Anish Banerjee, Shankh Gupta
COL759: Cryptography

Problem Set - 1
August 2023

Problem 1: Perfect 2 time security

Solution:

(a) Consider an Adversary $\mathcal{A}$ that sends challenger the following pairs of messages : (m_{00}, m_{01}) and (m_1, m_1). If the Encryption Scheme $\mathcal{E}$ were to be deterministic, the challenger would send either a pair of two distinct ciphertexts (in case it encrypts the first pair, i.e. b = 0) or a pair of two same ciphtertexts (in case it encrypted the second pair, i.e. b = 1). In this scenario, $\mathcal{A}$ would break the Encryption Scheme $\mathcal{E}$ with surety.

But we can have randomized encryption schemes with randomness drawn from $\{0,1\}^{l_3}$. Still there is a non-zero probability of $\frac{1}{2^{l_3}}$ of the two random strings selected to be same and thus the cipher texts will turn out to be same for two same messages.

(b) The encryption scheme $\mathcal{E} = (\mathsf{Enc}, \mathsf{Dec})$ which satisfies $O(2^{-l})$-perfect-two-time Security can be constructed as follows:

$$\mathsf{Enc}(k, m, r) = (r, m \oplus h_k(r))$$

and

$$\mathsf{Dec}(k, (ct_0, ct_1)) = ct_1 \oplus h_k(ct_0)$$

where r is a random string sampled from the Random space. To prove its security first consider the given PIF security game between a challenger $\mathcal{C}$ and adversary $\mathcal{A}$ (adversary can be inefficient):

PIF Security Game

- The challenger samples uniformly a bit $b \leftarrow \{0,1\}$ and a key $k \leftarrow \mathcal{K}$.
- The adversary then sends two queries x_0 and x_1 to the challenger. If $b = 0$ then the challenger computes $h_k(x_0)$ and $h_k(x_1)$ and sends them to $\mathcal{A}$. Else it samples uniformly a random function f with input and output space as $\{0,1\}^l \to \{0,1\}^l$ and outputs $f(x_0)$ and $f(x_1)$.
- The adversary sends a bit b' to the challenger.

Claim: If h_k is a Pairwise Independent Hash Function(PIF) then winning probability of $\mathcal{A}$ in the above security game will be $\frac{1}{2}$.

Proof : We will prove the contrapositive. Suppose an adversary $\mathcal{A}$ has non-zero advantage in the above PIF Game. $\mathcal{A}$ performs the following :

- For every $k \in \mathcal{K}$, it checks whether $h_k(x_0) = y_0$ & $h_k(x_1) = y_1$ where y_0 and y_1 are the outputs sent by the challenger.
- If it finds such k, then $\mathcal{A}$ sends 0 to the challenger else it sends 1.

Now

$$\mathsf{PIFAdv}[\mathcal{A}] = \Big|\Pr[b' = 0 | b = 0] - \Pr[b' = 0 | b = 1]\Big|$$

which is non-zero. Also,

$$\Pr[b' = 0 | b = 0] = \Pr[h_k(x_0) = y_0 \wedge h_k(x_1) = y_1]$$

and $\Pr[b' = 0 | b = 1]$ will be the same as getting two random strings of length l same as $h_k(x_0)$ and $h_k(x_1)$ which will be $\frac{1}{2^{2l}}$. So, this implies that $\Pr[h_k(x_0) = y_0 \wedge h_k(x_1) = y_1] \neq \frac{1}{2^{2l}}$ for some distinct x_0 and x_1 and y_0 and y_1. So, h_k is not a PIF.

