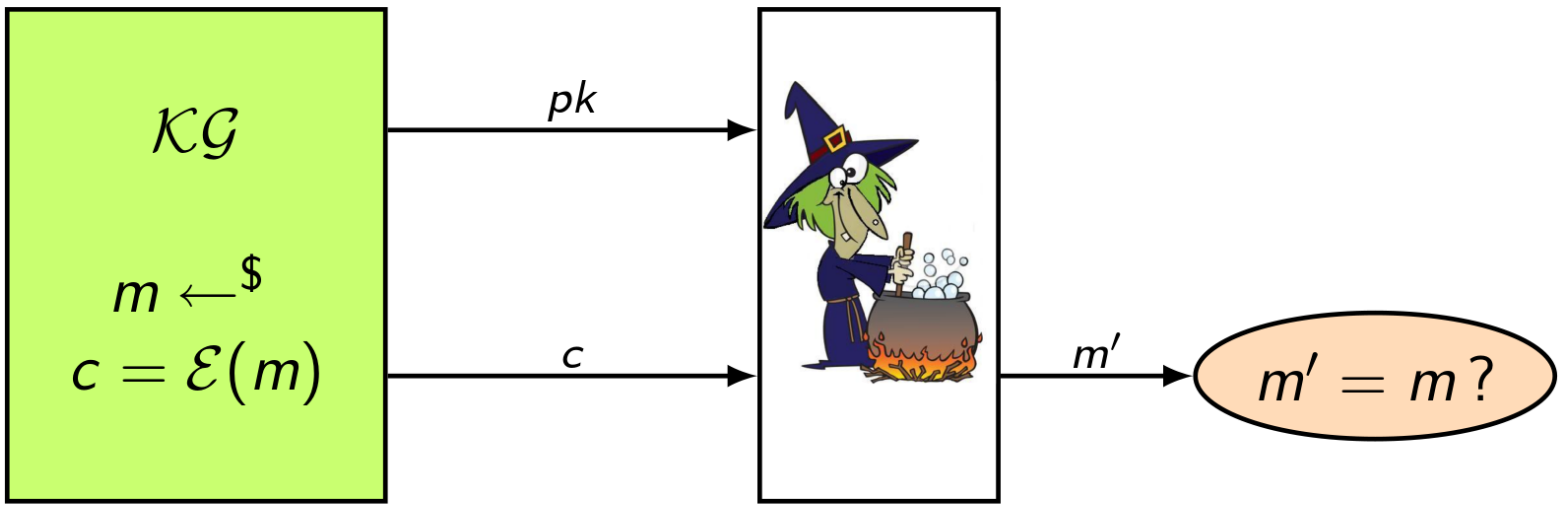
« Cassage total »



Inverser le chiffrement

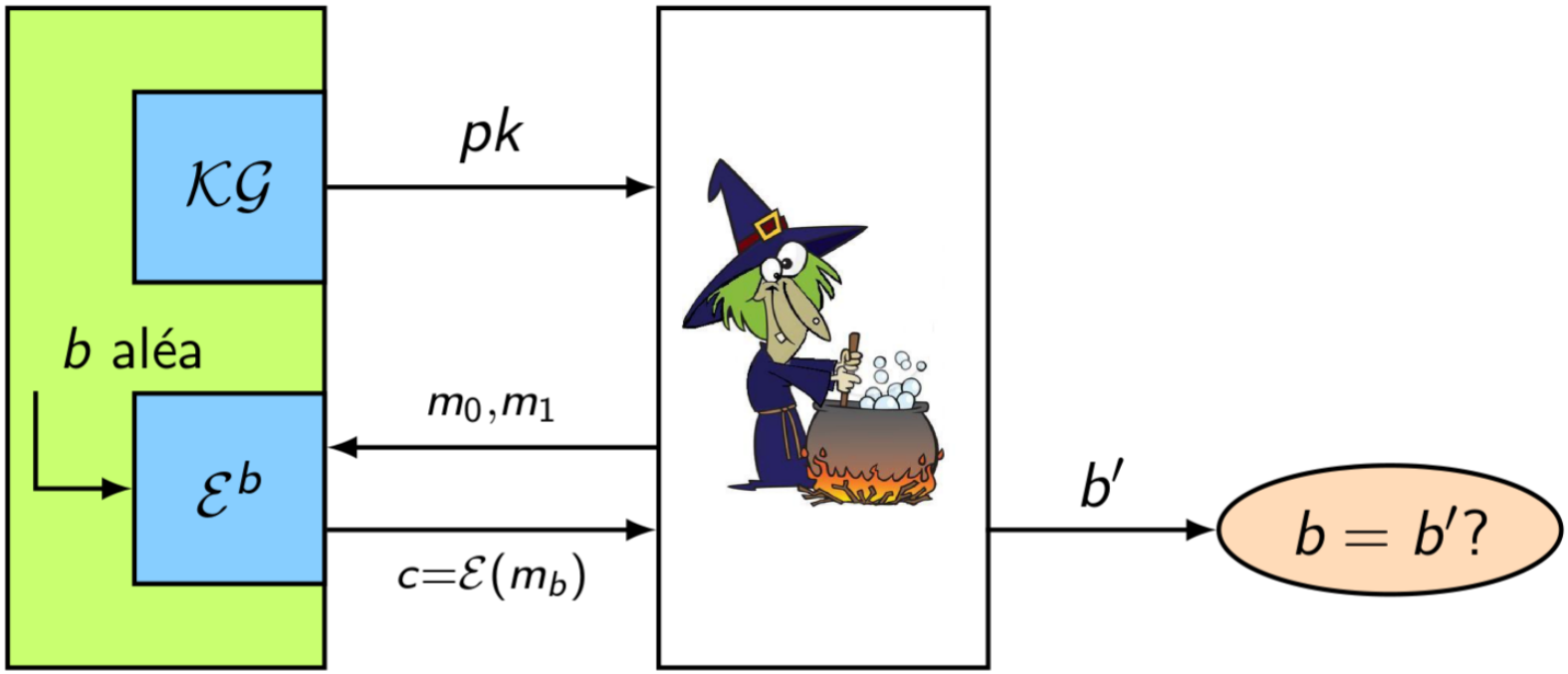
OW – One-Wayness

Déchiffrer un message arbitraire



Obtenir un bit d'information

IND – Indistinguishability

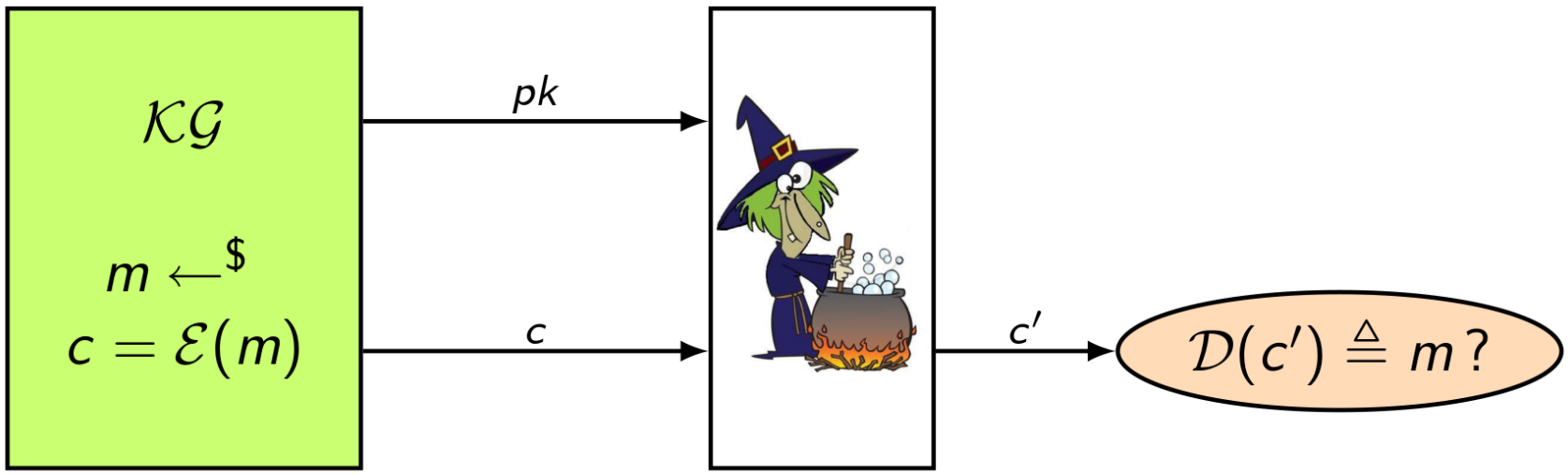


C'est $\mathcal{A}$ qui choisit les messages m_0 et m_1 !

