Foundations Of Cryptography (fall 2017/2018) - solution for problem set 1

Omri Shmueli

note: I had no collaborators and used no external resources.

Solution 1: Let (E, D) be an encryption scheme for messages of length l and keys of length $n < l-6$, such that for any message $m \in \{0,1\}^\ell$, E tosses at most n coins when encrypting m with a secret key sk. that is, the encryption of m is drawn from the distribution $\{ct \mid sk \leftarrow \{0,1\}^n, r \leftarrow \{0,1\}^n, ct = E_{sk}(m;r)\}$. we will show there exist two messages m_0, m_1 and an inefficient A such that:

$$\Pr\left[\begin{array}{l|l} A(ct) = m_b & \begin{array}{l} sk \leftarrow \{0,1\}^n \\ b \leftarrow \{0,1\} \\ r \leftarrow \{0,1\}^n \\ ct = E_{sk}(m_b;r) \end{array} \end{array}\right] > 0.99 \ .$$

fix arbitrarily a message $m_0 \in \{0,1\}^\ell$, and note that for any ciphertext ct from the distribution of m_0 (which means, there exists a key $sk \in U_n$ and coin toss $r \in U_n$ such that: $E_{sk}(m_0;r) = ct$):

$$\Pr\left[\exists sk \in U_n : D_{sk}(ct) = m \;\middle|\; m \leftarrow \{0,1\}^l\right] \leq \frac{2^n}{2^l} < 2^{-6}$$

the reason for this is that for each ciphertext ct, for each key $sk \in U_n$ there is **at most one message** $m \in \{0,1\}^l$ that sk and ct are mapped to: $D_{sk}(ct) = m$, as we assume D is deterministic. and since the sample space is of size 2^l we get the inequality.

the last inequality holds for every ciphertext, therefore it also holds for the expectation of the probability over the ciphertexts:

$$\mathop{\mathbf{E}}_{\substack{sk \leftarrow U_n \\ r \leftarrow U_n \\ ct = E_{sk}(m_0;r)}} \left[\Pr\left[\exists sk \in U_n : D_{sk}(ct) = m \;\middle|\; m \leftarrow \{0,1\}^l\right]\right] < 2^{-6}$$

now, observe that:

$$\mathop{\mathbf{E}}_{\substack{sk \leftarrow U_n \\ r \leftarrow U_n \\ ct = E_{sk}(m_0;r)}} \left[\Pr\left[\exists sk \in U_n : D_{sk}(ct) = m \;\middle|\; m \leftarrow \{0,1\}^l\right]\right] =$$

$$\Pr\left[\exists sk' \in U_n : D_{sk'}(ct) = m \;\middle|\; \begin{array}{l} m \leftarrow \{0,1\}^l \\ sk \leftarrow U_n \\ r \leftarrow U_n \\ ct = E_{sk}(m_0; r) \end{array}\right] =$$

$$\mathbf{E}_{m \leftarrow \{0,1\}^l}\left[\Pr\left[\exists sk' \in U_n : D_{sk'}(ct) = m \;\middle|\; \begin{array}{l} sk \leftarrow U_n \\ r \leftarrow U_n \\ ct = E_{sk}(m_0; r) \end{array}\right]\right] .$$

the fact that

$$\mathbf{E}_{m \leftarrow \{0,1\}^l}\left[\Pr\left[\exists sk' \in U_n : D_{sk'}(ct) = m \;\middle|\; \begin{array}{l} sk \leftarrow U_n \\ r \leftarrow U_n \\ ct = E_{sk}(m_0; r) \end{array}\right]\right] < 2^{-6}$$

implies the existence of a message $m_1 \in \{0,1\}^l$ such that:

$$Pr\left[\exists sk' \in U_n : D_{sk'}(ct) = m_1 \;\middle|\; \begin{array}{l} sk \leftarrow U_n \\ r \leftarrow U_n \\ ct = E_{sk}(m_0; r) \end{array}\right] < 2^{-6} .$$

now we can describe our attacker A: for a ciphertext input ct, A will go over all possible keys $sk \in \{0,1\}^n$ and run $D_{sk}(ct)$. if there was a key $sk \in \{0,1\}^n$ such that: $D_{sk}(ct) = m_1$, output m_1, otherwise output m_0. we will now calculate the probability of error:

$$\Pr\left[A(ct) = m_b \;\middle|\; \begin{array}{l} sk \leftarrow \{0,1\}^n \\ b \leftarrow \{0,1\} \\ r \leftarrow \{0,1\}^n \\ ct = E_{sk}(m_b; r) \end{array}\right] =$$

$$\Pr[b=0]\Pr\left[A(ct) = m_0 \;\middle|\; \begin{array}{l} sk \leftarrow \{0,1\}^n \\ r \leftarrow \{0,1\}^n \\ ct = E_{sk}(m_0; r) \end{array}\right] + \Pr[b=1]\Pr\left[A(ct) = m_1 \;\middle|\; \begin{array}{l} sk \leftarrow \{0,1\}^n \\ r \leftarrow \{0,1\}^n \\ ct = E_{sk}(m_1; r) \end{array}\right] =$$

$$\frac{1}{2}\left(\Pr\left[A(ct) = m_0 \;\middle|\; \begin{array}{l} sk \leftarrow \{0,1\}^n \\ r \leftarrow \{0,1\}^n \\ ct = E_{sk}(m_0; r) \end{array}\right] + \Pr\left[A(ct) = m_1 \;\middle|\; \begin{array}{l} sk \leftarrow \{0,1\}^n \\ r \leftarrow \{0,1\}^n \\ ct = E_{sk}(m_1; r) \end{array}\right]\right) .$$

recall that $A(ct) = m_1 \iff \exists sk' \in U_n : D_{sk'}(ct) = m_1$, and therefore:

$$\Pr\left[A(ct) = m_1 \;\middle|\; \begin{array}{l} sk \leftarrow \{0,1\}^n \\ r \leftarrow \{0,1\}^n \\ ct = E_{sk}(m_1; r) \end{array}\right] = 1$$

and for the probability of the second term:

$$\Pr\left[A(ct) = m_0 \;\middle|\; \begin{array}{l} sk \leftarrow \{0,1\}^n \\ r \leftarrow \{0,1\}^n \\ ct = E_{sk}(m_0; r) \end{array}\right] =$$

$$1 - \Pr\left[\exists sk' \in U_n : D_{sk'}(ct) = m_1 \;\middle|\; \begin{array}{l} sk \leftarrow \{0,1\}^n \\ r \leftarrow \{0,1\}^n \\ ct = E_{sk}(m_0; r) \end{array}\right] > 1 - 2^{-6}$$

and we got:

$$\Pr\left[A(ct) = m_b \;\middle|\; \begin{array}{l} sk \leftarrow \{0,1\}^n \\ b \leftarrow \{0,1\} \\ r \leftarrow \{0,1\}^n \\ ct = E_{sk}(m_b; r) \end{array}\right] > \frac{1}{2}\left(1 - 2^{-6} + 1\right) = 1 - 2^{-7} > 0.99$$

Q.E.D

Solution 2.a: we need to prove that for any algorithm A and distributions X_0, X_1 over the same support S:

$$\left|\Pr\left[A(x) = b \;\middle|\; \begin{array}{l} b \leftarrow \{0,1\} \\ x \leftarrow X_b \end{array}\right] - \frac{1}{2}\right| = \frac{\Delta_A(X_0, X_1)}{2} \quad .$$

note that

$$\Pr\left[A(x) = b \;\middle|\; \begin{array}{l} b \leftarrow \{0,1\} \\ x \leftarrow X_b \end{array}\right] = \Pr[b=0]\Pr[A(x)=0|x \leftarrow X_0] + \Pr[b=1]\Pr[A(x)=1|x \leftarrow X_1] =$$

$$\frac{1}{2}\left(\Pr[A(x)=0|x \leftarrow X_0] + \Pr[A(x)=1|x \leftarrow X_1]\right) =$$

$$\frac{1}{2}\left(1 - \Pr[A(x)=1|x \leftarrow X_0] + \Pr[A(x)=1|x \leftarrow X_1]\right) =$$

$$\frac{1}{2} + \frac{1}{2}\left(\Pr[A(x)=1|x \leftarrow X_1] - \Pr[A(x)=1|x \leftarrow X_0]\right)$$

and we get

$$\left|\Pr\left[A(x) = b \;\middle|\; \begin{array}{l} b \leftarrow \{0,1\} \\ x \leftarrow X_b \end{array}\right] - \frac{1}{2}\right| =$$

$$\frac{1}{2}\left|\Pr[A(x)=1|x \leftarrow X_1] - \Pr[A(x)=1|x \leftarrow X_0]\right| =$$

$$\frac{\Delta_A(X_0, X_1)}{2} \quad .$$

Q.E.D

Solution 2.b: we will first show

$$\frac{1}{2}\sum_{x \in S}\left|\Pr\left[X_0 = x\right] - \Pr\left[X_1 = x\right]\right| = \max_{T \subseteq S}\left|\Pr\left[X_0 \in T\right] - \Pr\left[X_1 \in T\right]\right|.$$