Now we need to prove that our construction which uses $\mathcal{H}$ function family is almost perfect two time secure. We will prove the contra-positive that if there exists a ppt. adversary $\mathcal{A}$ that breaks the security of our encryption scheme $\mathcal{E}$ then the function family $\mathcal{H}$ is not PIF, i.e., there exists a reduction algorithm $\mathcal{B}$ that breaks the PIF security game. To prove this consider the following worlds:

- **World-b** : Challenger encrypts message $m_b = (m_{b0}, m_{b1})$ and sends $ct = (ct_0, ct_1)$ to the adversary where $ct_0 = (r_0, Enc(k, m_{b0}, r_0))$ and $ct_1 = (r_1, Enc(k, m_{b1}, r_1))$. Let p_b be the probability that $\mathcal{A}$ outputs 0 in World-b.
- **Hybrid-World-b** : Challenger encrypts message $m_b = (m_{b0}, m_{b1})$ by sampling a random function f from the given input-output space and sends $ct = (ct_0, ct_1)$ to the adversary where $ct_0 = (r_0, m_{b0} \oplus f(r_0))$ and $ct_1 = (r_1, m_{b1} \oplus f(r_1))$. Let $p_{hyb,b}$ be the probability that $\mathcal{A}$ outputs 0 in Hybrid-World-b.

Now given that $|p_0 - p_1|$ is non negligible, we can say that either $|p_0 - p_{hyb,0}|$ or $|p_1 - p_{hyb,1}|$ is non-negligible. Note that $p_{hyb,0}$ and $p_{hyb,1}$ are negligibly far apart (because of part(a)).

Observation : There exists a reduction algorithm $\mathcal{B}_i$ such that PIFAdv$[\mathcal{B}] = |p_i - p_{hyb,i}|$. (where i = 0 or 1)
Proof : The algorithm $\mathcal{B}_i$ samples two random strings r_0 and r_1 and sends them to the PIF challenger. It receives m_0 and m_1 from A and $f(r_0)$ and $f(r_1)$ from challenger. Now it sends $m_{i0} \oplus f(r_0)$ and $m_{i1} \oplus f(r_1)$. If f is a PIF function, then that corresponds to World-i for $\mathcal{A}$ else it corresponds to Hybrid-World-i. So we can thereby say that the advantage of $\mathcal{B} = |p_i - p_{hyb,i}|$.

Problem 2 : Secure/Insecure PRGs and PRFs

Solution:

(a) PRGs

i. $\mathcal{G}' = \left\{G'_n : \{0,1\}^{2n} \to \{0,1\}^{3n}\right\}_{n\in\mathbb{N}}$, where

$$G'_n(s_1 \mid\mid s_2) = G_n(s_1) \wedge G_n(s_2).$$

The given PRG is **insecure**. Consider the PRG game between $\mathcal{A}$ and G' challenger where on input y, $\mathcal{A}$ outputs the last bit of y. Let $L(x)$ denote the last bit of x. Note that if G is secure, then $\Pr[L(G(s)) = 0]$ will be close to $1/2$. Otherwise, if it is $1/2 + \epsilon$, we can create an adversary breaking G with non-negligible advantage ϵ ($\mathcal{A}$ always outputs 0). So, if we take $\Pr[L(G(s)) = 0] = 1/2 + \mathsf{negl}(\lambda)$

$$\Pr[b' = 0|b = 0] = \Pr[L(G(s_1) \wedge G(s_2)) = 0]$$

$$\leq \Pr[L(G(s_1)) = 0 \wedge L(G(s_2)) = 0] + \Pr[L(G(s_1)) = 1 \wedge L(G(s_2)) = 0] + \Pr[L(G(s_1)) = 0 \wedge L(G(s_2)) = 1]$$

$$\approx 3/4 + \mathsf{negl}(\lambda)$$

and

$$\Pr[b' = 0|b = 1] = 1/2$$

Thus the $\mathsf{PRGAdv}[\mathcal{A}, \mathcal{G}] \approx 1/4$ which is non-negligible.

ii. $\mathcal{G}' = \left\{G'_n : \{0,1\}^{2n} \to \{0,1\}^{3n}\right\}_{n\in\mathbb{N}}$, where

$$G'_n(s_1 \mid\mid s_2) = G_n(s_1) \oplus G_n(s_2)$$

This is a **secure** PRG. We prove the security by a hybrid argument.

