

Mécanismes avancés/ Preuves par jeux

Olivier Sanders (Orange)

(Rappel) Chiffrement El Gamal

- Clé Publique: G un groupe d'ordre p . g un générateur de G . $h = g^x$
- Clé Secrète: x
- Encrypt(pk,m) : r aléa mod p . $C = (g^r, h^r * m)$
- Decrypt(sk,c): Soit $C = (c_1, c_2)$. $m = c_2 * c_1^{-x}$

(Rappel) Chiffrement El Gamal

- Clé Publique: G un groupe d'ordre p . g un générateur de G . $h = g^x$
- Clé Secrète: x
- $\text{Encrypt}(pk, m)$: r aléa mod p . $C = (g^r, h^r * m)$
- $\text{Decrypt}(sk, c)$: Soit $C = (c_1, c_2)$. $m = c_2 * c_1^{-x}$
- El Gamal satisfait la propriété IND-CPA sous l'hypothèse DDH
- El Gamal ne satisfait pas la propriété IND-CCA

Naor-Yung

- Soit E le protocole de chiffrement El-Gamal.
- Keygen: deux exécutions de E.Keygen pour obtenir (sk_1, pk_1) et (sk_2, pk_2)
 $pk = (pk_1, pk_2)$ et $sk = (sk_1)$
- Encrypt(pk, m):
 $C_1 = E.Encrypt(pk_1, m)$ et $C_2 = E.Encrypt(pk_2, m)$.
Génération d'une preuve P zéro-knowledge non-interactive que C_1 et C_2 chiffrés d'un même message m
 $C = (C_1, C_2, P)$
- Decrypt(sk, C)
Vérification de la preuve P
Si correct, $m = Decrypt(sk_1, C_1)$

Naor-Yung (Détails)

- On a $pk_1 = h_1 = g^{(x_1)}$ et $pk_2 = h_2 = g^{(x_2)}$
- $C_1 = [g^{(r_1)}, h_1^{(r_1)} * m]$ et $C_2 = [g^{(r_2)}, h_2^{(r_2)} * m]$
- Preuve P:

k_1 et k_2
aléatoires

$$K_1 = g^{(k_1)} ; K_2 = g^{(k_2)} ; K_3 = h_1^{(k_1)} / h_2^{(k_2)}$$

c , aléatoire

$$z_1 = k_1 + c * r_1 ; z_2 = k_2 + c * r_2$$

$$g^{(z_1)} = K_1 * C_1[1]^c$$

$$g^{(z_2)} = K_2 * C_2[1]^c$$

$$h_1^{(z_1)} / h_2^{(z_2)} = K_3 * (C_1[2] / C_2[2])^c$$

Naor-Yung (Détails)

- On a $pk_1 = h_1 = g^{(x_1)}$ et $pk_2 = h_2 = g^{(x_2)}$
- $C_1 = [g^{(r_1)}, h_1^{(r_1)} * m]$ et $C_2 = [g^{(r_2)}, h_2^{(r_2)} * m]$
- Preuve P:

k_1 et k_2
aléatoires

$$K_1 = g^{(k_1)} ; K_2 = g^{(k_2)} ; K_3 = h_1^{(k_1)} / h_2^{(k_2)}$$

c , aléatoire

$$z_1 = k_1 + c * r_1 ; z_2 = k_2 + c * r_2$$

$$g^{(z_1)} = K_1 * C_1[1]^c$$

$$g^{(z_2)} = K_2 * C_2[1]^c$$

$$h_1^{(z_1)} / h_2^{(z_2)} = K_3 * (C_1[2] / C_2[2])^c$$

On utilise Fiat-Shamir pour rendre la preuve non-interactive

Sécurité IND-CCA (Rappel)

- 1) $(sk, pk) \leftarrow \text{Keygen}(1^n)$
- 2) $(m_0, m_1) \leftarrow A^\circ(pk)$
- 3) $C^* \leftarrow \text{Encrypt}(pk, m_b)$ pour un bit b choisi aléatoirement.
- 4) $b' \leftarrow A^\circ(pk, C^*)$

