Advanced Cryptography — Final Exam
Solution

Serge Vaudenay

26.6.2019

- duration: 3h
- any document allowed
- a pocket calculator is allowed
- communication devices are not allowed
- the exam invigilators will **not** answer any technical question during the exam
- readability and style of writing will be part of the grade

The exam grade follows a linear scale in which each question has the same weight.

1 Minimal Number of Samples to Distinguish Distributions

We consider two probability distributions P_0 and P_1 over a set $\mathcal{Z}$. We denote by $d(P_0, P_1)$ the *statistical distance* between them, which is

$$d(P_0, P_1) = \frac{1}{2} \sum_{z \in \mathcal{Z}} |P_0(z) - P_1(z)|$$

We also define the *Hellinger distance*

$$H(P_0, P_1) = \sqrt{1 - \sum_{z \in \mathcal{Z}} \sqrt{P_0(z) P_1(z)}}$$

This is a distance in the sense that we always have $H(P_0, P_1) \geq 0$, $H(P_0, P_1) = 0 \iff P_0 = P_1$, and the triangular inequality. We further define the *fidelity*

$$F(P_0, P_1) = 1 - H(P_0, P_1)^2$$

The Fuchs - van de Graaf inequalities relate d and F as follows

$$1 - F(P_0, P_1) \leq d(P_0, P_1) \leq \sqrt{1 - F(P_0, P_1)^2}$$

Given two distributions P and Q, we denote by $P \otimes Q$ the distribution of a pair (X, Y) of independent variables X and Y such that X follows P and Y follows Q. We also denote $P^{\otimes n} = \overbrace{P \otimes \cdots \otimes P}^{n \text{ times}}$.

We are interested in distinguishing the two distributions based on a vector of n i.i.d. samples following one or the other distribution. Given a real number $t \in [0, 1]$, we let n_t be the minimal integer such that there exists a distinguisher using n_t samples with advantage at least t.

Q.1 By using an easy bound on the statistical distance, show that for all t, we have

$$n_t \geq \frac{t}{d(P_0, P_1)}$$

Let $\mathcal{A}$ be a distinguisher using n_t samples with advantage at least t. Due to the link between advantage and statistical distance, we have $\mathsf{Adv}(\mathcal{A}) \leq d(P_0^{\otimes n_t}, P_1^{\otimes n_t})$, where $P^{\otimes n}$ denotes the distribution of a vector of n i.i.d. random variables of distribution P. The easy bound on statistical distance says $d(P_0^{\otimes n}, P_1^{\otimes n}) \leq n \cdot d(P_0, P_1)$. Hence,

$$t \leq \mathsf{Adv}(\mathcal{A}) \leq d(P_0^{\otimes n_t}, P_1^{\otimes n_t}) \leq n_t \cdot d(P_0, P_1)$$

We deduce $n_t \geq \frac{t}{d(P_0,P_1)}$.

Q.2 Prove that $F(P_0^{\otimes n}, P_1^{\otimes n}) = F(P_0, P_1)^n$.
HINT: first prove $F(P_0 \otimes Q_0, P_1 \otimes Q_1) = F(P_0, P_1)F(Q_0, Q_1)$.

We have

$$F(P_0, P_1) = 1 - H(P_0, P_1)^2 = \sum_{z \in \mathcal{Z}} \sqrt{P_0(z)P_1(z)}$$

Hence,

$$\begin{aligned} F(P_0 \otimes Q_0, P_1 \otimes Q_1) &= \sum_{(z_1,z_2) \in \mathcal{Z}_1 \times \mathcal{Z}_2} \sqrt{P_0(z_1)Q_0(z_2)P_1(z_1)Q_1(z_2)} \\ &= \sum_{z_1 \in \mathcal{Z}_1} \sqrt{P_0(z_1)P_1(z_1)} \sum_{z_2 \in \mathcal{Z}_2} \sqrt{Q_0(z_2)Q_1(z_2)} \\ &= F(P_0, P_1)F(Q_0, Q_1) \end{aligned}$$

By induction, we deduce $F(P_0^{\otimes n}, P_1^{\otimes n}) = F(P_0, P_1)^n$.