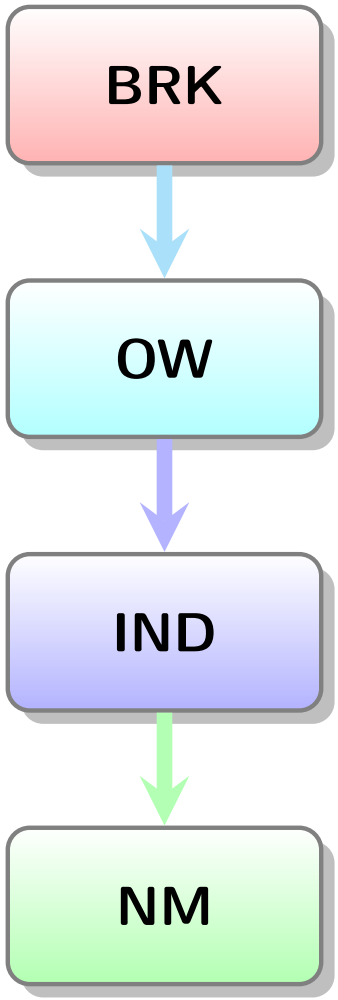
Modiffier un chiffré

NM – Non-Malleability

Fabriquer deux chiffrés « liés » (*plaintext-dependent*)



Hiérarchie de sécurité



Moyens

Oracles

L'adversaire a accès à des oracles :

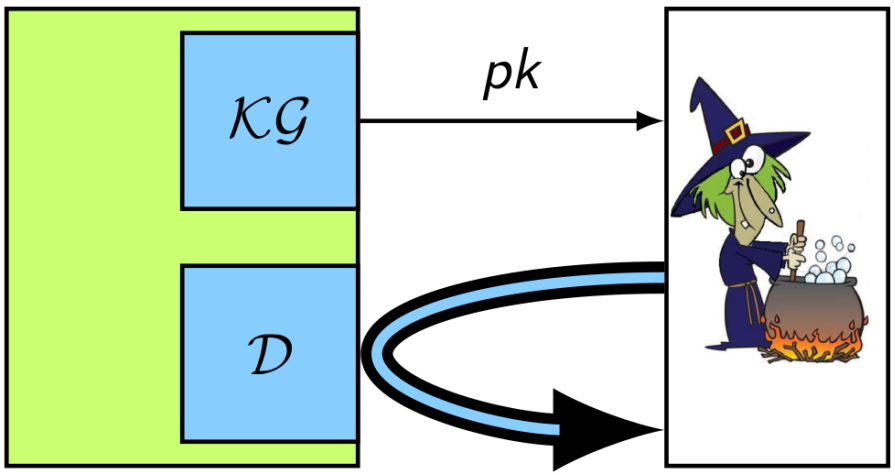
- **chiffrement** de tous les messages de son choix
- **déchiffrement** de tous les messages de son choix

Moyens de l'attaquant

1. **KPA – Clairs connus** : il voit des couples clairs/chiffrés
2. **CPA – Clairs choisis** : il chiffre des messages de son choix
3. **CCA1 – Chiffrés choisis** : il peut faire déchiffrer des messages arbitraires avant recevoir le "challenge"
4. **CCA2 – Attaque adaptative** : il peut faire déchiffrer des messages après avoir reçu le "challenge"
   (restriction de ne pas faire déchiffrer ce challenge)

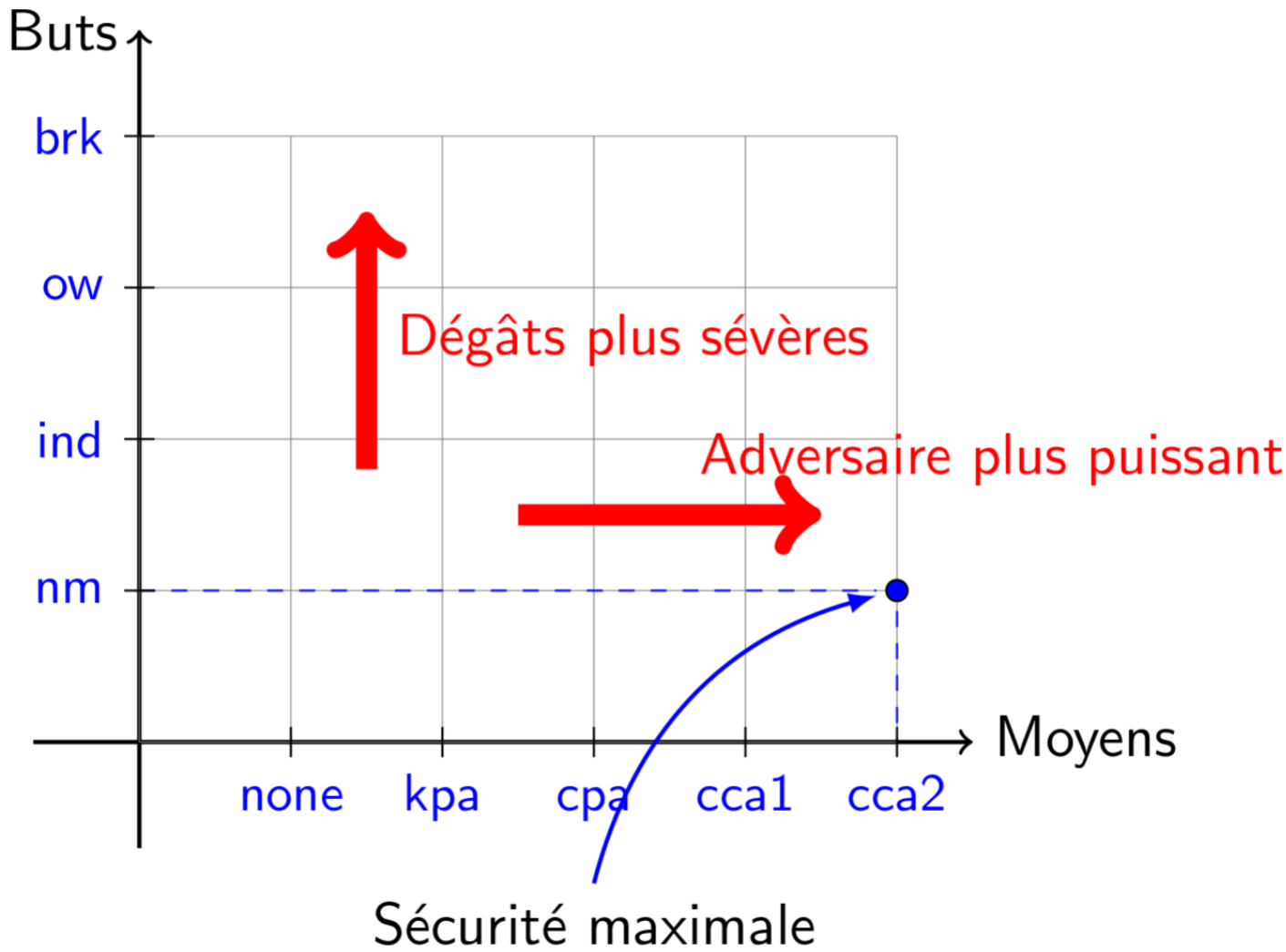
CCA – Attaques à chiffrés choisis

Attaques les plus sévères (*Chosen-Ciphertext Attacks, CCA*)