- **World0** The challenger sends $G_n(s_1) \oplus G_n(s_2)$ to the attacker
- **HybridWorld** The challenger sends $G_n(s_1) \oplus \mathsf{random}_1$ to the attacker
- **World1** The challenger sends $\mathsf{random}_1 \oplus \mathsf{random}_2$ to the attacker

Claim: If any adversary $\mathcal{A}$ can distinguish between World0 and HybridWorld then we can construct $\mathcal{B}$ which breaks the PRG security of G.

The reduction $\mathcal{B}$ receives y from the PRG challenger. It samples $s \leftarrow \{0,1\}^n$ and sends $G(s) \oplus y$ to $\mathcal{A}$. The advantage of $\mathcal{A}$ in distinguishing between World0 and HybridWorld will be equal to the advantage of $\mathcal{B}$ in breaking PRG security of G.

Similarly we can also claim that:

Claim: If any adversary $\mathcal{A}$ can distinguish between HybridWorld and World1 then we can construct $\mathcal{B}$ which breaks the PRG security of G.

The reduction $\mathcal{B}$ receives y from the PRG challenger. It samples $r \leftarrow \{0,1\}^n$ and sends $r \oplus y$ to $\mathcal{A}$. The advantage of $\mathcal{A}$ in distinguishing between HybridWorld and World1 will be equal to the advantage of $\mathcal{B}$ in breaking PRG security of G.

Now we can choose any reduction randomly to break the PRG security of G. Also note that we cannot use a similar argument in part i. because $\mathsf{random}_1 \wedge \mathsf{random}_2$ is not uniformly random.

(b) PRFs

i. $\mathcal{F}' = \left\{F'_n : \{0,1\}^n \times \{0,1\}^{2n} \to \{0,1\}^n\right\}_{n\in\mathbb{N}}$ where

$$F'_n(k,(x_1,x_2)) = F_n(k,x_1) \oplus F_n(k,x_2).$$

The given family $\mathcal{F}'$ is **insecure**. Consider a PPT attacker $\mathcal{A}$ who sends $\mathsf{poly}(\lambda)$ distinct (x_i, x_i) queries to the challenger. If the challenger chooses $b = 0$ then it will end up sending

$$F_n(k,x_i) \oplus F_n(k,x_i) = 0^n$$

for each of the queries. The attacker outputs 0 if all the responses are 0 and 1 otherwise. Advantage of the attacker is close to 1, precisely $1 - 2^{-n\mathsf{poly}(\lambda)}$.

ii. $\mathcal{F}' = \{F'_n : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n\}_{n\in\mathbb{N}}$ where

$$F'_n(k,x) = F_n(k,x) \oplus x.$$

The given family is **secure**. Given an adversary $\mathcal{A}$ which breaks PRF security of $\mathcal{F}'$, we can construct an adversary $\mathcal{B}$ which breaks the security of $\mathcal{F}$ (Fig. 1)

Problem 2(b)(ii)

- Challenger picks a uniformly random bit $b \leftarrow \{0,1\}$ and a seed $s \leftarrow \{0,1\}^n$.
- The adversary $\mathcal{A}$ makes polynomially many queries to $\mathcal{B}$, who passes them to the challenger. Challenger replies as in the PRF Game.
- Upon receiving the response y_i of each query, $\mathcal{B}$ sends $y_i \oplus x_i$ to $\mathcal{A}$
- After polynomially many queries, $\mathcal{B}$ forwards the response send by $\mathcal{A}$ (b') and wins if $b = b'$.