Q.3 By writing $D_{1/2}(P_0 \| P_1) = -2 \cdot \log_2 F(P_0, P_1)$, prove that

$$n_t \geq \frac{-\log_2(1 - t^2)}{D_{1/2}(P_0 \| P_1)}$$

HINT: use the same technique as in Q.1 but get rid of d.

Using the same technique as Q.1, we have

$$t \leq \mathsf{Adv}(\mathcal{A}) \leq d(P_0^{\otimes n_t}, P_1^{\otimes n_t})$$

We now use the upper bound of d in terms of F to obtain

$$t \leq d(P_0^{\otimes n_t}, P_1^{\otimes n_t}) \leq \sqrt{1 - F(P_0^{\otimes n_t}, P_1^{\otimes n_t})^2}$$

and, with the multiplicativity of F:

$$t \leq \sqrt{1 - F(P_0, P_1)^{2n_t}}$$

Hence

$$n_t \geq \frac{\ln(1-t^2)}{2 \cdot \ln F(P_0, P_1)} = \frac{-\log_2(1-t^2)}{D_{1/2}(P_0 \| P_1)}$$

Q.4 Complete the previous bound by proving

$$\frac{-\log_2(1-t^2)}{D_{1/2}(P_0 \| P_1)} \leq n_t < 1 + \frac{-2 \cdot \log_2(1-t)}{D_{1/2}(P_0 \| P_1)}$$

HINT: use the second Fuchs - van de Graaf inequality.

We take the best distinguisher $\mathcal{B}$ based on $n_t - 1$ samples, we have $\mathsf{Adv}(\mathcal{B}) = d(P_0^{\otimes n_t - 1}, P_1^{\otimes n_t - 1})$ and $\mathsf{Adv}(\mathcal{B}) \leq t$. Hence,

$$t \geq \mathsf{Adv}(\mathcal{B}) = d(P_0^{\otimes n_t - 1}, P_1^{\otimes n_t - 1})$$

We use the lower bound of d in terms of F to obtain

$$t > d(P_0^{\otimes n_t - 1}, P_1^{\otimes n_t - 1}) \geq 1 - F(P_0^{\otimes n_t - 1}, P_1^{\otimes n_t - 1})$$

and, with the multiplicativity of F:

$$t > 1 - F(P_0, P_1)^{n_t - 1}$$

Hence

$$n_t < 1 + \frac{\ln(1-t)}{\ln F(P_0, P_1)} = 1 + \frac{-2 \cdot \log_2(1-t)}{D_{1/2}(P_0 \| P_1)}$$

Q.5 Prove that the minimum number n of samples to distinguish P_0 from P_1 with advantage at least $\frac{1}{2}$ is such that

$$\frac{0.41}{D_{1/2}(P_0 \| P_1)} < n < 1 + \frac{2}{D_{1/2}(P_0 \| P_1)}$$

We apply the previous bound with $t = \frac{1}{2}$ *and see that* $\log_2(1-t) = -1$ *and* $-\log_2(1-t^2) > 0.41$.

2 An IND-CCA Variant of the ElGamal Cryptosytem

This exercise is inspired from Cash-Kiltz-Shoup, The Twin Diffie-Hellman Problem and Applications, *EUROCRYPT 2008, LNCS vol. 4965, Springer.*

Given a key derivation function H and a correct symmetric encryption scheme E/D which can be computed in polynomial time, we define the following cryptosystem:

$\mathsf{Setup}(1^s) \rightarrow \mathsf{pp}$**:** generate a group G and its prime order q and define some public parameters pp from which we can extract s, q, the neutral element 1, a generator g, and parameters to be able to make multiplications in polyomially bounded time in terms of s. We assume that group elements have a unique representation.

$\mathsf{Gen}(\mathsf{pp}) \rightarrow \mathsf{pk}, \mathsf{sk}$**:** pick $x_1, x_2 \in \mathbf{Z}_q$, compute $X_1 = g^{x_1}$, $X_2 = g^{x_2}$, and define $\mathsf{pk} = (\mathsf{pp}, X_1, X_2)$, $\mathsf{sk} = (\mathsf{pp}, x_1, x_2)$.