Jeu IND-CCA

Le jeu **IND-CCA** offre un accès à un **oracle de déchiffrement** O :
 $O(c)$ retourne $\text{Decrypt}(sk, m)$

Idée de la preuve

- Un chiffré Naor-Yung est constitué de

Un chiffré El Gamal C_1
sous pk_1

Un chiffré El Gamal C_2
sous pk_2

Une preuve P que tout est
bien formé

On veut montrer que le chiffrement est IND-CCA si El Gamal est IND-CPA:

- On fait l'hypothèse qu'il existe un adversaire A contre l'IND-CCA
- On montre comment utiliser A pour casser l'IND-CPA d'El Gamal

Idée de la preuve

- Un chiffré Naor-Yung est constitué de

Un chiffré El Gamal C_1
sous pk_1

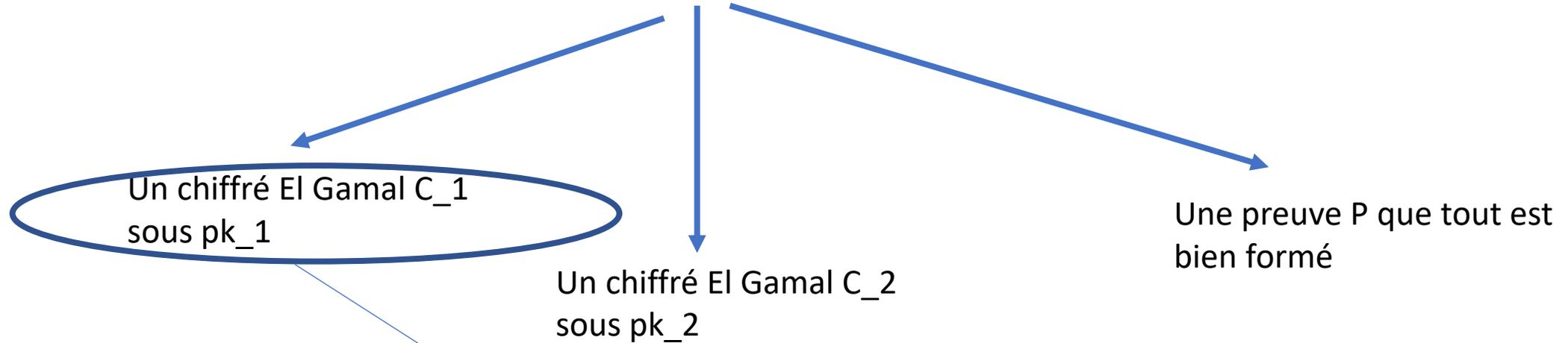
Un chiffré El Gamal C_2
sous pk_2

Une preuve P que tout est
bien formé

Difficulté 1: Insérer la clé publique pk du jeu IND-CPA ainsi que le challenge c^* tout en étant capable de répondre aux requêtes de déchiffrement

Idée de la preuve

- Un chiffré Naor-Yung est constitué de



On génère la paire de clés (sk_1, pk_1)
normalement

Difficulté 1: Insérer la clé publique pk du jeu IND-CPA ainsi que le challenge c^*
tout en étant capable de répondre aux requêtes de déchiffrement