Figure 1: Reduction for Problem 2(b)(ii)

It is clear from the reduction that the PRF advantage (F) of $\mathcal{B}$ in the above game is same as the PRF advantage (F') of $\mathcal{A}$

Problem 3 : PRG Security does not imply Related-Key-PRG Security

Solution:

(a) The construction of $\mathcal{G}'$ from $\mathcal{G}$ is as follows :
$\mathcal{G}' = \left\{ G'_n : \{0,1\}^{2n} \leftarrow \{0,1\}^{6n} \right\}_{n \in \mathbb{N}}$ where $G'_n(s) = G(s_0) \,||\, G(s_1)$ where s_0 and s_1 are the first n bits and the last n bits of s respectively and $||$ denotes join(append) operation.

(b) The following related key attack can be launched against our length-expanding function $\mathcal{G}'$:

- In our Related-key attack game, the challenger picks uniformly a bit $b \leftarrow \{0,1\}$ and a seed $s \leftarrow \{0,1\}^{2n}$.
- The adversary $\mathcal{A}$ makes two queries to the challenger. If $b = 1$, it picks a uniformly random string $y \leftarrow \{0,1\}^{6n}$ in each query and sends it to $\mathcal{A}$. Else it computes $G'(s)$ and $G'(s+1)$ and sends it to $\mathcal{A}$. Let the two strings sent by the challenger to the adversary be y_1 and y_2.
- The adversary checks whether the first $3n$ bits of y_1 are equal to the first $3n$ bits of y_2 or not. If that's the case, then $\mathcal{A}$ sends 0 to the challenger else it sends 1 to the challenger.

We can show that this adversary $\mathcal{A}$ wins the related-key security game with a non-negligible advantage. This is because if $s = s_0||s_1$ then $s+1 = s_0||(s_1+1)$ unless $s_1 = 1^n$. Meaning the first n bits remain same in most of the cases after adding 1 to our 2n bit string s. Since $G'_n(s) = G(s_0) \,||\, G(s_1)$ and $G'_n(s+1) = G(s_0) \,||\, G(s_1+1)$, the first 3n bits will be the same in both the strings and hence the adversary can guess whether the strings sent by the challenger are psuedorandom or completely random.

This will be the case for all but 2^n seeds s out of total 2^{2n} possible seeds s (where the last n bits of s are 1^n). So for all these $2^{2n} - 2^n$ seeds, the adversary $\mathcal{A}$ can guess b' with surety. $\mathcal{A}$ will incorrectly send 0 only on the 2^n seeds with last n bits 1^n. Also it may happen that when challenger chooses $b = 1$ and samples two uniformly random $6n$ bits, they may have first $3n$ bits same. So the advantage of $\mathcal{A}$ in the related-key security game is :

$$RKAdv[\mathcal{A}, \mathcal{G}'] = Pr[b' = 0 | b = 0] - Pr[b' = 0 | b = 1]$$

where $Pr[b' = 0|b = 0]$ is $\frac{2^{2n} - 2^n}{2^{2n}}$ which is same as $\frac{1}{2^n}$. and $Pr[b' = 0|b = 1]$ is $\frac{1}{2^{3n}}$ which is much lesser.
So winning probability of $\mathcal{A}$ is approximately $1 + \frac{1}{2^n}$

(c) Consider the following worlds, World-0, World-1 and Hybrid-World, in the PRG security game *wrt* G' and a ppt. adversary $\mathcal{A}$:

- **World-0** : Challenger sends $G'(s)$ to $\mathcal{A}$, where s is sampled uniformly from $\{0,1\}^{2n}$. Let $Pr[b' = 0] = p_0$ in World-0.
- **World-1** : Challenger sends a random $y \leftarrow \{0,1\}^{6n}$ to $\mathcal{A}$. Let $Pr[b' = 0] = p_1$ in World-1.
- **Hybrid-World** : Challenger samples a seed $s \leftarrow \{0,1\}^n$ and a random string $y' \leftarrow \{0,1\}^{3n}$ and sends $G(s)||y'$ to $\mathcal{A}$. Let $Pr[b' = 0] = p_{hyb}$ in Hybrid-World.