$\mathsf{Enc}(\mathsf{pk}, m) \rightarrow \mathsf{ct}$**:** pick $y \in \mathbf{Z}_q$, compute $Y = g^y$, $Z_1 = X_1^y$, $Z_2 = X_2^y$, $k = H(Y, Z_1, Z_2)$, $c = E_k(m)$, and define $\mathsf{ct} = (Y, c)$.

$\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) \rightarrow m$**:** [to be defined]

We want to prove the $\mathsf{IND\text{-}CCA}$ security in the random oracle model, which is defined by the following game Γ_b with an adversary $\mathcal{A}$ and the bit b:

```
Game Γ_b
  pick a function H at random
  Setup →$ pp
  Gen(pp) →$ (pk, sk)
  A_1^{OH,ODec_1}(pk) →$ (pt_0, pt_1, st)
  if |pt_0| ≠ |pt_1| then return 0
  ct* ←$ Enc^{OH}(pk, pt_b)
  A_2^{OH,ODec_2}(st, ct*) →$ z
  return z

Oracle OH(input)
  return H(input)

Oracle ODec_1(ct):
  return Dec^{OH}(sk, ct)

Oracle ODec_2(ct):
  if ct = ct* then return ⊥
  return Dec^{OH}(sk, ct)
```

Q.1 Describe the decryption algorithm and prove that we have a correct public-key cryptosystem.

Decryption of ciphertext (Y, c) with secret key (x_1, x_2) works as follows: We compute $Y^{x_1} = Z'_1$, $Y^{x_2} = Z'_2$, $H(Y, Z'_1, Z'_2) = k'$, and finally $D_{k'}(c) = m'$.
Since we can do multiplications in polynomial time, we can exponentiate in polynomial time using the square-and-multiply algorithm. Hence, we have a public-key cryptosystem.
We have $Z'_1 = Y^{x_1} = g^{yx_1} = X_1^y = Z_1$, $Z'_2 = Y^{x_2} = g^{yx_2} = X_2^y = Z_2$, so $k' = H(Y, Z_1, Z_2) = k$, and finally $m' = D_k(c) = m$ due to the correctness of the E/D scheme. Hence, the cryptosystem is correct.

Q.2 Let Γ'_b be the following variant of Γ_b:

Game Γ'_b

1: $\mathsf{Setup} \xrightarrow{\$} \mathsf{pp}$
2: $\mathsf{Gen}(\mathsf{pp}) \xrightarrow{\$} (\mathsf{pk}, \mathsf{sk})$
3: $(\mathsf{pp}, X_1, X_2) \leftarrow \mathsf{pk}$
4: initialize associative array T to empty
5: $\mathcal{A}_1^{\mathsf{OH},\mathsf{ODec}_1}(\mathsf{pk}) \xrightarrow{\$} (\mathsf{pt}_0, \mathsf{pt}_1, \mathsf{st})$
6: **if** $|\mathsf{pt}_0| \neq |\mathsf{pt}_1|$ **then return** 0
7: pick $y^* \in \mathbf{Z}_q$
8: $Y^* \leftarrow g^{y^*}$, $Z_1^* \leftarrow X_1^{y^*}$, $Z_2^* \leftarrow X_2^{y^*}$
9: $k^* \leftarrow \mathsf{OH}(Y^*, Z_1^*, Z_2^*)$
10: $c^* \leftarrow E_{k^*}(\mathsf{pt}_b)$
11: $\mathsf{ct}^* \leftarrow (Y^*, c^*)$
12: $\mathcal{A}_2^{\mathsf{OH},\mathsf{ODec}_2}(\mathsf{st}, \mathsf{ct}^*) \xrightarrow{\$} z$
13: **return** z

Oracle $\mathsf{OH}(\mathsf{input})$

1: **if** $T(\mathsf{input})$ is not defined **then**
2: pick $T(\mathsf{input})$ at random
3: **end if**
4: **return** $T(\mathsf{input})$

Oracle $\mathsf{ODec}_1(\mathsf{ct})$:

5: **return** $\mathsf{Dec}^{\mathsf{OH}}(\mathsf{sk}, \mathsf{ct})$

Oracle $\mathsf{ODec}_2(\mathsf{ct})$:

6: $(Y, c) \leftarrow \mathsf{ct}$
7: **if** $(Y, c) = \mathsf{ct}^*$ **then return** $\perp$
8: **if** $Y = Y^*$ **then return** $D_{k^*}(c)$
9: **return** $\mathsf{Dec}^{\mathsf{OH}}(\mathsf{sk}, \mathsf{ct})$

Prove that $\Pr[\Gamma_b \to 1] = \Pr[\Gamma'_b \to 1]$ for all b.

