LECTURE 16 2/11

LEMMA: $HYB_{\pi,\mathcal{A}}(\lambda,0) \approx_s HYB_{\pi,\mathcal{A}}(\lambda,1)$

PROOF: The challenge ciphertext looks like $c_1^* = g_1^r = g_3$

$$c_2^* = g_2^{r'} = g_4$$

$$c_3^* = g_3^{x_1} g_4^{y_1} m_b^* \; ; \; c_4^* = g_3^{x_2} g_4^{y_2}$$

where $(g_1, g_2 = g_1^\alpha, g_3 = g_1^r, g_4 = g_2^{r'})$ is a NON-DDH tuple

WLOG, assume $r \neq r'$ and $\alpha \neq 0$

Let $\mathcal{A}$ be UNBOUNDED. By looking at pK, $\mathcal{A}$ knows:

$$h_1 = g_1^{x_1} g_2^{y_1} \rightarrow \log_{g_1} h_1 = x_1 + \alpha y_1 \qquad (*)$$

$\Rightarrow$ All pairs of (x_1, y_1) are possible: $\#q$ pairs.

Let's see what $\mathcal{A}$ learns from decryption queries.

Give $c = (c_1, c_2, c_3, c_4)$ call it LEGAL if

$\exists\, r''$ s.t. $c_1 = g_1^{r''}$ and $c_2 = g_2^{r''}$ i.e. $\log_{g_1} c_1 = \log_{g_2} c_2$

CLAIM: $\mathcal{A}$ obtains additional information on x_1, y_1 only if it makes a decryption query s.t.

i. c is ILLEGAL

ii. $Dec(sK, c) \neq \perp$

PROOF: Clearly, if $Dec(sK, c) = \perp$, the adversary learns $h_2 \neq c_1^{x_2} c_2^{y_2}$. Since x_2, y_2 are independent of x_1, y_1, such a query reveals nothing on x_1, y_1

So, assume $Dec(sK, c) \neq \perp$ and c is LEGAL. This means $\log_{g_1} c_1 = \log_{g_2} c_2 = r''$. The challenger returns

$$m = c_3 / c_1^{x_1} c_2^{y_1}$$

$$\Rightarrow \log_{g_1} m = \log_{g_1} c_3 - x_1 \log_{g_1} c_1 - y_1 \log_{g_1} c_2$$

$$= \log_{g_1} c_3 - x_1 r'' - y_1 \alpha \cdot r'' \quad (**)$$

However, this reveals nothing new on x_1, y_1 because (*) and (**) are linearly DEPENDENT

$$\begin{pmatrix} 1 & \alpha \\ -r'' & -\alpha r'' \end{pmatrix} \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} = \begin{pmatrix} \log_{g_1} h_1 \\ \log_{g_1} m - \log_{g_1} c_3 \end{pmatrix}$$

But

$$\det \begin{pmatrix} 1 & \alpha \\ -r'' & -\alpha r'' \end{pmatrix} = -\alpha r'' + \alpha r'' = 0$$

So, the only case in which A_0 learns something about x_1, y_1 is when, in a decryption query, $Dec(sK, c) \neq \perp$ and c is ILLEGAL ■

CLAIM: So long as A_0 makes poly(λ) decryption queries the probability that it makes a query c that is ILLEGAL and not rejected ($\neq \perp$) is negligible.

PROOF: In order for c to be not rejected we need $c_4 = c_1^{x_2} c_2^{y_2}$. Assume $\log_{g_1} c_1 = r_1 \neq r_2 = \log_{g_1} c_2$

What does A_0 know about x_2, y_2? From pk

$$h_2 = g_1^{x_2} g_2^{y_2} \rightarrow \log_{g_1} h_2 = x_2 + \alpha y_2 \qquad (*')$$

Fix any $c_4 \in G$ s.t. $c_4 = c_1^{x_2} c_2^{y_2}$. This means

$$\log_{g_1} c_4 = x_2 \log_{g_1} c_1 + y_2 \log_{g_1} c_2$$
$$= x_2 \cdot r_1 + y_2 \cdot r_2 \qquad (**')$$

aplhar2

However, $(*')$ and $(**')$ are linearly INDEPENDENT

$$\det \begin{pmatrix} 1 & \alpha \\ r_1 & \alpha r_2 \end{pmatrix} = \alpha r_2 - \alpha r_1 = \alpha (r_2 - r_1) \neq 0$$

$\Rightarrow \exists!$ solution x_2, y_2

So, the probability that A_0 produces such a query c the first time (given pk) is at most $\frac{1}{q}$

Whenever an illegal query is rejected, A_0 learns that $c_4 \neq c_1^{x_2} c_2^{y_2}$ and thus can exclude one pair x_2, y_2. By UNION BOUNDS:

$$\Pr[\exists\, i : (i+1)\text{-th decryption query not rejected}]$$

# of DEC. QUERIES $\rightarrow p(\lambda)$

$$\leq \sum_{i=1}^{p(\lambda)} \Pr[(i+1)\text{-th decryption query not rejected}]$$

$$\leq \sum_{i} \frac{1}{q-1} \leq p(\lambda) \cdot \frac{1}{q - p(\lambda)} \in \text{negl}(\lambda)$$

q-i

■

So, what does A_0 know about x_1, y_1?