Solution: On traite différemment chaque chiffré

Idée de la preuve

- Un chiffré Naor-Yung est constitué de

Un chiffré El Gamal C_1
sous pk_1

Une preuve P que tout est
bien formé

Un chiffré El Gamal C_2
sous pk_2

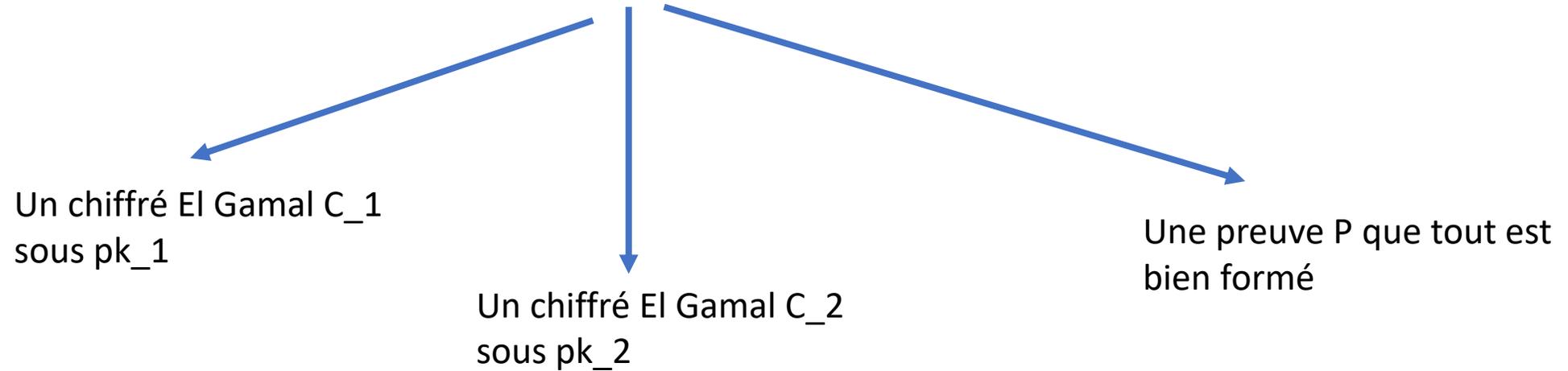
On insère le challenge dans la deuxième clé:
 $pk_2 = pk$

Difficulté 1 : Insérer la clé publique pk du jeu IND-CPA ainsi que le challenge c^*
tout en étant capable de répondre aux requêtes de déchiffrement

Solution: On traite différemment chaque chiffré

Idée de la preuve

- Un chiffré Naor-Yung est constitué de

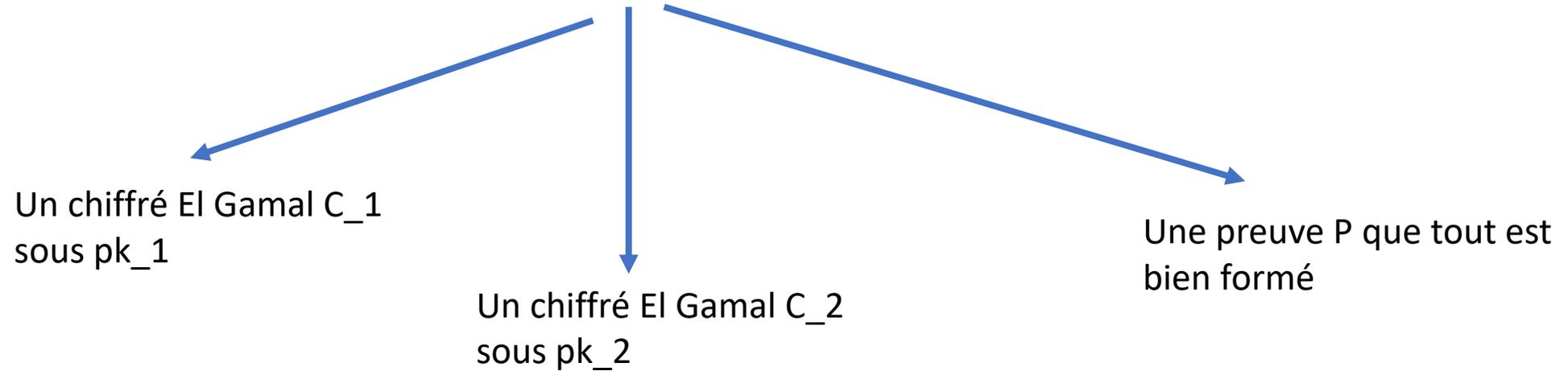


Lorsque l'adversaire soumet une requête de déchiffrement sur $C = (C_1, C_2, P)$:

1) Je déchiffre en utilisant sk_1 et je retourne m ?