$\mathcal{A}$ observe des déchiffrements et espère obtenir de l'information

Les axes d'analyse



Modèle de sécurité

Attaques possibles

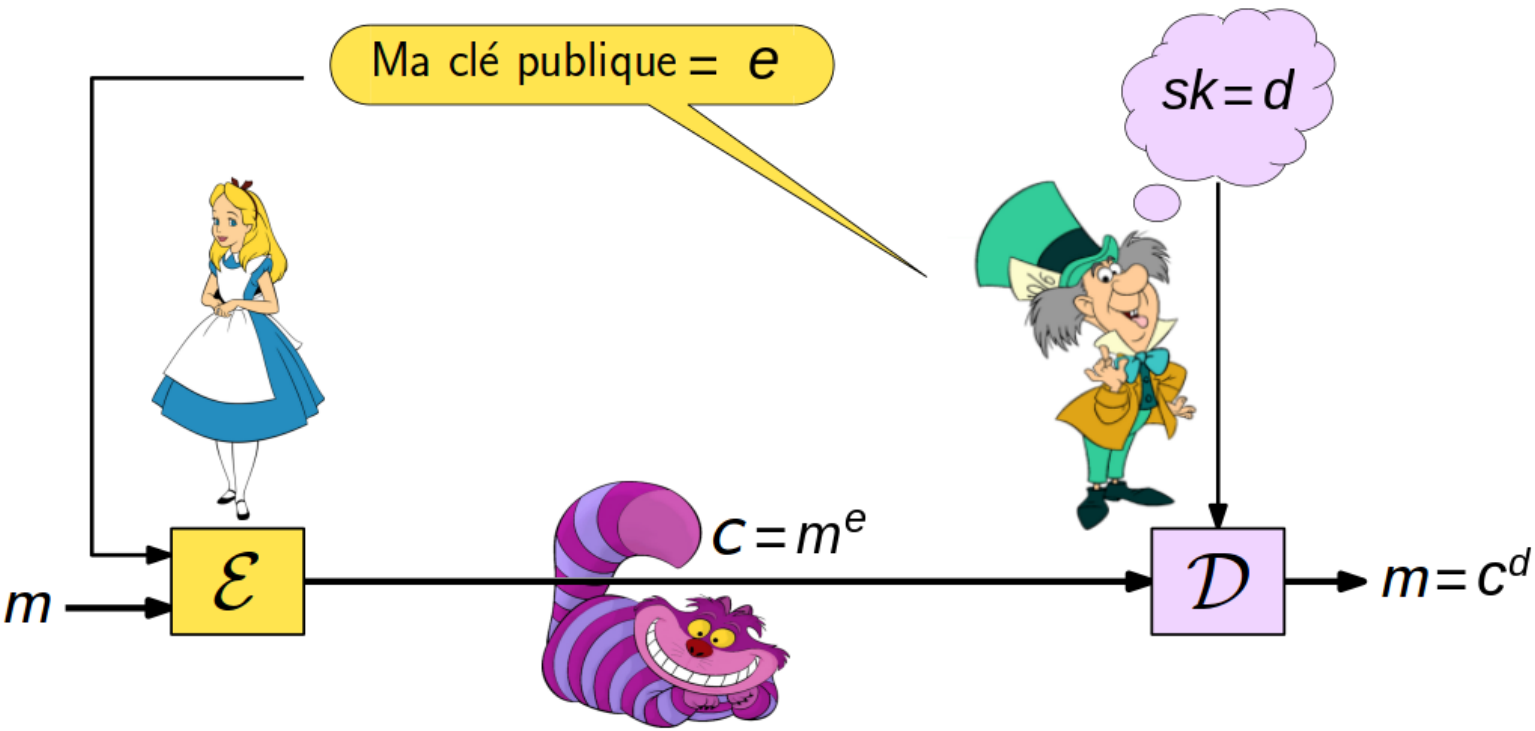
Un modèle d'attaque = un but + des moyens

Exemple :

- Sécurité **OW-CPA** : chiffrement à sens unique sous une attaque à clair choisi

Prendre le cas de RSA : quel niveau peut-on espérer atteindre ?

Analyse de sécurité pour RSA



Analyse de sécurité pour RSA

On suppose RSA bien utilisé

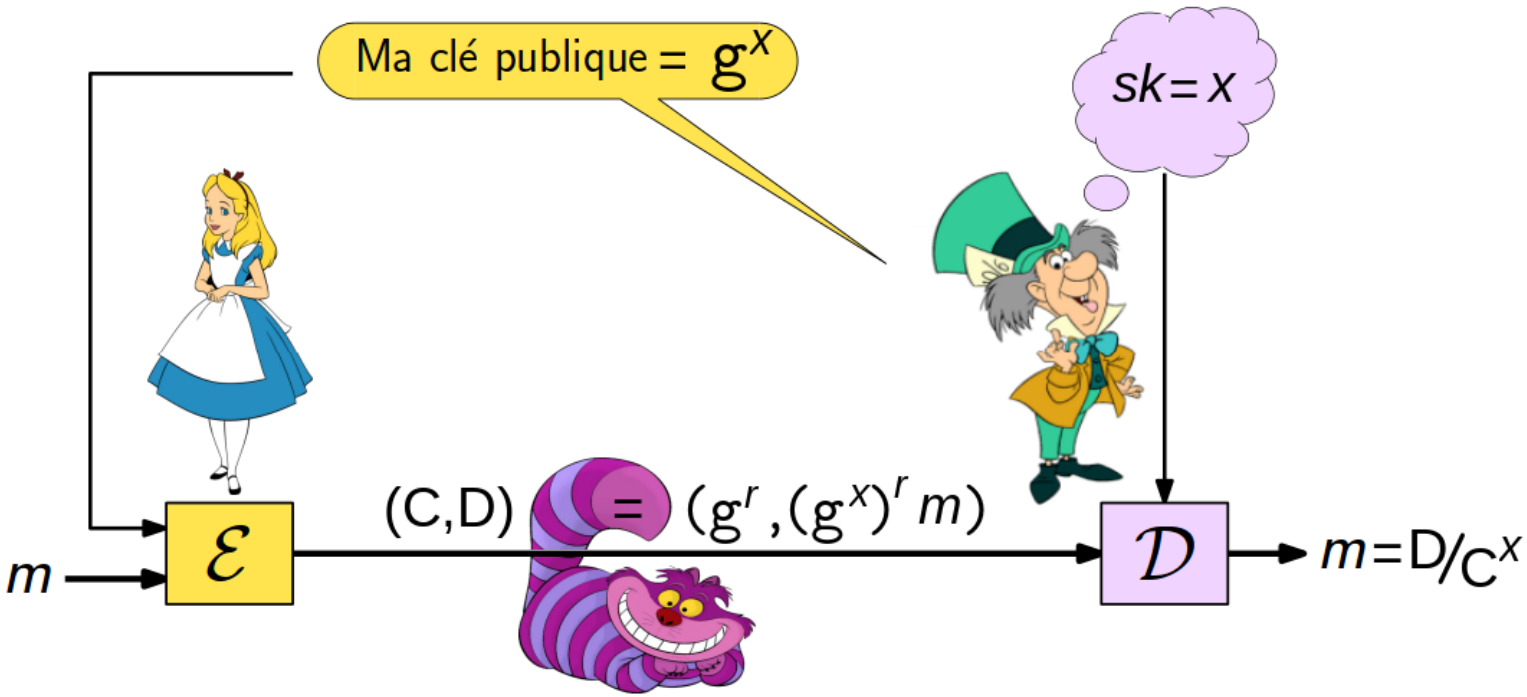
Sécurité

- **BRK-CPA** : clé secrète —> **factorisation**
- **OW-CPA** : extraction de m à partir de $c = m^e$ racine e-ième, soit **problème RSA**

Vulnérabilités

- **IND-CPA** : il y a (au moins) un bit d'information qui fuit $(m = 1)$
- **OW-CCA** : étant donné $c = m^e$, l'attaquant $\mathcal{A}$ demande à déchiffrer $c' = 2^e \cdot c$ et obtient $2m$

Analyse de sécurité pour ElGamal



Problèmes difficiles

Soit $\mathbb{G}$ un groupe multiplicatif cyclique, $\mathbb{G} = \langle g \rangle$:

Logarithme discret (DLOG)

Etant donnés $g \in \mathbb{G}$ et $X = g^x$,

Calculer $\log_g(X) = x$

Calculer Diffie-Hellman (CDH)

Etant donnés g, $A = g^a$ et $B = g^b$,

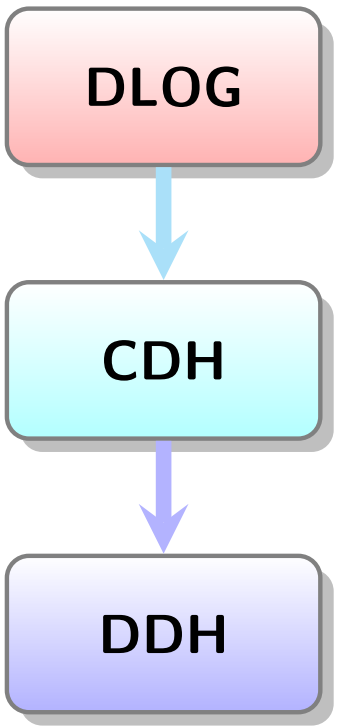
Calculer $C = CDH(A, B) = g^{ab}$

Décider si Diffie-Hellman (DDH)