1: note that for $b \in \{0,1\}$:

$$\Pr\left[X_b \in T\ \right] = \Pr\left[x \in T\ \ | \ x \leftarrow X_b\right] = \sum_{x' \in T}\Pr\left[x = x' \mid x \leftarrow X_b\right] \quad .$$

2: define for $x' \in S$: $dif(x') := \Pr[x = x' \mid x \leftarrow X_0] - \Pr[x = x' \mid x \leftarrow X_1]$, and note that:

$$\sum_{x' \in T} (dif(x')) + \sum_{x' \in S \setminus T} (dif(x')) =$$

$$\sum_{x' \in S} (dif(x')) = \sum_{x' \in S} (\Pr[x = x' \mid x \leftarrow X_0] - \Pr[x = x' \mid x \leftarrow X_1]) =$$

$$\sum_{x' \in S} \Pr[x = x' \mid x \leftarrow X_0] - \sum_{x' \in S} \Pr[x = x' \mid x \leftarrow X_1] = 1 - 1 = 0 \ .$$

thus: $\sum_{x' \in T} (dif(x')) = -\sum_{x' \in S \setminus T} (dif(x')) \Rightarrow \left|\sum_{x' \in T} (dif(x'))\right| = \left|\sum_{x' \in S \setminus T} (dif(x'))\right|$. so, for any set T:

$$2|\Pr[X_0 \in T] - \Pr[X_1 \in T]| = 2\left|\sum_{x' \in T} dif(x')\right| =$$

$$\left|\sum_{x' \in T} dif(x')\right| + \left|\sum_{x' \in S \setminus T} dif(x')\right|$$

and we get the first direction of inequality:

$$\left|\sum_{x' \in T} dif(x')\right| + \left|\sum_{x' \in S \setminus T} dif(x')\right| \leq \sum_{x' \in T} |dif(x')| + \sum_{x' \in S \setminus T} |dif(x')| = \sum_{x' \in S} |dif(x')| \ .$$

for the second direction, consider T, a subset of S which maximizes $\left|\sum_{x' \in T} dif(x')\right|$, we need to show:

$$\left|\sum_{x' \in T} dif(x')\right| + \left|\sum_{x' \in S \setminus T} dif(x')\right| \geq \sum_{x' \in S} |dif(x')|$$

we claim that no two $x', x'' \in T$, suffice that $dif(x') > 0$ and $dif(x'') < 0$, and that applies not only for T, but also for $S \setminus T$. the reason for this is that if we assume there are two such samples, we can look at the sum of $dif(.)$ over samples from T, and if it is negative we take out from T the sample that was positive, and if the sum is positive we take out the sample that was negative, in both cases we increased the absolute value of the sum, in contradiction to the assumption that T maximizes the sum. as for $S \setminus T$, if it has a sample $x \in S \setminus T$ such that $dif(x)$ is of the same sign like the elements of the sum of T, than adding it to T will increase it's sum's absolute value, so it cannot have two samples of different signs, as one of them will surely add to T. if the elements of each of the two sums all have the same sign (which means all terms are (positives or zeros) OR (negatives or zeros)) we can deduce:

$$\left|\sum_{x' \in T} dif(x')\right| + \left|\sum_{x' \in S \setminus T} dif(x')\right| = \sum_{x' \in T} |dif(x')| + \sum_{x' \in S \setminus T} |dif(x')| = \sum_{x' \in S} |dif(x')| \ .$$

and that settles the first equality, we will now prove the second equality:

$$\max_A \Delta_A(X_0, X_1) = \max_{T \subseteq S} |\Pr[X_0 \in T] - \Pr[X_1 \in T]| \ .$$

let T be a subset of S, we will construct an algorithm A such that:

$$\Delta_A(X_0, X_1) \geq |\Pr[X_0 \in T] - \Pr[X_1 \in T]| \ .$$

for a sample $x \in S$, A will check if $x \in T$, if it is than A outputs 0, and if it is not it outputs 1.