Idée de la preuve

- Un chiffré Naor-Yung est constitué de



Lorsque l'adversaire soumet une requête de déchiffrement sur $C = (C_1, C_2, P)$:

1) ~~Je déchiffre en utilisant sk_1 et je retourne m ?~~

Rien ne dit que C est bien formé !

Répondre serait une divergence par rapport au fonctionnement normal

Idée de la preuve

- Un chiffré Naor-Yung est constitué de

Un chiffré El Gamal C_1
sous pk_1

Un chiffré El Gamal C_2
sous pk_2

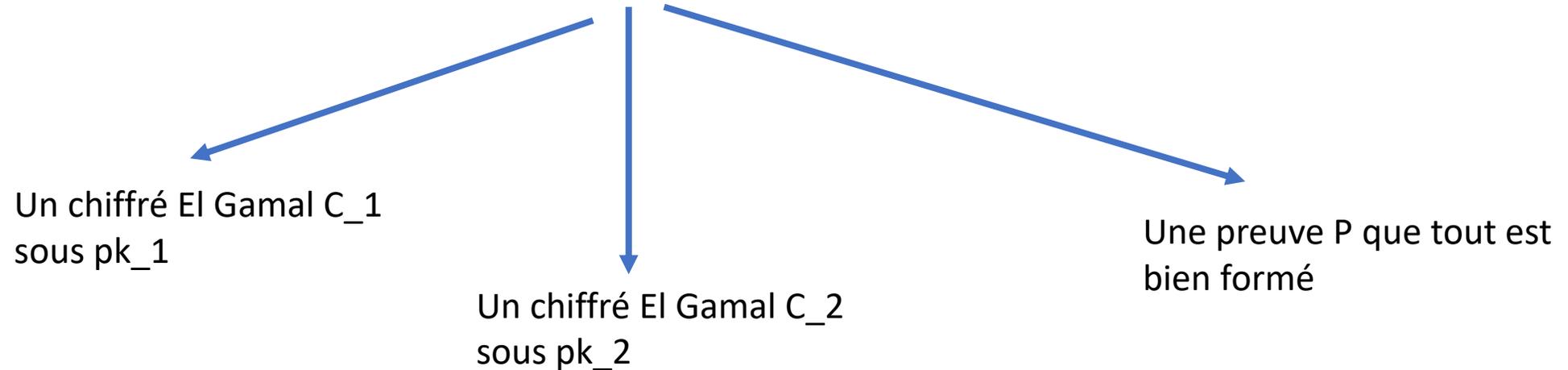
Une preuve P que tout est
bien formé

Lorsque l'adversaire soumet une requête de déchiffrement sur $C = (C_1, C_2, P)$:

1) Je vérifie que P est valide

Idée de la preuve

- Un chiffré Naor-Yung est constitué de



Lorsque l'adversaire soumet une requête de déchiffrement sur $C = (C_1, C_2, P)$:

- 1) Je vérifie que P est valide
- 2) Je déchiffre C_1 en utilisant sk_1

Le déchiffrement peut être parfaitement simulé

Idée de la preuve

- Un chiffré Naor-Yung est constitué de

Un chiffré El Gamal C_1
sous pk_1

Un chiffré El Gamal C_2
sous pk_2

Une preuve P que tout est
bien formé

Difficulté 2: Utiliser le chiffré challenge (IND-CCA) pour distinguer le chiffré challenge c^* IND-CPA