Etant donnés g, $A = g^a$, $B = g^b$ et $C = g^c$ dans $\mathbb{G}$

Décider si $C = g^{ab}$

Hiérarchie



CDH $<$ DLOG

Etant donnés g, $A = g^a$ et $B = g^b$,

- on calcule b =DLOG(B)
- on trouve $C = A^b = g^{ab}$

DDH $<$ CDH

Etant donnés g, $A = g^a$, $B = g^b$ et $C = g^c$

- on calcule $CDH(A, B) = g^{ab}$
- on compare avec C

Analyse de sécurité pour ElGamal

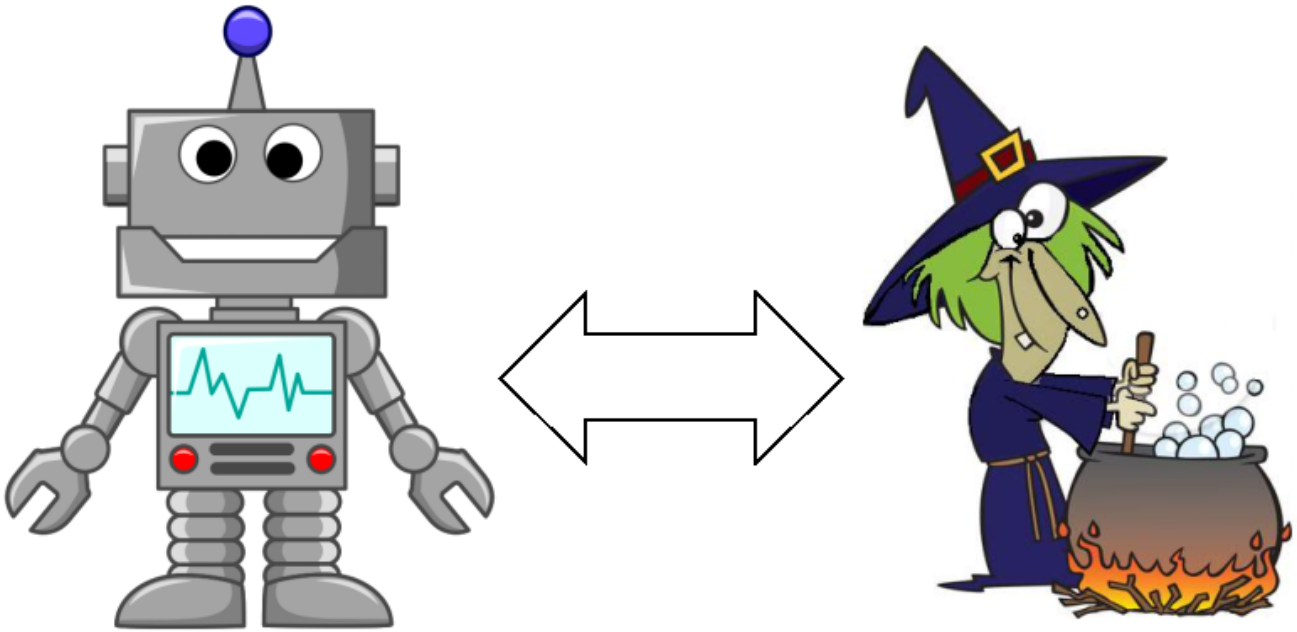
Sécurité

- **BRK-CPA** : clé secrète —> **DLOG**
- **OW-CPA** : extraction de $M = D \cdot C^{-x}$ à partir de (C, D) probléme calculatoire Diffie-Hellman **CDH**
- **IND-CPA** : distinguer entre M_0 et M_1 problème décisionnel Diffie-Hellman **DDH**

Vulnérabilités

- **OW-CCA** : étant donné (C, D), l'attaquant $\mathcal{A}$ demande à déchiffrer $(C, 2D)$ et obtient $2M$

Preuves de sécurité



Principe de la preuve

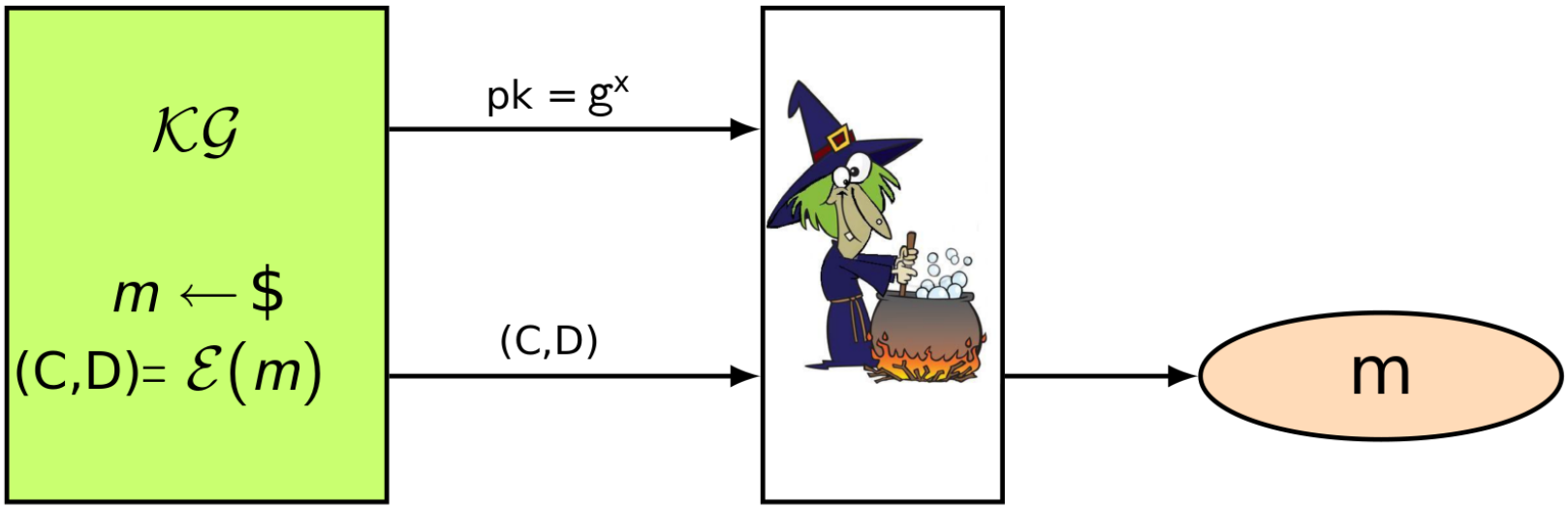
Hypothèse : Un problème **P** diffcile

Réduction : Si $\mathcal{A}$, un adversaire (polynomial), casse le schéma de chiffrement, alors $\mathcal{A}$ peut être utilisé pour construire un algorithme **ROBOT** qui resout **P**