note that

$$\Pr\left[A(x) = b \;\middle|\; \begin{array}{c} b \leftarrow \{0,1\} \\ x \leftarrow X_b \end{array}\right] = \frac{1}{2}\left(\Pr[A(x) = 0 \mid x \leftarrow X_0] + \Pr[A(x) = 1 \mid x \leftarrow X_1]\right) =$$

$$\frac{1}{2}\left(\Pr[x \in T \mid x \leftarrow X_0] + \Pr[x \notin T \mid x \leftarrow X_1]\right) =$$

$$\frac{1}{2} + \frac{1}{2}(\Pr[x \in T \mid x \leftarrow X_0] - \Pr[x \in T \mid x \leftarrow X_1])$$

and thus we get

$$\Delta_A(X_0, X_1) = 2\left|\Pr\left[A(x) = b \;\middle|\; \begin{array}{c} b \leftarrow \{0,1\} \\ x \leftarrow X_b \end{array}\right] - \frac{1}{2}\right| =$$

$$2\left|\frac{1}{2} + \frac{1}{2}\left(\Pr[x \in T | x \leftarrow X_0] - \Pr[x \in T | x \leftarrow X_1]\right) - \frac{1}{2}\right| =$$

$$|\Pr[X_0 \in T] - \Pr[X_1 \in T]| \ .$$

for the second direction of inequality, let A be an algorithm that gets $x \in S$ and outputs 0 or 1 (and is allowed to toss coins, which means that for an input x it might not always output the same bit), and we will show there exists a subset $T \subseteq S$ such that:

$$\Delta_A(X_0, X_1) \leq |\Pr[X_0 \in T] - \Pr[X_1 \in T]| \ .$$

recall that from the definition

$$\Delta_A(X_0, X_1) = |\Pr[A(x) = 0 \mid x \leftarrow X_0] - \Pr[A(x) = 0 \mid x \leftarrow X_1]|$$

and observe that for $b \in \{0, 1\}$:

$$\Pr[A(x) = 0 \mid x \leftarrow X_b] = \sum_{x \in S} \Pr[X_b = x] \Pr[A(x) = 0]$$

so, we get:

$$\Delta_A(X_0, X_1) = \left|\sum_{x \in S} \Pr[A(x) = 0]\left(\Pr[X_0 = x] - \Pr[X_1 = x]\right)\right| = \left|\sum_{x \in S} \Pr[A(x) = 0]\left(dif(x)\right)\right| \ .$$

now, consider the following sets: $B^+ = \{x \in S | dif(x) \geq 0\}$, and $B^- = \{x \in S | dif(x) < 0\}$ (B^- is the complement of B^+ in relation to S).

assume w.l.o.g that $\left|\sum_{x\in B^+} dif(x)\right| \geq \left|\sum_{x\in B^-} dif(x)\right|$, than it is easy to see that (because for all $x \in S$: $\Pr[A(x) = 0] \in [0, 1]$) :

$$\left|\sum_{x\in B^+} \Pr[A(x) = 0](dif(x)) + \sum_{x\in B^-} \Pr[A(x) = 0](dif(x))\right| \leq$$

$$\left|\sum_{x\in B^+} (dif(x)) + \sum_{x\in B^-} \Pr[A(x) = 0](dif(x))\right|$$

in addition, from $\left|\sum_{x\in B^+} dif(x)\right| \geq \left|\sum_{x\in B^-} dif(x)\right|$ it follows that $\left|\sum_{x\in B^+} dif(x)\right| \geq \left|\sum_{x\in B^-} \Pr[A(x) = 0] dif(x)\right|$ (again, because for all $x \in S$: $\Pr[A(x) = 0] \in [0, 1]$), and becasue the two sums are of different signs we get:

$$\left|\sum_{x\in B^+} (dif(x)) + \sum_{x\in B^-} \Pr[A(x) = 0](dif(x))\right| \leq \left|\sum_{x\in B^+} (dif(x))\right|$$

as a note, we could assume w.l.o.g that $\left|\sum_{x\in B^+} dif(x)\right| \geq \left|\sum_{x\in B^-} dif(x)\right|$ because if the opposite would have occurred, the exact same explanation would have worked, by replacing B^+ with B^-.