Idée de la preuve

- Un chiffré Naor-Yung est constitué de

Un chiffré El Gamal C_1
sous pk_1

Un chiffré El Gamal C_2
sous $pk_2 = pk$

Une preuve P que tout est
bien formé

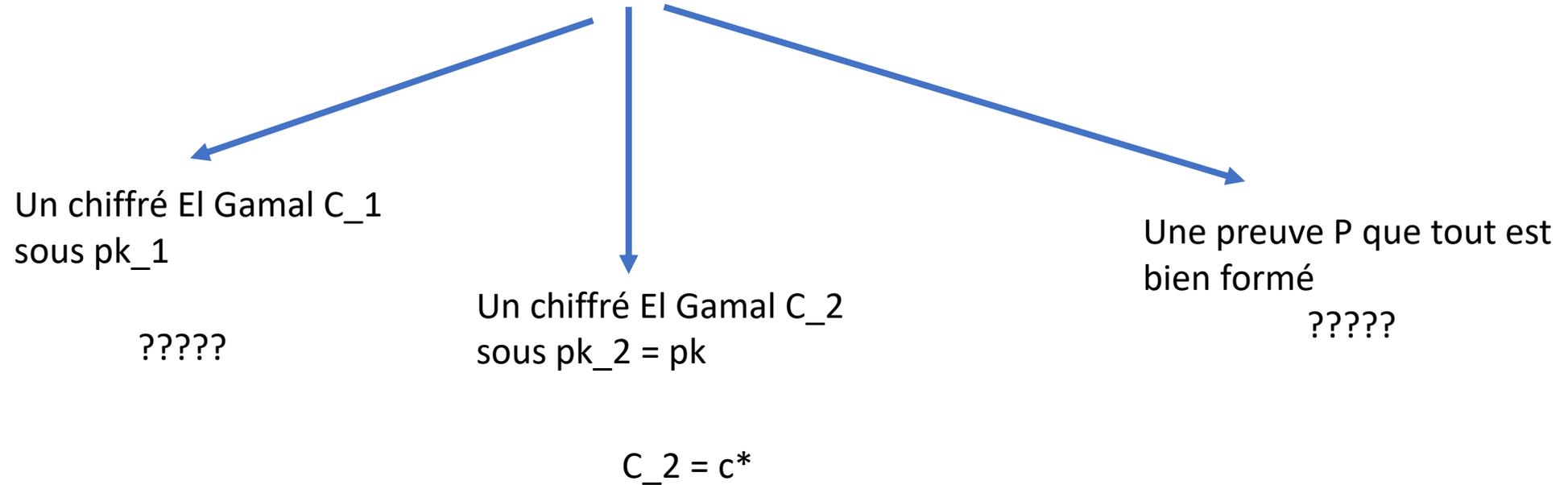
$$C_2 = c^*$$

Difficulté 2: Utiliser le chiffré challenge (IND-CCA) pour distinguer le chiffré challenge c^* IND-CPA

Solution (partielle): Utiliser c^* pour C_2^*

Idée de la preuve

- Un chiffré Naor-Yung est constitué de



Mais comment construire de manière cohérente les autres éléments???