Rétrouver le message chiffré

OW - CPA

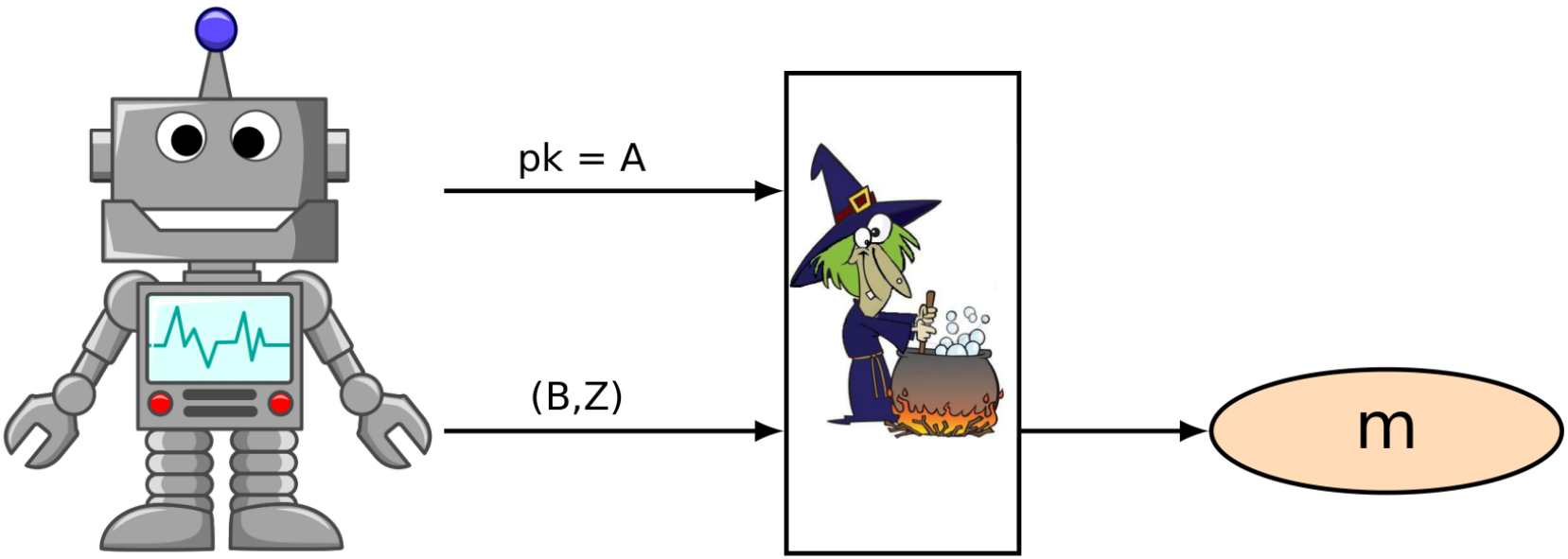
Déchiffrer un message arbitraire



$$C = g^r \quad \text{et} \quad D = (g^x)^r \cdot m \quad \Rightarrow \quad \mathcal{A} \text{ calcule } m = D/(C^x)$$