and we got what we wanted - a subset of S suffices the inequality:

$$\Delta_A(X_0, X_1) = \left|\sum_{x\in S} \Pr[A(x) = 0](dif(x))\right| =$$

$$\left|\sum_{x\in B^+} \Pr[A(x) = 0](dif(x)) + \sum_{x\in B^-} \Pr[A(x) = 0](dif(x))\right| \leq$$

$$\left|\sum_{x\in B^+} (dif(x))\right| = \left|\Pr\left[X_0 \in B^+\right] - \Pr\left[X_1 \in B^+\right]\right| .$$

Q.E.D

Solution 2.c: let A be a random distinguisher as described, and note that

$$\Pr\left[A(x;r) = b \;\middle|\; \begin{array}{l} b \leftarrow \{0,1\} \\ x \leftarrow X_b \\ r \leftarrow U_n \end{array}\right] = \mathbf{E}_{r\leftarrow U_n}\left[\Pr\left[A(x;r) = b \;\middle|\; \begin{array}{l} b \leftarrow \{0,1\} \\ x \leftarrow X_b \end{array}\right]\right]$$

let r_{best} be a string in U_n such that $\Pr\left[A(x;r_{best}) = b \;\middle|\; \begin{array}{l} b \leftarrow \{0,1\} \\ x \leftarrow X_b \end{array}\right]$ is the furthest from $\frac{1}{2}$, that is, r_{best} maximizes (over $r \in U_n$) the expression: $\left|\Pr\left[A(x;r) = b \;\middle|\; \begin{array}{l} b \leftarrow \{0,1\} \\ x \leftarrow X_b \end{array}\right] - \frac{1}{2}\right|$. and we conclude:

$$\left|\mathbf{E}_{r\leftarrow U_n}\left[\Pr\left[A(x;r) = b \;\middle|\; \begin{array}{l} b \leftarrow \{0,1\} \\ x \leftarrow X_b \end{array}\right]\right] - \frac{1}{2}\right| \leq \left|\Pr\left[A(x;r_{best}) = b \;\middle|\; \begin{array}{l} b \leftarrow \{0,1\} \\ x \leftarrow X_b \end{array}\right] - \frac{1}{2}\right|$$

finally, by recalling that (what we proved in question 2.a):

$$\Delta_A((X_0;U_n),(X_1;U_n)) = 2\left|\Pr\left[A(x;r)=b \;\middle|\; \begin{array}{l} b\leftarrow\{0,1\} \\ x\leftarrow X_b \\ r\leftarrow U_n \end{array}\right]-\frac{1}{2}\right|$$

$$\Delta_A((X_0;r_{best}),(X_1;r_{best})) = 2\left|\Pr\left[A(x;r_{best})=b \;\middle|\; \begin{array}{l} b\leftarrow\{0,1\} \\ x\leftarrow X_b \end{array}\right]-\frac{1}{2}\right|$$

we get the wanted inequality:

$$\Delta_A((X_0;U_n),(X_1;U_n)) = 2\left|\Pr\left[A(x;r)=b \;\middle|\; \begin{array}{l} b\leftarrow\{0,1\} \\ x\leftarrow X_b \\ r\leftarrow U_n \end{array}\right]-\frac{1}{2}\right| =$$

$$2\left|\mathbf{E}_{r\leftarrow U_n}\left[\Pr\left[A(x;r)=b \;\middle|\; \begin{array}{l} b\leftarrow\{0,1\} \\ x\leftarrow X_b \end{array}\right]\right]-\frac{1}{2}\right| \leq 2\left|\Pr\left[A(x;r_{best})=b \;\middle|\; \begin{array}{l} b\leftarrow\{0,1\} \\ x\leftarrow X_b \end{array}\right]-\frac{1}{2}\right| =$$
$$\Delta_A((X_0;r_{best}),(X_1;r_{best}))\ .$$