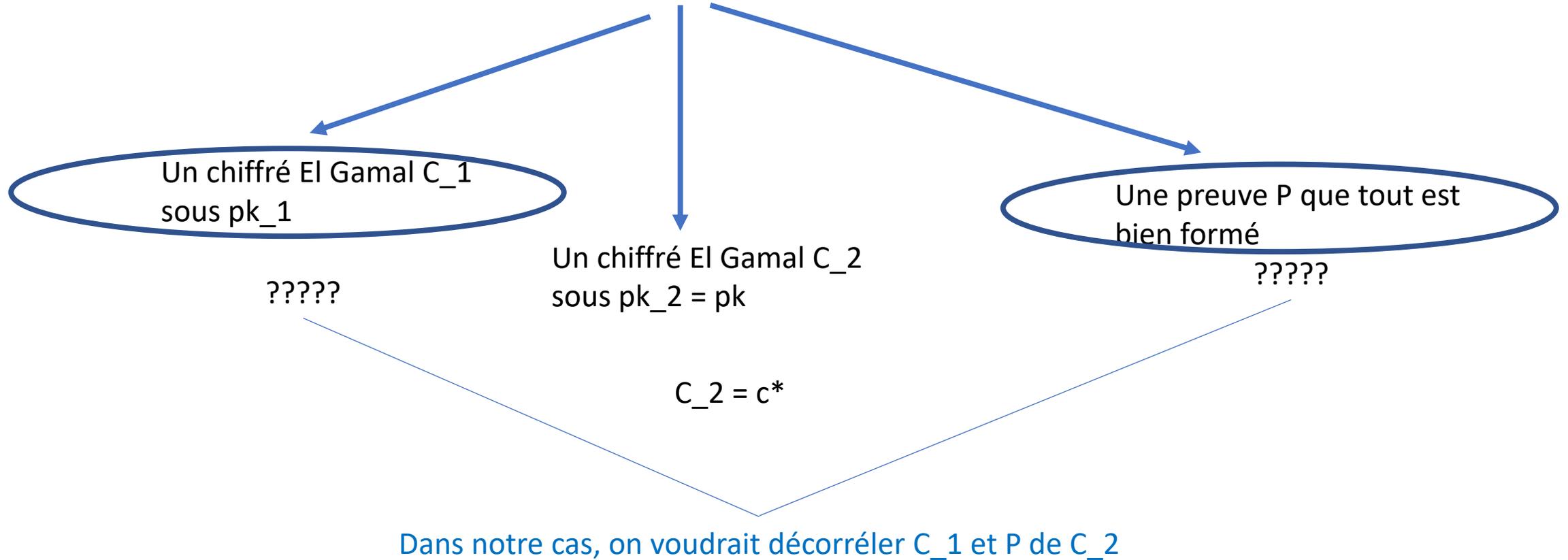
Preuves par Jeux

Le principe d'une preuve par jeux est de modifier progressivement le jeu initial G_0 pour arriver au jeu de notre choix G_n

- Chaque jeu intermédiaire G_i est construit de manière à être indistinguishable du précédent.
- Ainsi, la probabilité de succès d'un adversaire dans le jeu G_n doit rester proche de celui de ce même adversaire dans le jeu initial
- Permet de limiter le nombre d'éléments à changer à chaque tour.

Idée de la preuve

- Un chiffré Naor-Yung est constitué de



Sécurité IND-CCA (Rappel)

- 1) $(sk, pk) \leftarrow \text{Keygen}(1^n)$
- 2) $(m_0, m_1) \leftarrow A^\circ(pk)$
- 3) $C^* = (C_{1^*}, C_{2^*}, P^*) \leftarrow \text{Encrypt}(pk, m_b)$ pour un bit b choisi aléatoirement.
- 4) $b' \leftarrow A^\circ(pk, C^*)$

Jeu IND-CCA –G_0

Preuves par Jeux

- 1) $(sk, pk) \leftarrow \text{Keygen}(1^n)$
- 2) $(m_0, m_1) \leftarrow A^\circ(pk)$
- 3) Pour un bit b choisi aléatoirement:
 $C^*_1 \leftarrow \text{Enc}(pk_1, m_b)$
 $C^*_2 \leftarrow \text{Enc}(pk_2, m_b)$
 P' généré à l'aide du simulateur ZK
 $C^* = (C^*_1, C^*_2, P')$
- 4) $b' \leftarrow A^\circ(pk, C^*)$

On commence par changer P^*

Jeu G_1

Preuves par Jeux

- Il faut alors prouver que G_0 et G_1 sont indistinguables
- Ici l'argument est simple, distinguer ces jeux revient à distinguer une preuve simulée d'une vraie preuve:
- On a $\text{Adv}_1(A) > \text{Adv}_0(A) - \text{Adv}_{\text{ZK}}(A)$, où $\text{Adv}_i(A)$ est l'avantage de A dans le jeu G_i

Concrètement, si le comportement de A diffère selon les jeux, alors A peut casser la propriété de zero-knowledge des preuves