Réduction OW - CPA $\Rightarrow$ CDH

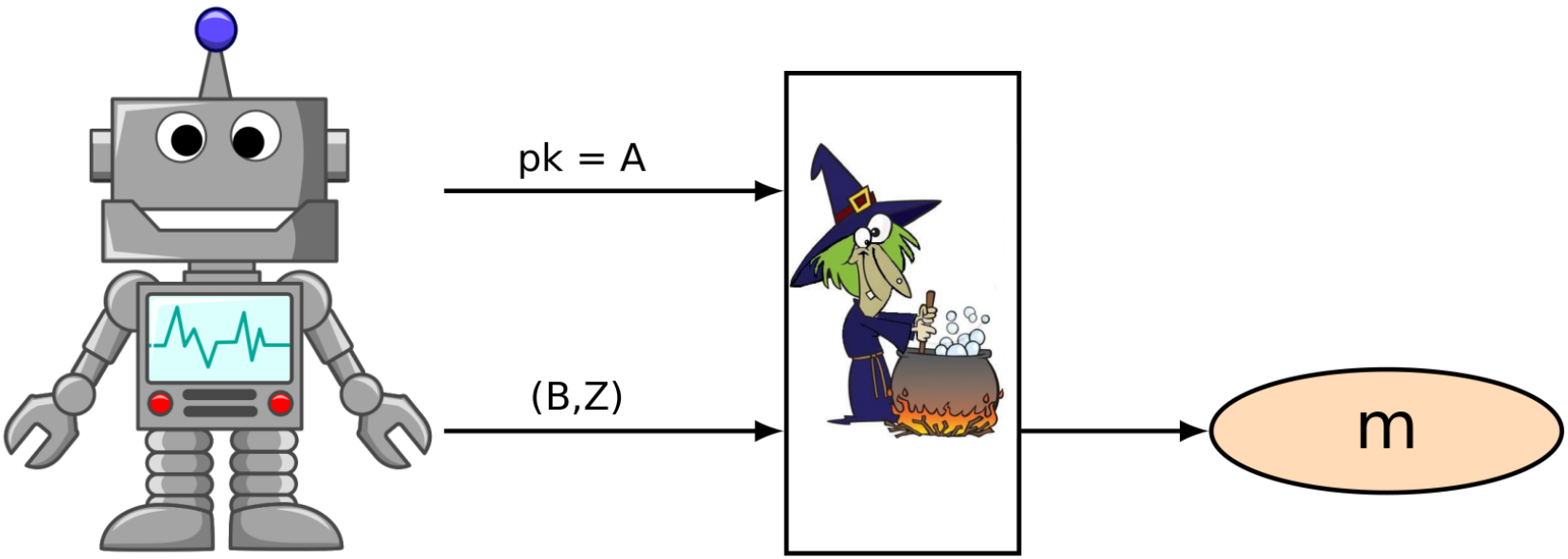
input $(A = g^a, B = g^b)$



But : $C = g^{ab} = B^a$

Réduction OW - CPA $\Rightarrow$ CDH

input $(A = g^a, B = g^b)$

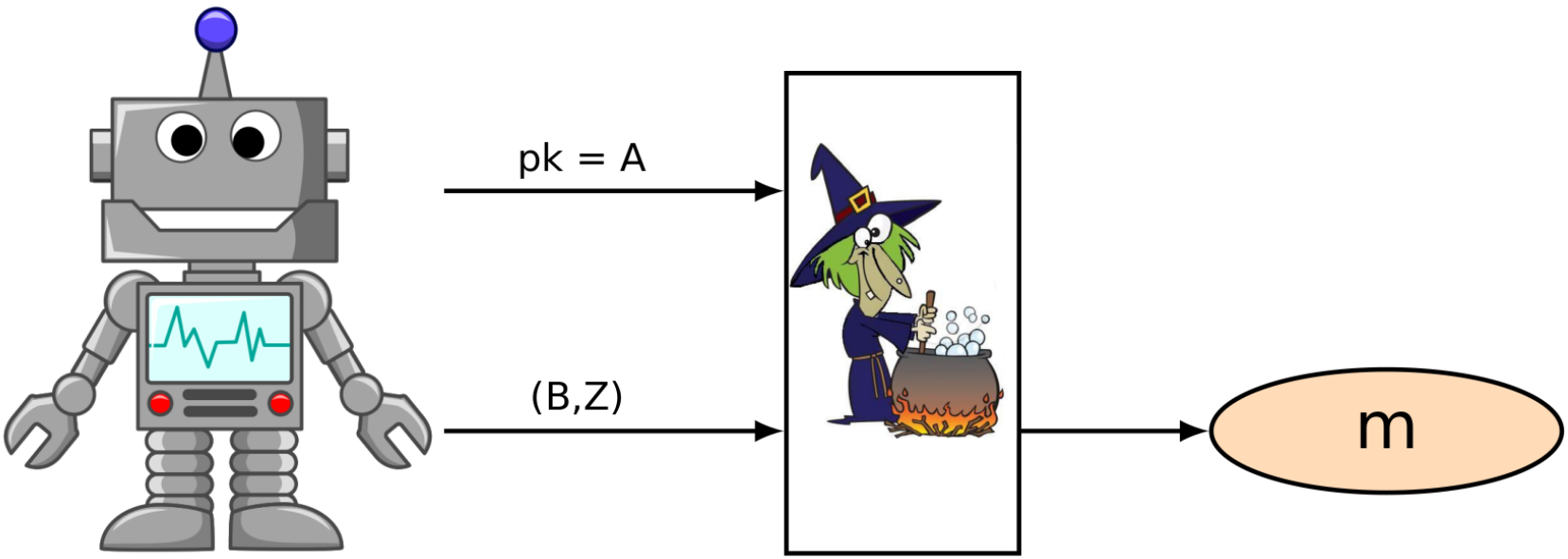


But : $C = g^{ab} = B^a$

$\mathcal{A}$ calcule $m = Z/(B^a)$

Réduction OW - CPA $\Rightarrow$ CDH

input $(A = g^a, B = g^b)$



But : $C = g^{ab} = B^a$

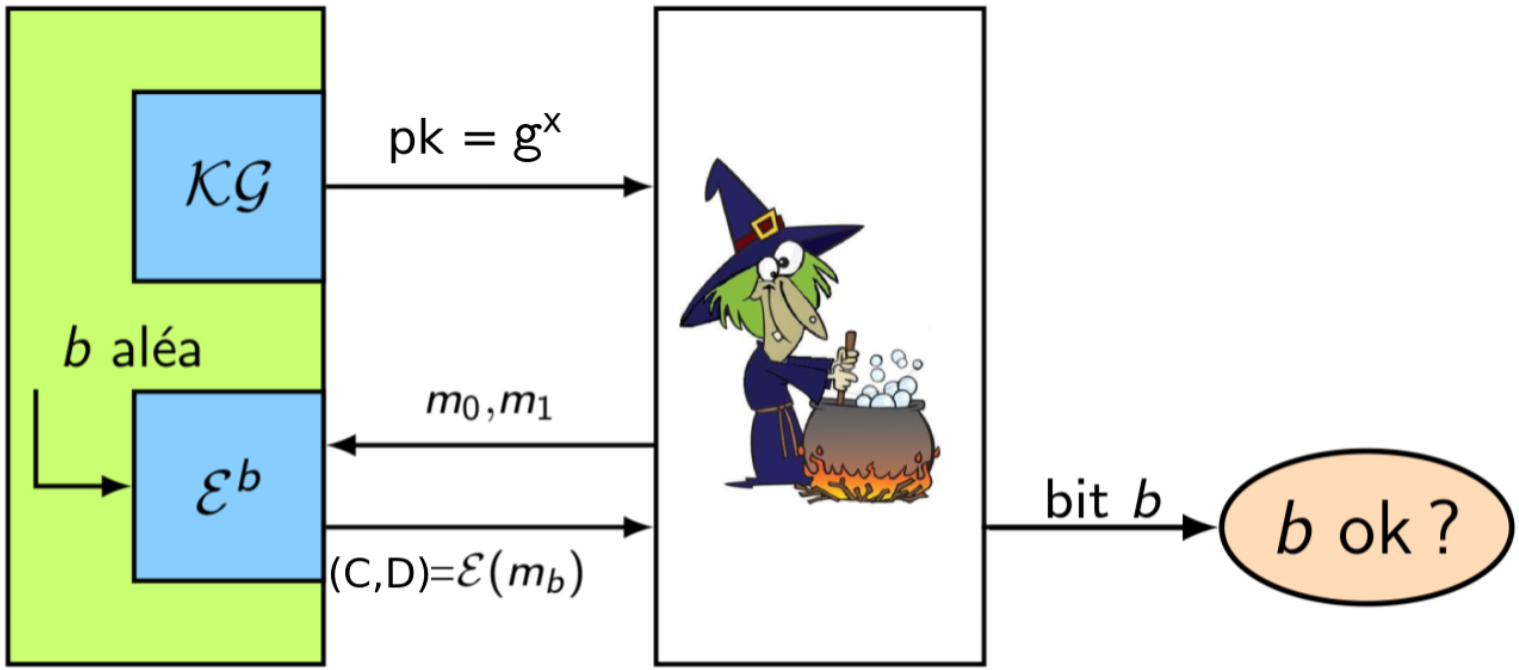