By the above claims, except with negl(λ) probability, all it knows after making poly(λ) dec. queries is (*)

Now, we look at $g_3^{x_1} g_4^{y_1}$ which is the "PAD" used to hide m_b^*. What is the probability $g_3^{x_1} g_4^{y_1} = h \in G$?

$$\log_{g_1} h = x_1 \log_{g_1} g_3 + y_1 \log_{g_1} g_4$$
$$= x_1 r + \alpha y_1 r' \qquad (***)$$

Now, E.q. (*) and (***) are linearly independent

$$\det\begin{pmatrix} 1 & \alpha \\ r & \alpha r' \end{pmatrix} = \alpha r' - \alpha r = \alpha(r' - r) \neq 0$$
$$\alpha \neq 0 \qquad r \neq r'$$

For arbitrary h, there is unique solution, so all h are equally likely.

⇒ UNBOUNDEND A_0 can't predict $h = g_3^{x_1} g_4^{y_1}$ w.p. better than $1/q$

⇒ Msg information Theoretically HIDDEN

If h UNIFORM ⇒ $h \cdot m_b^*$ is UNIFORM

⇒ No info about b. ■

Q: Why not CCA-2?

A: Our proof breaks in case of CCA-2. Because given

C^*,

$$C_4^* = (C_1^*)^{x_2} (C_2^*)^{y_2} \rightarrow \log_{g_1} C_4^* = x_2 \log_{g_1} C_1^* + y_2 \log_{g_1} C_2^*$$
$$= x_2 \log_{g_1} g_3 + y_2 \log_{g_1} g_4$$
$$= x_2 r + y_2 \alpha r'$$

Combining this with (*') ⇒ can compute x_2, y_2

Given x_2, y_2 ⇒ can make decryption query that is ILLEGAL not REJECTED

⇝ Given the output, I can also compute x_1, y_1

CRAMER - SHOUP PKE (The real one)

Let $\Pi = (KGen, Enc, Dec)$

$KGen(1^\lambda)$: $params = (G, g_1, g_2, q) \leftarrow \$ \; GroupGen(1^\lambda)$

$x_1, x_2, x_3, y_1, y_2, y_3 \leftarrow \$ \; \mathbb{Z}_q$

$h_1 = g_1^{x_1} g_2^{y_1}$; $h_2 = g_1^{x_2} g_2^{y_2}$; $h_3 = g_1^{x_3} g_2^{y_3}$

$pK = (params, h_1, h_2, h_3)$; $sK = (x_1, x_2, x_3, y_1, y_2, y_3)$

$Enc(pK, m)$: $r \leftarrow \$ \; \mathbb{Z}_q$

$C = (C_1, C_2, C_3, C_4) = (g_1^r, g_2^r, h_1^r \cdot m, (h_2 \cdot h_3^\beta)^r)$

$\beta = H(C_1, C_2, C_3)$ for CRH H (collision resistant hash. func.)

$Dec(sK, c)$: Let $c = (C_1, C_2, C_3, C_4)$, $\beta = H(C_1, C_2, C_3)$

If $C_1^{x_2 + \beta x_3} \cdot C_2^{y_2 + \beta y_3} = C_4$

$$m = C_3 / C_1^{x_1} C_2^{y_1}$$

Else $m = \perp$

CORRECTNESS: $C_4 = (h_2 \cdot h_3^{\beta})^r = h_2^r \cdot h_3^{\beta r}$

$$= (g_1^{x_2} g_2^{y_2})^r \cdot (g_1^{x_3} g_2^{y_3})^{\beta r}$$

$$= (g_1^r)^{x_2 + \beta x_3} \cdot (g_2^r)^{y_2 + \beta y_3}$$

$$= C_1^{x_2 + \beta x_3} \cdot C_2^{y_2 + \beta y_3}$$ ✓

$$C_3 / C_1^{x_1} C_2^{y_1} = \frac{h_1^r \cdot m}{C_1^{x_1} C_2^{y_1}} = \frac{h_1^r \cdot m}{(g_1^r)^{x_1} (g_2^r)^{y_2}} =$$