Preuves par Jeux

- 1) $(sk, pk) \leftarrow \text{Keygen}(1^n)$
- 2) $(m_0, m_1) \leftarrow A^\circ(pk)$
- 3) Pour un bit b choisi aléatoirement:
 C^*_1 aléatoire dans G^2
 $C^*_2 \leftarrow \text{Enc}(pk_2, m_b)$
 P' généré à l'aide du simulateur ZK
 $C^* = (C^*_1, C^*_2, P')$
- 4) $b' \leftarrow A^\circ(pk, C^*)$

On change ensuite C^*_1 pour le remplacer par de l'aléa

Jeu G_2

Preuves par Jeux

- Il faut alors prouver que G_1 et G_2 sont indistinguables
- On peut prouver ici que ces deux jeux sont indistinguables sous l'hypothèse DDH
- On a $\text{Adv}_2(A) > \text{Adv}_1(A) - \text{Adv}_{\text{DDH}}(A)$,

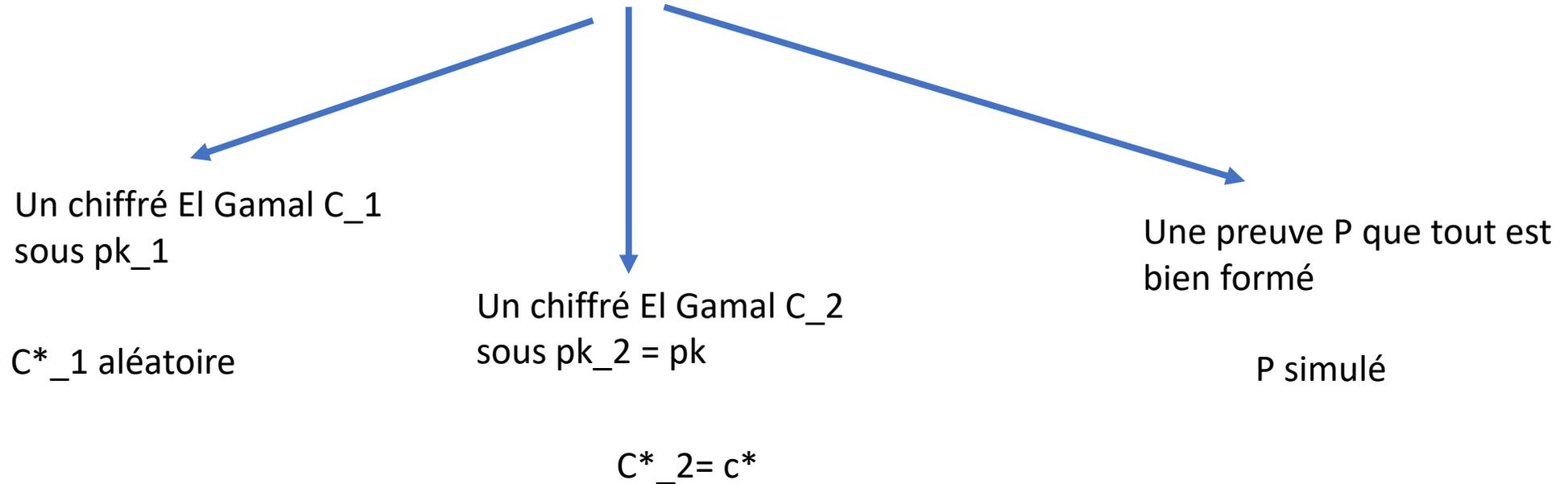
Concrètement, si le comportement de A diffère selon les jeux, alors A peut résoudre DDH

- Au total:

$$\text{Adv}_2(A) > \text{Adv}_1(A) - \text{Adv}_{\text{DDH}}(A) > \text{Adv}_0(A) - \text{Adv}_{\text{DDH}}(A) - \text{Adv}_{\text{ZK}}(A)$$