Q.E.D

Solution 3.a: let $S_0 = \{S_{0,n}\}_{n\in N}, S_1 = \{S_{1,n}\}_{n\in N}$ be two non-uniform PPT algorithms and $X = \{X_n\}_{n\in N}$ be a distribution ensemble such that: $(X, S_0(X)) \approx_c (X, S_1(X))$, and let $p(n)$ be a polynomial. we need to show that $Y_0 \approx_c Y_1$, where for $b\in\{0,1\}$, Y_b is the distribution ensemble $\{Y_{b,n}\}_{n\in N}$ given by:

$$\forall n\in N: Y_{b,n} = \left(X_n, \overbrace{S_{b,n}(X_n),...,S_{b,n}(X_n)}^{p(n)\ \ times}\right).$$

we will prove that if $Y_0 \not\approx_c Y_1$ than $(X, S_0(X)) \not\approx_c (X, S_1(X))$.

assume $Y_0 \not\approx_c Y_1$, this implies the existence of a non-uniform PPT adversary $A' = \{A'_n\}_{n\in N}$ and a polynomial $q'(n)$ and infinitely many $n\in N$ such that:

$$\Delta_{A'_n}(Y_{0,n}, Y_{1,n}) \geq \frac{1}{q'(n)}$$

we will construct a non-uniform PPT adversary $A = \{A_n\}_{n\in N}$ such that there exists a polynomial $q(n)$ (which will be stated later) and infinitely many $n\in N$ such that:

$$\Delta_{A_n}\left((X_n, S_{0,n}(X)), (X_n, S_{1,n}(X))\right) \geq \frac{1}{q(n)}$$

let n be a natural number such that: $\Delta_{A'_n}(Y_{0,n}, Y_{1,n}) \geq \frac{1}{q'(n)}$, and for $i\in\{0,1,...,p(n)\}$ consider the distribution:

$$Z_i = \left(X_n, \overbrace{S_{0,n}(X_n),...,S_{0,n}(X_n)}^{i\ \ times}, \overbrace{S_{1,n}(X_n),...,S_{1,n}(X_n)}^{p(n)-i\ \ times}\right).$$

and note that: $Z_0 = Y_{1,n}$, $Z_{p(n)} = Y_{0,n}$. it follows from the triangle inequality (for Δ) that:

$$\frac{1}{q'(n)} \leq \Delta_{A'_n}(Y_{0,n}, Y_{1,n}) \leq \sum_{i \in \{1,...,p(n)\}} \Delta_{A'_n}(Z_{i-1}, Z_i)$$

thus, there exists $i \in \{1, ..., p(n)\}$ such that:

$$\Delta_{A'_n}(Z_{i-1}, Z_i) \geq \frac{1}{q'(n)p(n)}$$

denote $q(n) = q'(n)p(n)$, and we now describe the adversary A that suffices:

$$\Delta_{A_n}((X_n, S_{0,n}), (X_n, S_{1,n})) \geq \frac{1}{q'(n)p(n)} = \frac{1}{q(n)}$$

for a sample (x, y), A_n will plant y in the i-ith place and compute through the $S_{b,n}$ algorithms all the rest, that is, A_n will produce the following input to A'_n, and answer exactly what A'_n outputs:

$$\left(x, \overbrace{S_{0,n}(x), ..., S_{0,n}(x)}^{i-1 \ \ times}, y, \overbrace{S_{1,n}(x), ..., S_{1,n}(x)}^{p(n)-i \ \ times} \right).$$

note that A is polynomial time, as it activates a polynomial number of times (specifically, $p(n) - 1$) algorithms that are both polynomial time. now, observe that:

- if the input to A was from $(X_n, S_{0,n})$ than the input to A' that A generated is from Z_i.
- if the input to A was from $(X_n, S_{1,n})$ than the input to A' that A generated is from Z_{i-1}.

and that finishes the proof, as it implies that:

$$\Delta_{A_n}((X_n, S_{0,n}), (X_n, S_{1,n})) = \Delta_{A'_n}(Z_{i-1}, Z_i) \geq \frac{1}{q'(n)p(n)} = \frac{1}{q(n)}$$

Q.E.D

Solution 3.b (bonus question): we will show that if there exists computationally-secure (secret-key) encryption for keys of length n and messages of length $l(n) > n$, than there exists a distribution ensemble $X = \{X_n\}_{n \in N}$ and an inefficient algorithm S_0 and an efficient randomized algorithm S_1 (we are allowed for both to be inefficient, but we will need only S_0 to be) such that: $(X, S_0(X)) \approx_c (X, S_1(X))$ but $Y_0 \not\approx_c Y_1$, where the Y_b distributions are defined as before (later in the proof we will state what exactly is the polynom $p(n)$ that denotes the number of S_b outputs in Y_b).