$\mathcal{A}$ calcule $m = Z/(B^a)$

output $Z/m = B^a$

Distinguer entre deux chiffrés

IND - CPA

$\mathcal{A}$ choisit deux messages m_0 et m_1 :

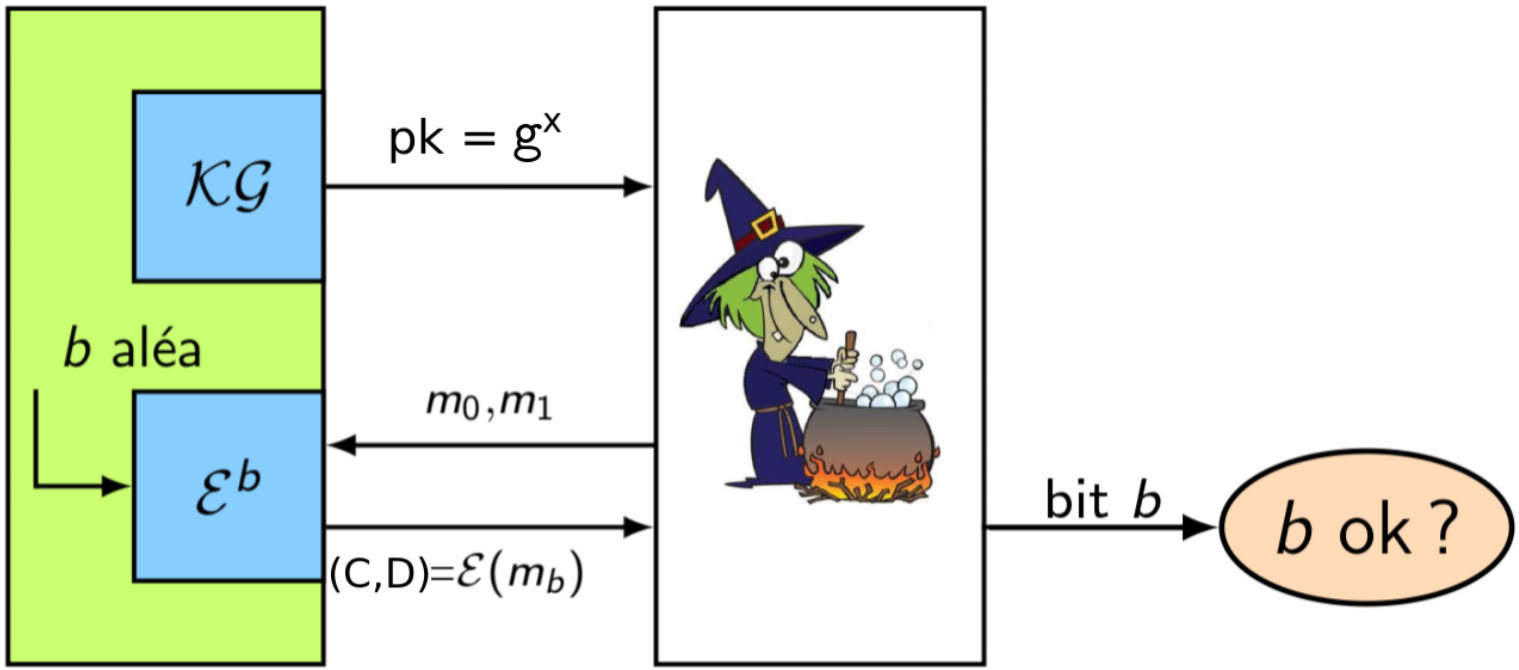


$C = g^r$ et $D = (g^x)^r \cdot m_b \Rightarrow \mathcal{A}$ reconnaît le chiffré de m_b

Distinguer entre deux chiffrés

IND - CPA

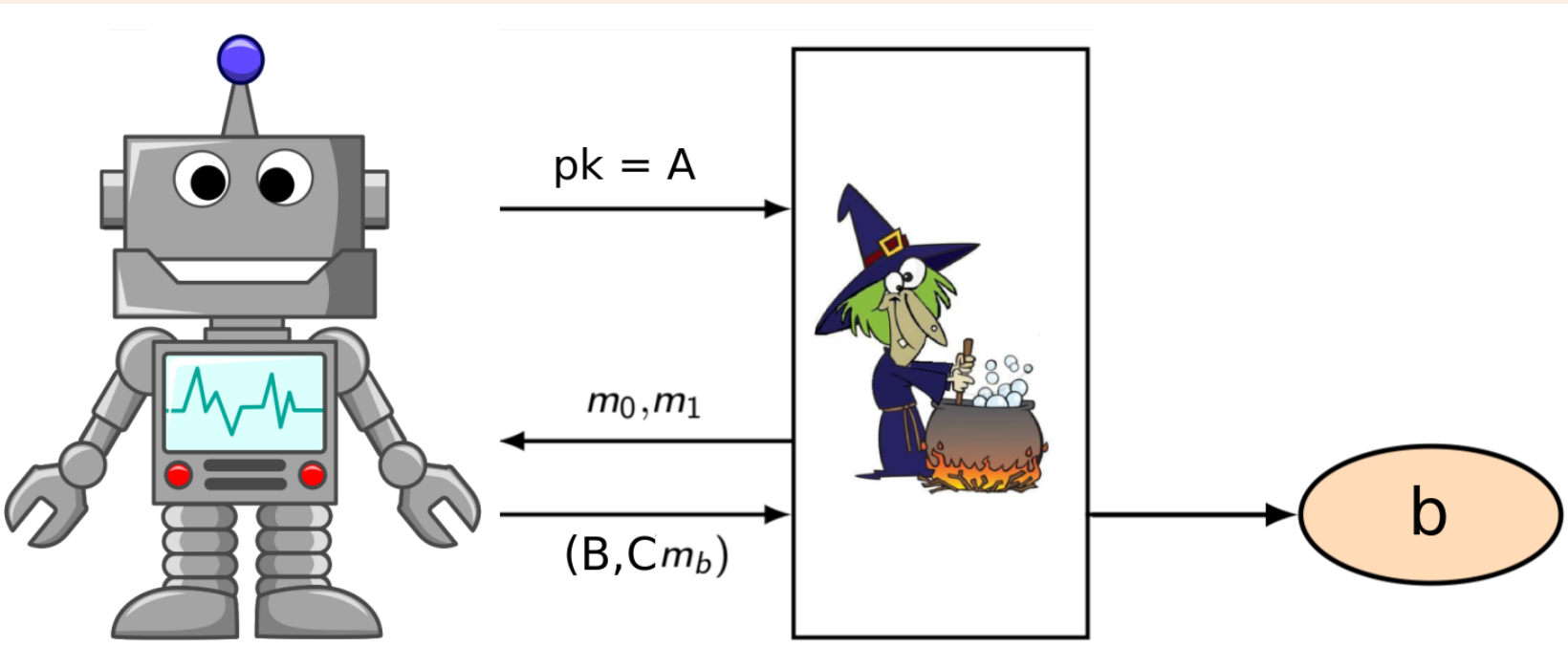
$\mathcal{A}$ choisit deux messages m_0 et m_1 :