We will show that if A breaks the PRG security of $\mathcal{G}'$ then there exists a ppt. reduction algorithm $\mathcal{B}$ such that $\mathcal{B}$ breaks the PRG security of $\mathcal{G}$ using $\mathcal{A}$. This means we that $|p_0 - p_1| = \epsilon$, where ϵ is non-negligible. So this follows that either $|p_0 - p_{hyb}| >= \epsilon/2$ or $|p_1 - p_{hyb}| >= \epsilon/2$, in either case we can find a reduction algorithm $\mathcal{B}$ such that B breaks PRG security of $\mathcal{G}$.

Observation 1 : There exists a reduction algorithm B_0 such that

$$\mathsf{PRGAdv}[B_0, G] = |p_0 - p_{hyb}|$$

Proof : The reduction algorithm B_0 receives a 3n bit string y from the PRG challenger and samples a uniformly random seed $s \leftarrow \{0,1\}^n$ and computes $G(s)$. It sends $G(s)||y$ to adversary $\mathcal{A}$. Now if y is pseudorandom then that corresponds to World-0 for $\mathcal{A}$ whereas if y is completely random, then that corresponds to Hybrid-World for $\mathcal{A}$. So if $b' = 0$ then B_0 sends 0 to the challenger indicating pseudorandom else it sends 1.
$\Pr[B_0 \text{ sends } 0 \mid \text{Challenger chooses } 0] = \Pr[\mathcal{A} \text{ sends } 0 \text{ in World-0}] = p_0$
$\Pr[B_0 \text{ sends } 0 \mid \text{Challenger chooses } 1] = \Pr[\mathcal{A} \text{ sends } 0 \text{ in Hybrid-World}] = p_{hyb}$

Observation 2 : There exists a reduction algorithm B_1 such that $PRGAdv[B_1, G] = |p_1 - p_{hyb}|$.
Proof : The reduction algorithm B_1 receives a 3n bit string y from the PRG challenger and samples a uniformly random sring $y' \leftarrow \{0,1\}^{3n}$. It sends $y||y'$ to adversary $\mathcal{A}$. Now if y is pseudorandom then that corresponds to Hybrid-World for $\mathcal{A}$ whereas if y is completely random, then that corresponds to World-1 for $\mathcal{A}$. So if $b' = 0$ then B_1 sends 0 to the challenger indicating pseudorandom else it sends 1.
$\Pr[B_1 \text{ sends } 0 \mid \text{Challenger chooses } 0] = \Pr[\mathcal{A} \text{ sends } 0 \text{ in World-1}] = p_1$
$\Pr[B_0 \text{ sends } 0 \mid \text{Challenger chooses } 1] = \Pr[\mathcal{A} \text{ sends } 0 \text{ in Hybrid-World}] = p_{hyb}$

Problem 4 : Constructing PRFs from PRGs

Solution: We will use a tree construction similar to the one given in the book (Fig. 2)