The difference between Γ_b and Γ'_b is in
– *expanding* Enc *in the game to define the variables Y^* and k^*;*
– *the simulation of* OH *by the lazy sampling technique;*
– *Step 8 of* ODec_2.
All those changes induce no behavior modification. These are bridging steps.

Q.3 Let Γ''_b be a variant of Γ'_b in which Step 9 of the game is replaced by

9: pick k^* at random

We define the failure event F that OH is queried with input (Y^*, Z_1^*, Z_2^*) in Γ'_b at some time during the game except on Step 9. Prove that $|\Pr[\Gamma'_b \to 1] - \Pr[\Gamma''_b \to 1]| \leq \Pr[F]$.

The difference between Γ'_b and Γ''_b is that T is not used any more in Step 9. Hence, $T(Y^*, Z_1^*, Z_2^*)$ is neither set nor checked. If F never occurs, $T(Y^*, Z_1^*, Z_2^*)$ is never used anywhere else. This, it is the same to query H with (Y^*, Z_1^*, Z_2^*) and to pick a random k^*. Hence, Γ'_b and Γ''_b are identical when F does not occur. Due to the difference lemma, we obtain $|\Pr[\Gamma'_b \to 1] - \Pr[\Gamma''_b \to 1]| \leq \Pr[F]$.

Q.4 We say that E/D is secure if for any PPT algorithm $\mathcal{B}$, the advantage

$$\mathsf{Adv}_{\mathcal{B}} = \Pr[\Gamma_1^* \to 1] - \Pr[\Gamma_0^* \to 1]$$

is negligible, with Γ_b^* defined as follows:

Game Γ_b^*

1: $\mathcal{B}_1() \xrightarrow{\$} (m_0, m_1, \mathsf{st})$
2: **if** $|m_0| \neq |m_1|$ **then return** 0
3: pick a random key k^*
4: $c^* \leftarrow E_{k^*}(m_b)$
5: $\mathcal{B}_2^{\mathsf{OD}}(\mathsf{st}, c^*) \xrightarrow{\$} z$
6: **return** z

Oracle $\mathsf{OD}(c)$:

1: **if** $c = c^*$ **then return** $\perp$
2: **return** $D_{k^*}(c)$

Prove that if E/D is secure, then $\Pr[\Gamma''_1 \to 1] - \Pr[\Gamma''_0 \to 1]$ is negligible.

Given the adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ playing in Γ''_0 and Γ''_1, we construct an adversary $\mathcal{B} = (\mathcal{B}_1, \mathcal{B}_2)$ playing in Γ^*_0 and Γ^*_1.

$\mathcal{B}_1$:
1: *simulate $\Gamma''_{\mathcal{A}}$ but stop before Step 9*
2: *set* $\mathsf{st}' = (\mathsf{st}, \mathsf{sk}, T, Y^*)$
3: ***return*** $(\mathsf{pt}_0, \mathsf{pt}_1, \mathsf{st}')$

$\mathcal{B}_2(\mathsf{st}', c^*)$:
1: $\mathsf{st}' \to (\mathsf{st}, \mathsf{sk}, T, Y^*)$
2: $\mathsf{ct}^* \leftarrow (Y^*, c^*)$
3: *simulate $\mathcal{A}_2(\mathsf{st}, \mathsf{ct}^*) \to z$ with oracles* OH *and* ODec_2 *with a modification in oracle* ODec_2*: replace $D_{k^*}(c)$ in Step 8 by an oracle call* $\mathsf{OD}(c)$ *to get the result*
4: ***return*** z

Clearly, the simulation is perfect (in the sense that Γ^*_b is obtained from Γ''_b by a sequence of bridging steps) and we have $\Pr[\Gamma''_b \to 1] = \Pr[\Gamma^*_b \to 1]$. We apply the security of E/D to obtain the result.