consider a computationally-secure encryption scheme (E, D) as described, with messages that are at least 7 bits longer than keys (we can assume that, because if the encryption does not suffice the assumption, we can, as we seen in lecture 2, create a new computationally-secure scheme with long enough messages for our current need of at least 7 extra bits).

this, as we proved in question 1 (in this current problem set) implies the existence of 2 messages $m_0, m_1 \in \{0,1\}^{l(n)}$ and an inefficient attacker A such that:

$$\Pr\left[A(ct) = m_b \;\middle|\; \begin{array}{l} sk \leftarrow \{0,1\}^n \\ b \leftarrow \{0,1\} \\ r \leftarrow \{0,1\}^n \\ ct = E_{sk}(m_b; r) \end{array} \right] > 0.99 \ .$$

for every $n \in N$ there are such messages and attacker for (E, D), and we will take X to be the distribution ensemble of these two messages:

$$\forall n \in N : X_n = \{ct | b \leftarrow \{0,1\}, sk \leftarrow \{0,1\}^n, ct \leftarrow E_{sk}(m_b)\}$$

we will take $S_0 = \{S_{0,n}\}_{n \in N}$ to be the ensemble of attackers: for every n, $S_{0,n}$ will be the attacker that distinguishes well between the two messages sampled and encrypted from X_n. as for S_1, for an input ct, S_1 will choose one bit at random (and uniformly) and output it - which means S_1 will just guess where the ciphertext came from.

it is hard for a PPT to tell between $(X, S_0(X))$ and $(X, S_1(X))$, because it is hard for a PPT to tell between ciphertexts of m_0 and ciphertexts of m_1 (comes from the assumption the (E, D) is computationally-secure): assume there is an adversary A' and a polynomial $q(n)$ such that for infinitely many $n \in N$:

$$\Delta_{A'_n}\left((X_n, S_{0,n}(X)), (X_n, S_{1,n}(X)) \right) \geq \frac{1}{q(n)} \ .$$

than for those values of n we can break (E, D) (for the two messages we know $S_{0,n}$ can distinguish): consider an adversary A that given a ciphertext ct, runs $A'(ct, 0)$ and outputs it's result. the intuition is that we know S_0 knows to distinguish ciphertexts of m_0, m_1 with good probability, and also that A' knows to tell when the last bit came from S_0 or from S_1 with good enough (polynomial) probability, so, if A' output 0 it means we have a good chance of being correct (and we gave zero as input, so zero might be correct) and if we got 1, that means A' thinks that bit was from S_1 which have a great chance of being wrong, so if we gave zero and it was not correct (with good enough probability) than 1 have a good chance of being correct.

it is left to show that when repeated answers come from S_b, it is easy for a PPT to guess who's who. indeed, it will be enough to just get one more answer from S_b (which means $p(n) = 2$ will do) to distinguish, formally, we will show:

$$(X, S_0(X), S_0(X)) \not\approx_c (X, S_1(X), S_1(X))$$

consider an adversary A such that for an input (ct, b_1, b_2) outputs 0 iff $b_1 = b_2$. recall that S_0 is deterministic, thus it's bits will always be the same, and it

follows that:

$$\Pr\left[A(ct,b_1,b_2)=0 \;\middle|\; \begin{array}{l} ct \leftarrow X \\ b_1 \leftarrow S_0(ct) \\ b_2 \leftarrow S_0(ct) \end{array}\right] = 1$$

$$\Pr\left[A(ct,b_1,b_2)=1 \;\middle|\; \begin{array}{l} ct \leftarrow X \\ b_1 \leftarrow S_1(ct) \\ b_2 \leftarrow S_1(ct) \end{array}\right] = \frac{1}{2}$$

thus we get:

$$\Delta_A\left((X,S_0(X),S_0(X)),(X,S_1(X),S_1(X))\right) =$$

$$2\left|\Pr\left[A(ct,b_1,b_2)=b \;\middle|\; \begin{array}{l} b \leftarrow \{0,1\} \\ ct \leftarrow X \\ b_1 \leftarrow S_b(ct) \\ b_2 \leftarrow S_b(ct) \end{array}\right] - \frac{1}{2}\right| =$$

$$2\left|\frac{1}{2}\left(1+\frac{1}{2}\right)-\frac{1}{2}\right| = \frac{1}{2} \ .$$

Q.E.D