Idée de la preuve

- Un chiffré Naor-Yung est constitué de

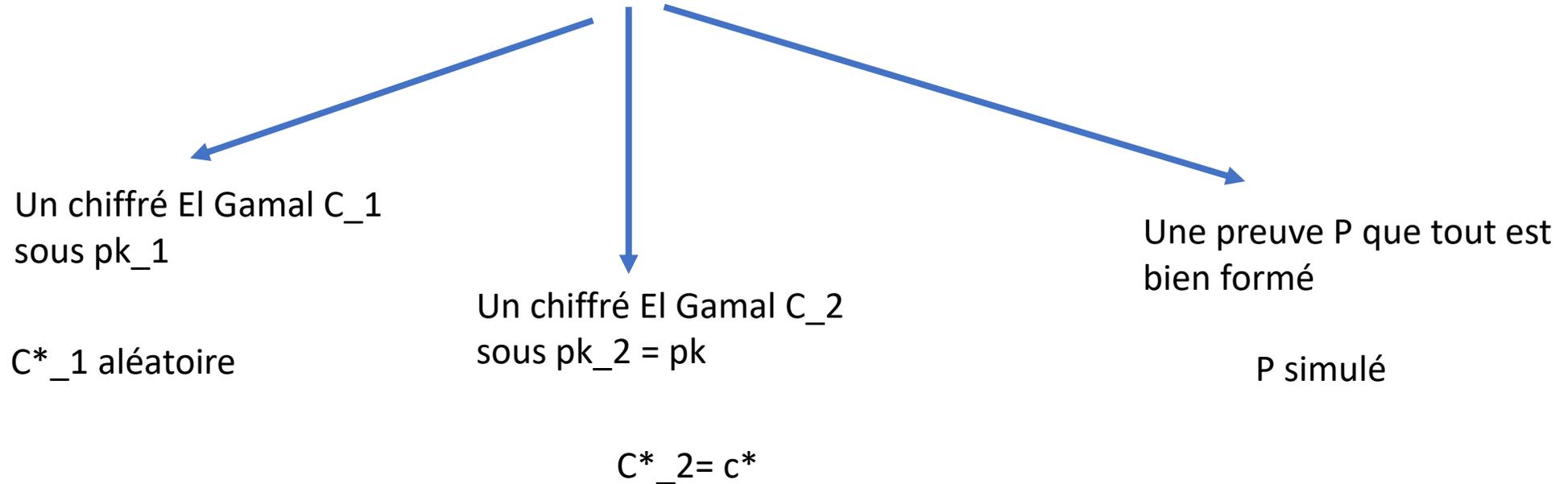


Difficulté 2: Utiliser le chiffré challenge (IND-CCA) pour distinguer le chiffré challenge c^* IND-CPA

Solution : Utiliser c^* pour C_2^* et générer le reste comme dans le jeu G_2

Idée de la preuve

- Un chiffré Naor-Yung est constitué de



On peut alors montrer qu'un adversaire contre le jeu G_2 peut casser l'IND-CPA d'El-Gamal avec même probabilité

Preuves par Jeux

- On avait $\text{Adv}_2(A) > \text{Adv}_1(A) - \text{Adv}_{\text{DDH}}(A) > \text{Adv}_0(A) - \text{Adv}_{\text{DDH}}(A) - \text{Adv}_{\text{ZK}}(A)$
- Si El Gamal IND-CPA alors $\text{Adv}_2(A)$ négligeable:
 - Soit $\text{Adv}_0(A)$ négligeable \rightarrow La construction est IND-CCA
 - Soit $\text{Adv}_0(A)$ non négligeable:
 $\text{Adv}_{\text{DDH}}(A)$ et/ou $\text{Adv}_{\text{ZK}}(A)$ non négligeable également \rightarrow contradiction

La construction est IND-CCA si El Gamal est IND-CPA, DDH est difficile et P est généré à l'aide d'un système de preuves zero-knowledge