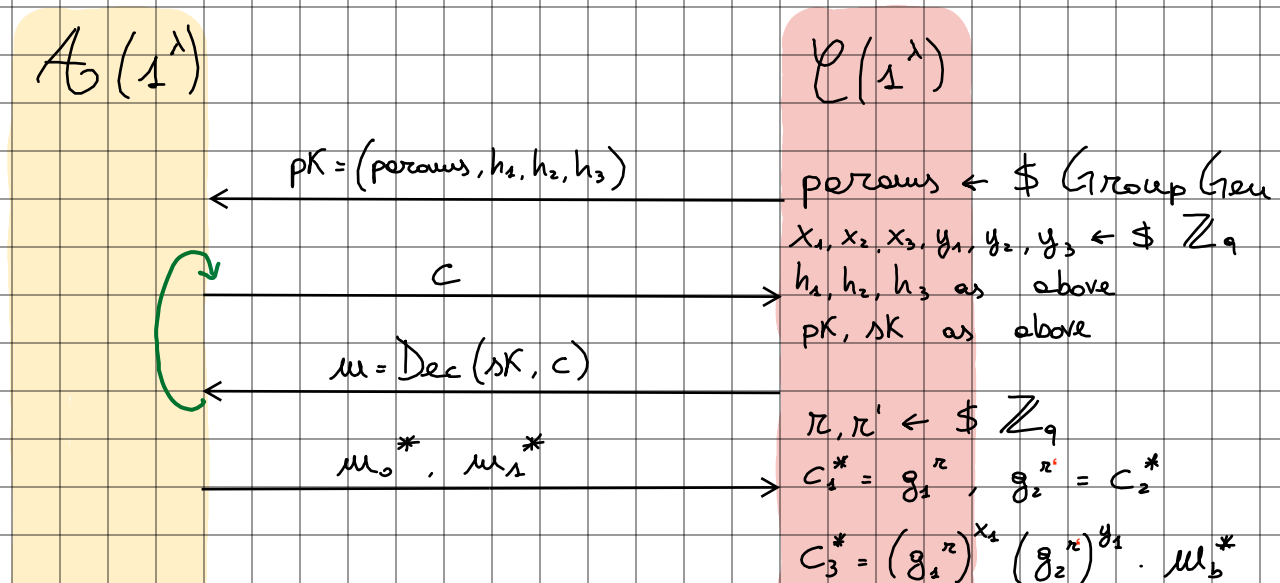
$$= \frac{h_1^r \cdot m}{h_1^r} = m$$ ✓

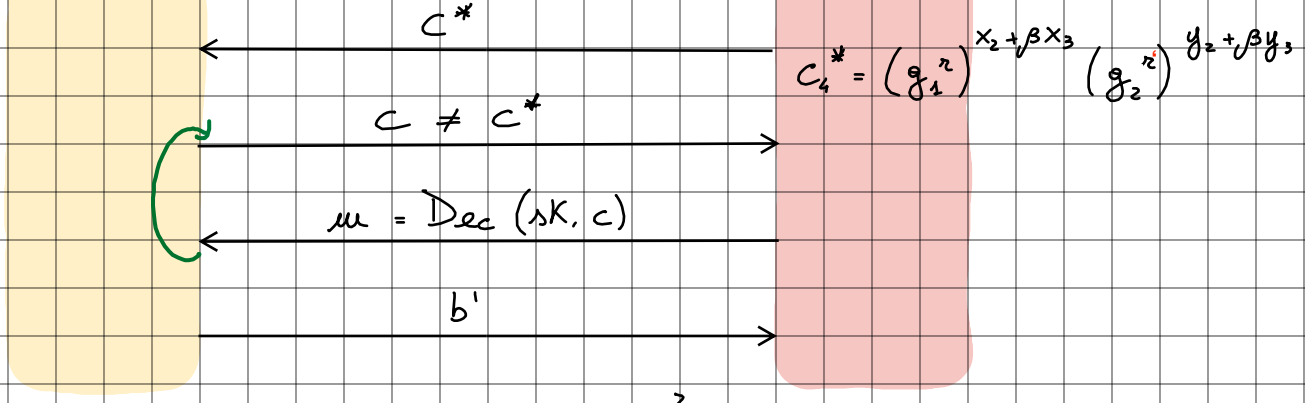
THM: The CS PKE achieves CCA-2 security under DDH

(Note: DDH ⇒ CRH)

PROOF (sketch): We consider similar hybrid as before

$HYB_{\pi, A}(\lambda, b)$





LEMMA: Under DDH: $GAME^{ccar?}_{\pi,A}(\lambda, b) \approx_c HYB_{\pi,A}(\lambda, b)$

PROOF: Left as exercise

LEMMA: $HYB_{\pi,A}(\lambda, 0) \approx_s HYB_{\pi,A}(\lambda, 1)$

To be continued ...