$C = g^r$ et $D = (g^x)^r \cdot m_b \Rightarrow \mathcal{A}$ reconnaît le chiffré de m_b

Réduction IND - CPA $\Rightarrow$ DDH

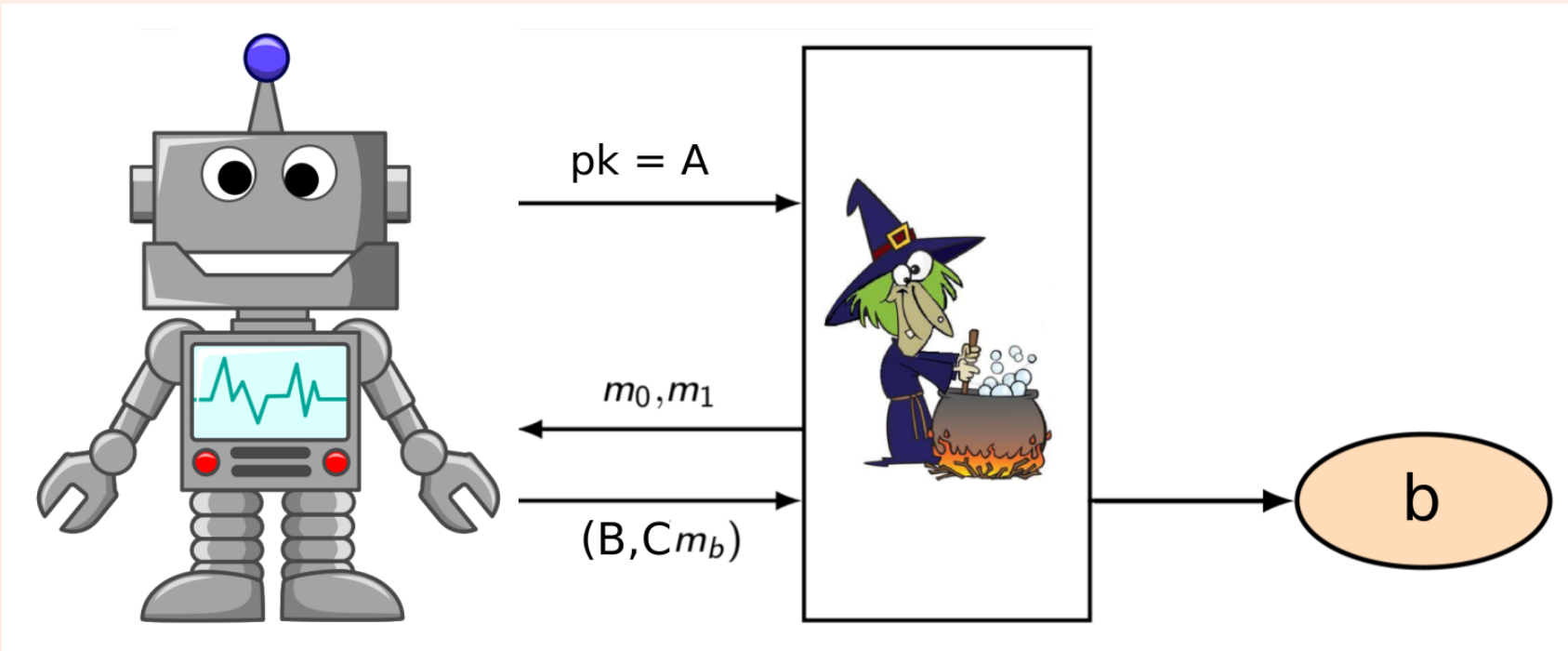
input $(A = g^a, B = g^b, C = g^c)$



But : $C \stackrel{?}{=} g^{ab}$

Réduction IND - CPA ⇒ DDH

input $(A = g^a, B = g^b, C = g^c)$

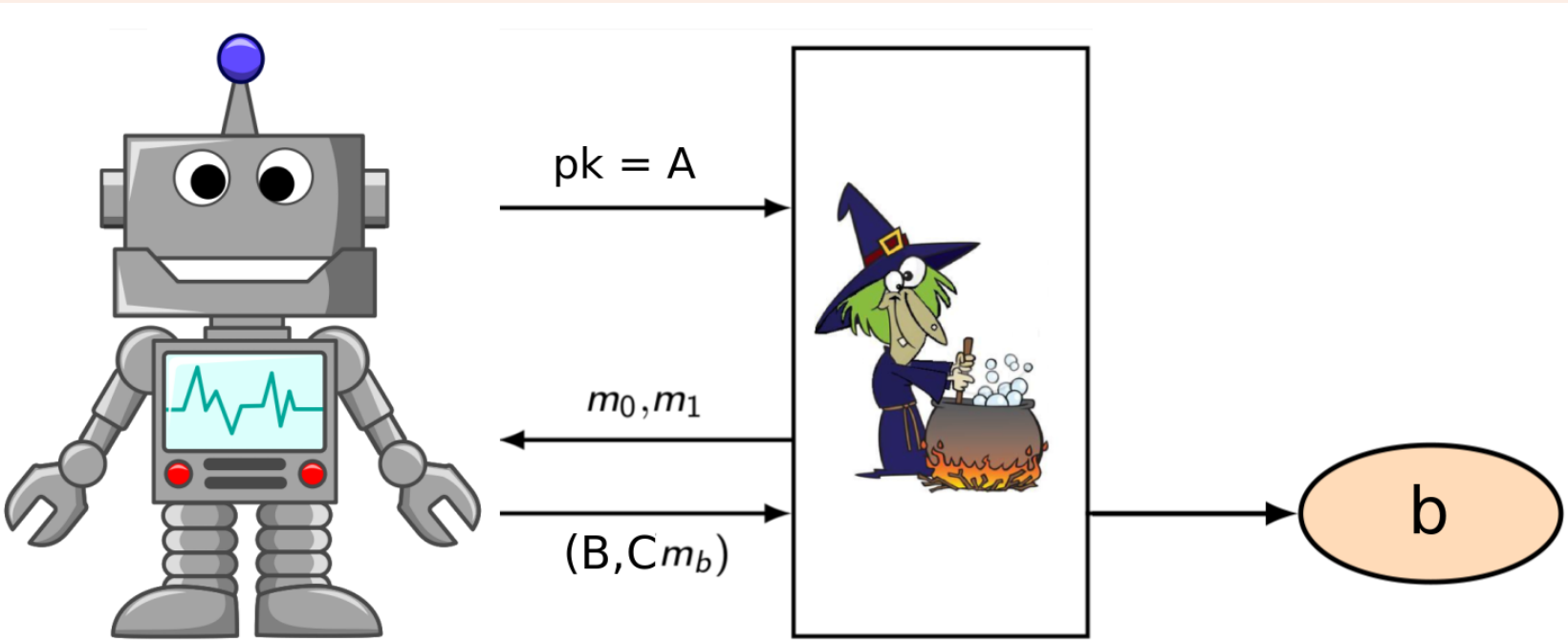


But : $C \stackrel{?}{=} g^{ab}$

Si $b' = 0$ **output** oui

Réduction IND - CPA $\Rightarrow$ DDH

input $(A = g^a, B = g^b, C = g^c)$



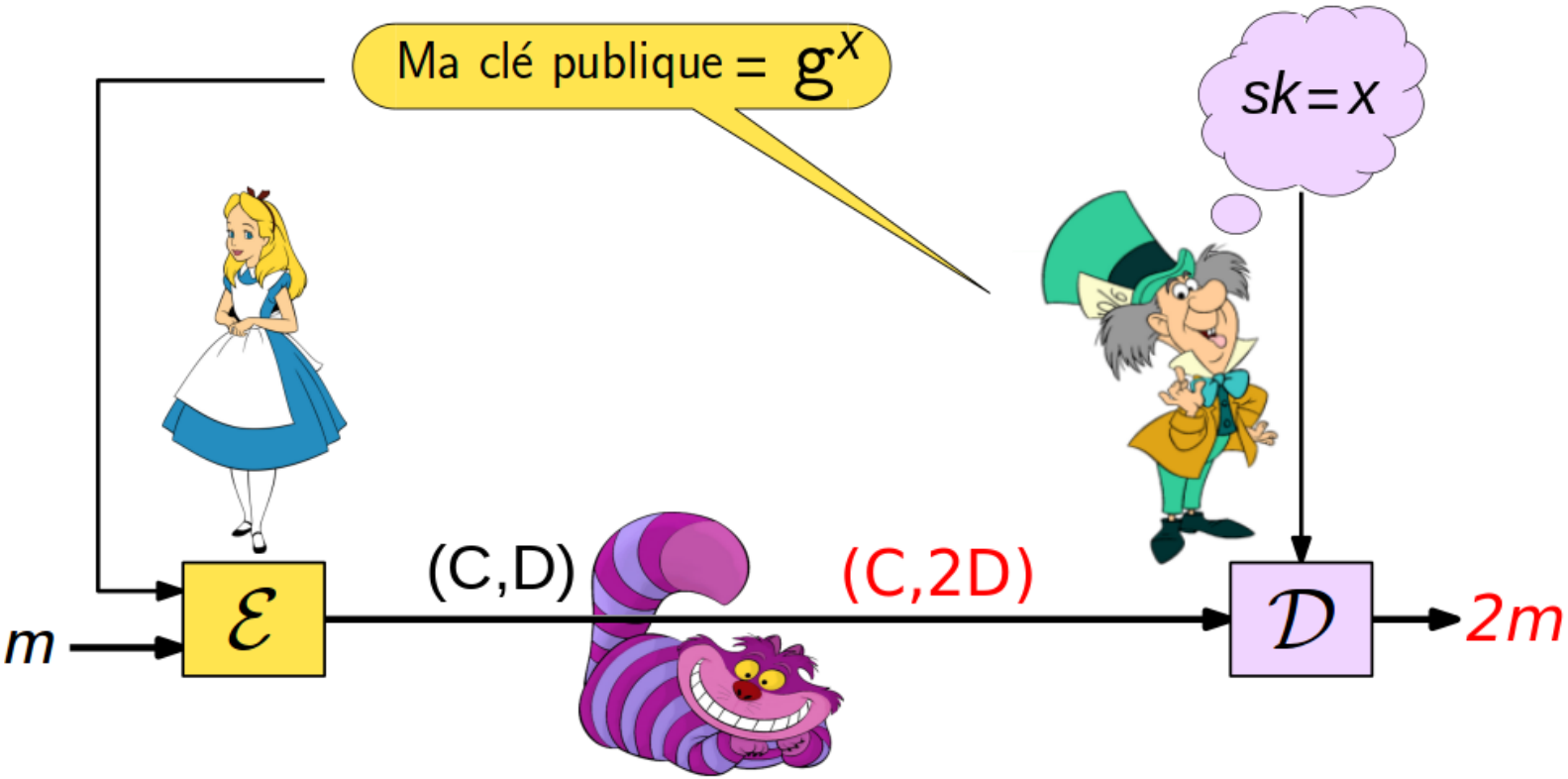
But : $C \stackrel{?}{=} g^{ab}$

Si $b' = 0$ **output** oui

Si $C = g^{ab}$: $(B, Cm_0) = (g^b, g^{ab}m_0) = \mathcal{E}(m_0)$

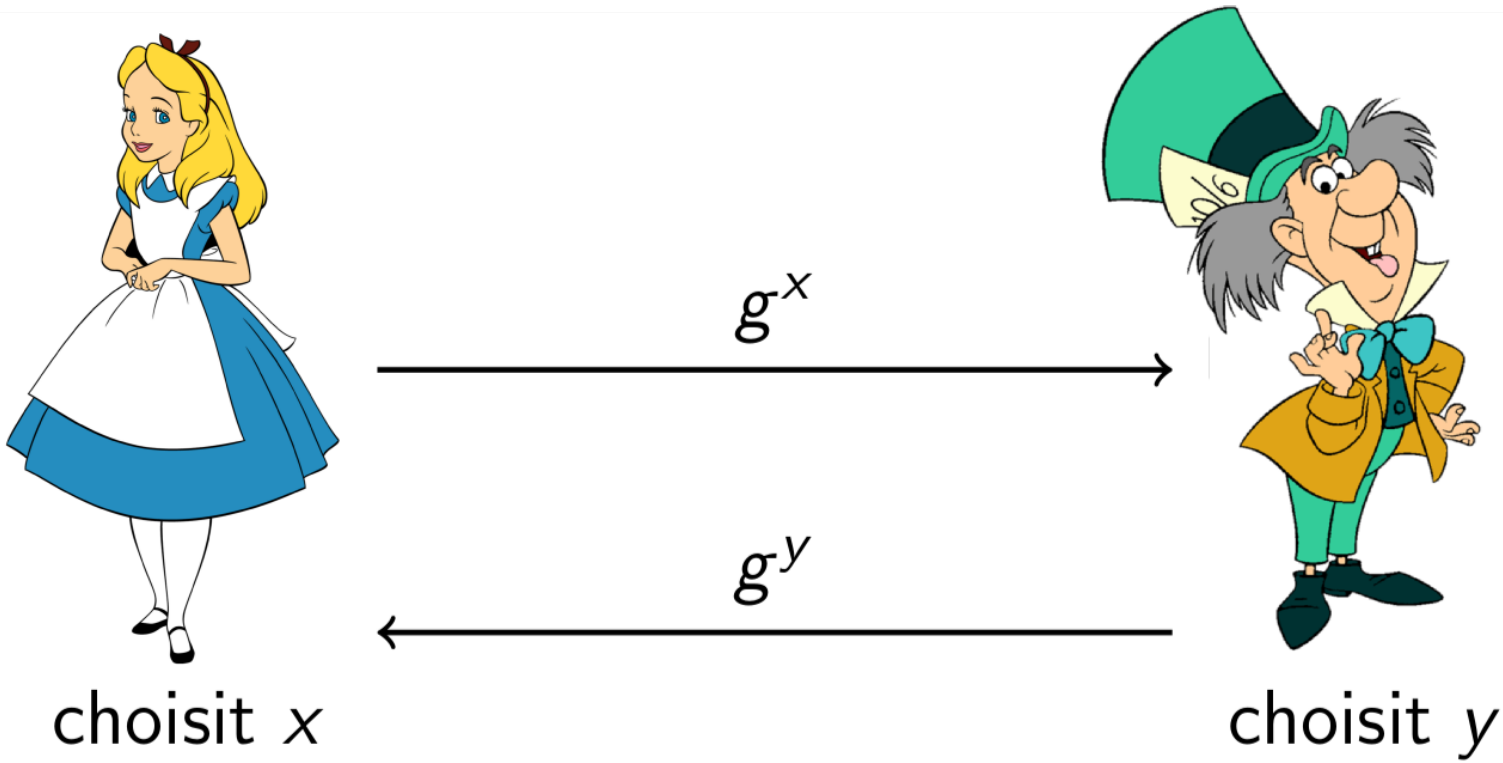
Si $C = g^c$: $(B, Cm_0) = (g^b, g^c m_0)$ est un aléa

Attack IND - CCA



Solution : Authentification

Echange de clé



clé commune g^{xy}

Problèmes difficiles

Soit $\mathbb{G}$ un groupe multiplicatif cyclique, $\mathbb{G} = \langle g \rangle$:

Calculer Diffie-Hellman (CDH)

Etant donnés g, $A = g^a$ et $B = g^b$,

Calculer $C = CDH(A, B) = g^{ab}$

Décider si Diffie-Hellman (DDH)

Etant donnés

g, $A = g^a$, $B = g^b$ et $C = g^c$ dans $\mathbb{G}$

Décider si $C = g^{ab}$

Réduction de Diffie-Hellman

Attaquer Diffie-Hellman

- Si le problème **CDH** est résolu, alors l'attaquant peut calculer une clé Diffie-Hellman
- Si le problème **DDH** est résolu, alors l'attaquant peut distinguer entre une clé valide et une clé fausse