Q.5 We consider the game Γ'_b from Q.2 and the event F from Q.3. We consider a variant $\overline{\Gamma}_b$ of Γ'_b as follows:

Game $\overline{\Gamma}_b$
1: $\mathsf{Setup} \xrightarrow{\$} \mathsf{pp}$
2: $\mathsf{Gen}(\mathsf{pp}) \xrightarrow{\$} (\mathsf{pk}, \mathsf{sk})$
3: $(\mathsf{pp}, X_1, X_2) \leftarrow \mathsf{pk}$, $(\mathsf{pp}, x_1, x_2) \leftarrow \mathsf{sk}$
4: initialize associative arrays Good and T to empty
5: $\mathcal{A}_1^{\mathsf{OH},\mathsf{ODec}_1}(\mathsf{pk}) \xrightarrow{\$} (\mathsf{pt}_0, \mathsf{pt}_1, \mathsf{st})$
6: **if** $|\mathsf{pt}_0| \neq |\mathsf{pt}_1|$ **then return** 0
7: pick $y^* \in \mathbf{Z}_q$
8: $Y^* \leftarrow g^{y^*}$, $Z_1^* \leftarrow X_1^{y^*}$, $Z_2^* \leftarrow X_2^{y^*}$
9: $k^* \leftarrow \mathsf{OH}(Y^*, Z_1^*, Z_2^*)$
10: $c^* \leftarrow E_{k^*}(\mathsf{pt}_b)$
11: $\mathsf{ct}^* \leftarrow (Y^*, c^*)$
12: $\mathcal{A}_2^{\mathsf{OH},\mathsf{ODec}_2}(\mathsf{st}, \mathsf{ct}^*) \xrightarrow{\$} z$
13: **return** z

Oracle $\mathsf{OH}(\mathsf{input})$
1: $(Y, Z_1, Z_2) \leftarrow \mathsf{input}$
2: **if** $Z_1 = Y^{x_1}$ and $Z_2 = Y^{x_2}$ **then**
3:     **if** $\mathsf{Good}(Y)$ undefined **then**
4:         pick $\mathsf{Good}(Y)$ at random
5:     **end if**
6:     **return** $\mathsf{Good}(Y)$
7: **else**
8:     **if** $T(\mathsf{input})$ is not defined **then**
9:         pick $T(\mathsf{input})$ at random
10:     **end if**
11:     **return** $T(\mathsf{input})$
12: **end if**

Oracle $\mathsf{ODec}_1(\mathsf{ct})$:
13: **return** $\mathsf{Dec}^{\mathsf{OH}}(\mathsf{sk}, \mathsf{ct})$

Oracle $\mathsf{ODec}_2(\mathsf{ct})$:
14: $(Y, c) \leftarrow \mathsf{ct}$
15: **if** $(Y, c) = \mathsf{ct}^*$ **then return** $\perp$
16: **if** $Y = Y^*$ **then return** $D_{k^*}(c)$
17: **return** $\mathsf{Dec}^{\mathsf{OH}}(\mathsf{sk}, \mathsf{ct})$

We define the event $\overline{F}$ in $\overline{\Gamma}_b$ as the event F in Γ'_b. Prove that $\Pr[\overline{\Gamma}_b \to 1] = \Pr[\Gamma'_b \to 1]$ and that $\Pr[F] = \Pr[\overline{F}]$.

The only change is in setting up a new array Good *and in a new* OH *oracle. We can see that* OH *only treats differently the inputs (Y, Z_1, Z_2) of the form (Y, Y^{x_1}, Y^{x_2}). For each Y, there is one and only one triplet of this form. It does not matter if we store the output k in T or in* Good*. Hence,* OH *implements a random oracle as well.*

Q.6 We define the Strong Twin Diffie-Hellman game as follows:

Game STDH:
$\mathsf{Setup} \xrightarrow{\$} \mathsf{pp}$
pick $x_1, x_2 \in \mathbf{Z}_q$
$X_1 \leftarrow g^{x_1}$, $X_2 \leftarrow g^{x_2}$
pick $y^* \in \mathbf{Z}_q$
$Y^* \leftarrow g^{y^*}$, $Z_1^* \leftarrow X_1^{y^*}$, $Z_2^* \leftarrow X_2^{y^*}$
$\mathcal{C}^{\mathsf{ODTDH}}(\mathsf{pp}, X_1, X_2, Y^*) \xrightarrow{\$} (Z_1, Z_2)$
return $1_{Z_1 = Z_1^*, Z_2 = Z_2^*}$

Oracle $\mathsf{ODTDH}(Y, Z_1, Z_2)$:
return $1_{Z_1 = Y^{x_1} \wedge Z_2 = Y^{x_2}}$

We consider the game $\overline{\Gamma}_b$ and the event $\overline{F}$. Given an adversary $\mathcal{A}$ playing the $\overline{\Gamma}_b$ game, construct an adversary $\mathcal{C}$ playing the STDH game such that

$$\Pr[\overline{F}] = \Pr[\mathsf{STDH}_{\mathcal{C}} \to 1]$$

HINT: find a way to simulate $\overline{\Gamma}_b$ without sk.

We define $\mathcal{C}$ by simulating the game Γ_b' until the solution is found.

$\mathcal{C}_i(\mathsf{pp}, X_1, X_2, Y^*)$
1: $\mathsf{pk} \leftarrow (\mathsf{pp}, X_1, X_2)$
2: $\mathsf{Result} \leftarrow \perp$
3: *simulate* $\overline{\Gamma}_b$ *from Step 4*
- *use* $\mathsf{OD}(\mathsf{ct})$ *at the place of* $\mathsf{Dec}^{\mathsf{OH}}(\mathsf{sk}, \mathsf{ct})$
- *use a new* OH

4: ***return*** Result

Oracle $\mathsf{OD}(\mathsf{ct})$:
5: $(Y, c) \leftarrow \mathsf{ct}$
6: ***if*** $\mathsf{Good}(Y)$ *undefined* ***then***
7: *pick* $\mathsf{Good}(Y)$ *at random*
8: ***end if***
9: $\mathsf{Good}(Y) \to k$
10: ***return*** $D_k(c)$

Oracle $\mathsf{OH}(\mathsf{input})$
1: $(Y, Z_1, Z_2) \leftarrow \mathsf{input}$
2: ***if*** $\mathsf{ODTDH}(Y, Z_1, Z_2) = 1$ ***then***
3: ***if*** $Y = Y^*$ ***then*** $\mathsf{Result} \leftarrow (Z_1, Z_2)$
4: ***if*** $\mathsf{Good}(Y)$ *undefined* ***then***
5: *pick* $\mathsf{Good}(Y)$ *at random*
6: ***end if***
7: ***return*** $\mathsf{Good}(Y)$
8: ***else***
9: ***if*** $T(\mathsf{input})$ *is not defined* ***then***
10: *pick* $T(\mathsf{input})$ *at random*
11: ***end if***
12: ***return*** $T(\mathsf{input})$
13: ***end if***

The only change in the simulation is that Dec *is simulated without knowing* sk *by using the* Good *array. There are also two changes in* OH*:*
- *the test of Step 2 is simulated by* $\mathsf{ODTDH}(Y, Z_1, Z_2) = 1$*, which is a perfect simulation without knowing* sk*.*
- *the extra Step 3 stores something in* Result *which was not used before.*

The simulation is perfect. Hence, the game $\overline{\Gamma}_b$ *executes the same. When* $\overline{F}$ *happens, we can see in* OH *that the* (Z_1, Z_2) *value corresponding to* Y^* *is stored in* Result*. As a matter of fact, this is precisely the answer to the* STDH *problem. Hence,* $\Pr[\overline{F}] = \Pr[\mathsf{STDH}_{\mathcal{C}} \to 1]$*.*

Q.7 Summarize all what we did and prove that the cryptosystem is $\mathsf{IND\text{-}CCA}$ secure in the random oracle model, under the assumption that the strong twin Diffie-Hellman problem STDH is hard and that the E/D scheme is secure.
NOTE: in a twin exercise, we show STDH is equivalent to CDH.

We have

- *for all* $b \in \{0,1\}$, $\Pr[\mathit{\Gamma}_b \to 1] = \Pr[\mathit{\Gamma}'_b \to 1]$,
- *for all* $b \in \{0,1\}$, $|\Pr[\mathit{\Gamma}'_b \to 1] - \Pr[\mathit{\Gamma}''_b \to 1]| \leq \Pr[F]$,
- $\Pr[F] = \Pr[\overline{F}]$,
- $\Pr[\overline{F}] = \Pr[\mathsf{STDH} \to 1]$,
- $|\Pr[\mathit{\Gamma}''_1 \to 1] - \Pr[\mathit{\Gamma}''_0 \to 1]| \leq |\Pr[\mathit{\Gamma}^*_1 \to 1] - \Pr[\mathit{\Gamma}^*_0 \to 1]|$.

Hence,

$$|\Pr[\mathit{\Gamma}_1 \to 1] - \Pr[\mathit{\Gamma}_0 \to 1]| \leq 2\Pr[\mathsf{STDH} \to 1] + |\Pr[\mathit{\Gamma}^*_1 \to 1] - \Pr[\mathit{\Gamma}^*_0 \to 1]|$$

which is negligible, assuming that the strong twin Diffie-Hellman problem is hard and that E/D *is secure. This means that the cryptosystem is* IND-CCA *secure.*

3 Equivalence of CDH and the Strong Twin DH Problems

Note: this is a twin exercise of "An IND-CCA Variant of the ElGamal Cryptosystem". However, both exercises are totally independent.

This exercise is inspired from Cash-Kiltz-Shoup, The Twin Diffie-Hellman Problem and Applications, *EUROCRYPT 2008, LNCS vol. 4965, Springer.*

We define the Strong Twin Diffie-Hellman STDH game and the classical CDH game as follows:

Game STDH:
1: $\mathsf{Setup} \xrightarrow{\$} \mathsf{pp}$
2: pick $x_1, x_2 \in \mathbf{Z}_q$
3: $X_1 \leftarrow g^{x_1}$, $X_2 \leftarrow g^{x_2}$
4: pick $y^* \in \mathbf{Z}_q$
5: $Y^* \leftarrow g^{y^*}$, $Z_1^* \leftarrow X_1^{y^*}$, $Z_2^* \leftarrow X_2^{y^*}$
6: $\mathcal{A}^{\mathsf{ODTDH}}(\mathsf{pp}, X_1, X_2, Y^*) \xrightarrow{\$} (Z_1, Z_2)$
7: **return** $1_{Z_1=Z_1^*, Z_2=Z_2^*}$

Oracle $\mathsf{ODTDH}(Y, Z_1, Z_2)$:
8: **return** $1_{Z_1=Y^{x_1} \wedge Z_2=Y^{x_2}}$

Game CDH
1: $\mathsf{Setup} \xrightarrow{\$} \mathsf{pp}$
2: pick $x, y \in \mathbf{Z}_q$
3: $X \leftarrow g^x$, $Y \leftarrow g^y$
4: $\mathcal{B}(\mathsf{pp}, X, Y) \xrightarrow{\$} Z$
5: **return** $1_{Z=Y^x}$

Our goal is to prove the equivalence between the two problems.

Here, $\mathsf{Setup}(1^s) \to \mathsf{pp}$ is an algorithm which generates a group G and its prime order q in some public parameters pp. Given pp, we can extract q, the neutral element 1, a generator g, and parameters to be able to make multiplications in polyomially bounded time. We assume that group elements have a unique representation.

Q.1 Given an adversary $\mathcal{B}$ playing the CDH game, construct and adversary $\mathcal{A}$ playing the STDH game such that $\Pr[\mathsf{STDH} \to 1] \geq \Pr[\mathsf{CDH} \to 1]^2$.

$\mathcal{A}(\mathsf{pp}, X_1, X_2, Y^*)$:
1: pick $r \in \mathbf{Z}_q$
2: $\mathcal{B}(\mathsf{pp}, X_1, Y^*) \xrightarrow{\$} Z_1$
3: $\mathcal{B}(\mathsf{pp}, X_2, Y^* g^r) \xrightarrow{\$} Z$
4: $Z_2 \leftarrow Y^* X_2^{-r}$
5: ***return*** (Z_1, Z_2)

The uniform $r \in \mathbf{Z}_q$ *separates the two runs of* $\mathcal{B}$ *which become independent, but for* pp*. If* p_{pp} *is the probability that* CDH *yields 1 conditioned to* pp*, then the same probability for* STDH *is* p_{pp}^2*. Hence, the probability that* STDH *succeeds is* $E(p_{\mathsf{pp}}^2)$*. Thanks to the Jensen inequality, this is greater than* $E(p_{\mathsf{pp}})^2$*. Hence,* $\Pr[\mathsf{STDH} \to 1] \geq \Pr[\mathsf{CDH} \to 1]^2$*.*

Q.2 We define the following random variables: $x, u, v, y, z_1, z_2 \in \mathbf{Z}_q$, $x_1 = x$, and $x_2 = v - xu \bmod q$. We assume that (x, u, v) is uniformly distributed in $\mathbf{Z}_q^3$ and that $(y, z_1, z_2) = f(x_1, x_2)$ for some function f.

Q.2a Prove that (x_1, x_2, u) is uniformly distributed in $\mathbf{Z}_q^3$.

The function mapping (x, u, v) to (x_1, x_2, u) is $(x, u, v) \mapsto (x, v - xu, u)$ which is a permutation of $\mathbf{Z}_q^3$. Hence, (x_1, x_2, u) is also uniform.

Q.2b Prove that

$$\Pr[z_1 u + z_2 = yv | z_1 = yx_1, z_2 = yx_2] = 1 \quad , \quad \Pr[z_1 u + z_2 = yv | z_1 \neq yx_1 \vee z_2 \neq yx_2] \leq \frac{1}{q}$$

(where equalities are modulo q).

$z_1 u + z_2 = yv$ is equivalent to

$$(z_1 - yx_1)u + (z_2 - yx_2) = 0$$

Hence, the first equation is quite clear. For the second we recall that x_1, x_2, u are independent and that (y, z_1, z_2) is a function of x_1, x_2. Hence, u is independent from all the rest. For any values of x_1, x_2 giving $z_1 \neq yx_1$, the probability over u is $\frac{1}{q}$. For any values of x_1, x_2 giving $z_1 = yx_1$ and $z_2 \neq yx_2$, the probability over u is 0. Hence, for any values of x_1, x_2 giving $z_1 \neq yx_1 \vee z_2 \neq yx_2$, the probability over u is at most $\frac{1}{q}$.

Q.3 Given an adversary $\mathcal{A}$ playing the STDH game, prove that the following $\mathcal{B}$ playing the CDH game is such that $\Pr[\mathsf{CDH} \to 1] \geq \Pr[\mathsf{STDH} \to 1] - \frac{Q}{q}$ where Q is the total number of queries of $\mathcal{A}$.

$\mathcal{B}(\mathsf{pp}, X, Y)$:
pick $u, v \in \mathbf{Z}_q$
$X_1 \leftarrow X$, $X_2 \leftarrow g^v X^{-u}$
simulate $\mathcal{A}(\mathsf{pp}, X_1, X_2, Y) \xrightarrow{\$} (Z_1, Z_2)$
with oracle O instead of ODTDH
return Z_1

Oracle $\mathsf{O}(\hat{Y}, \hat{Z}_1, \hat{Z}_2)$
return $1_{\hat{Z}_1^u \hat{Z}_2 = \hat{Y}^v}$

Let x be the discrete logarithm of X, $x_1 = x$, and $x_2 = v - xu$. The random variables x, r, s are uniform and independent. Let E_i be the event that the ith query to O returns 1 but that either $\hat{Z}_1 \neq \hat{Y}^{x_1}$ or $\hat{Z}_2 \neq \hat{Y}^{x_2}$. Thanks to the previous question, we have $\Pr[E_i] \leq \frac{1}{q}$. Hence, the probability that at least one out of the Q total number of queries produce this failure event is bounded by $\frac{Q}{q}$. Except in this failure case, the simulation is perfect. Hence, using the difference lemma, we obtain the result.