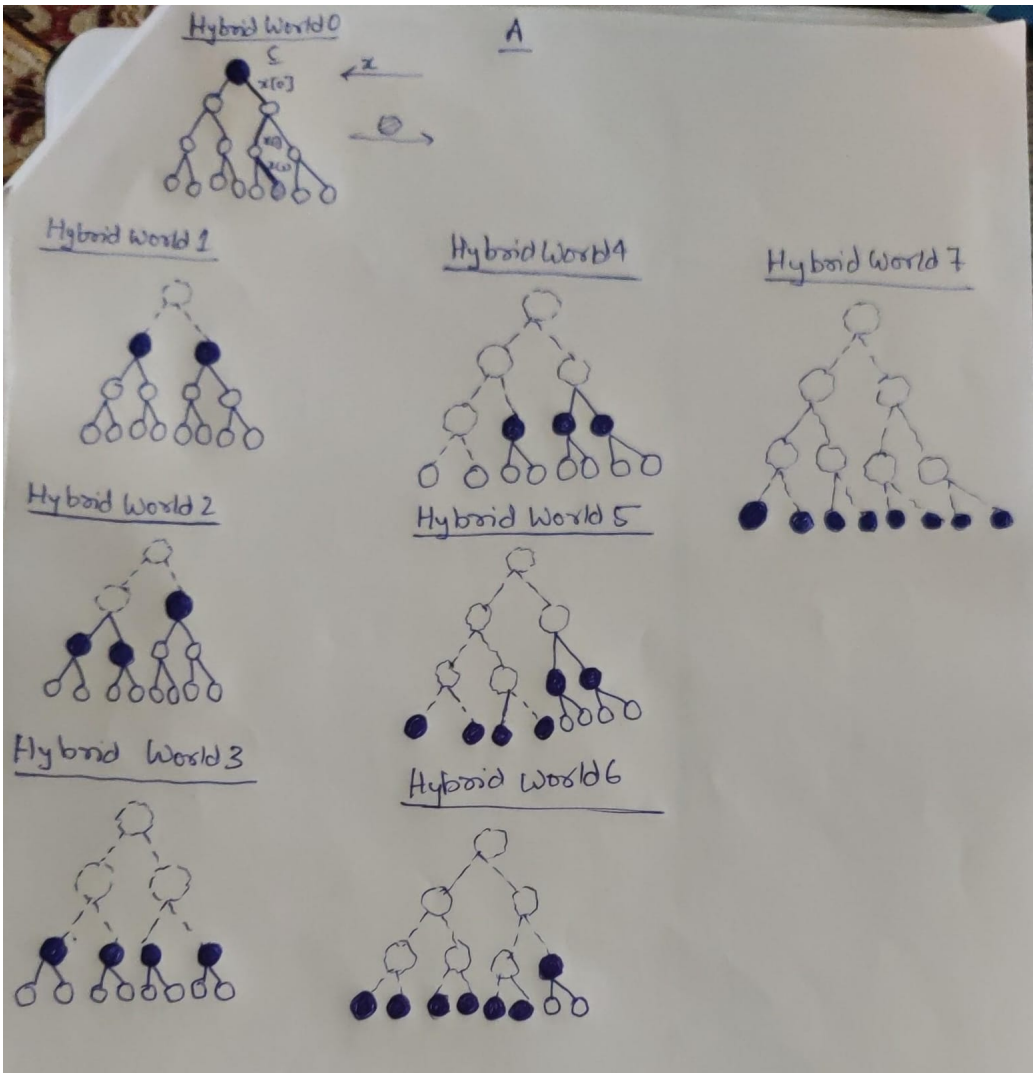

Figure 2: Construction of Hybrid worlds for the case $\log n = 3$. The randomly generated nodes are shadedand the nodes which can be ignored are dotted

(a) Construct n hybrid worlds in the following way: In Hybrid j, the challenger builds an evaluation tree whose nodes are labeled as follows:

- The first j nodes (as appearing in the level order traversal of the tree) can be ignored.
- The next $j+1$ nodes are labelled with random values.
- Remaining nodes are derived from their parents.

In response to a query $x \in \{0,1\}^{\log n}$ in Hybrid j, the challenger sends to the adversary the label of the leaf addressed by x.

Observe that Hybrid 0 corresponds to the case $b = 0$ in the PRF game when the challenger sends $F_k(x)$ and Hybrid l corresponds to $b = 1$ when the challenger uses a truly random function.

Claim: If there exists an adversary $\mathcal{A}$ which can distinguish between Hybrid i and Hybrid $i+1$ then we can construct an adversary $\mathcal{B}$ which breaks the PRG security of G.

Problem 4(a)

- Challenger picks a uniformly random bit $b \leftarrow \{0,1\}$ and a seed $s \leftarrow \{0,1\}^n$. If $b = 0$, he sends $y = G(s)$ to $\mathcal{B}$ otherwise he sends a $y = r \leftarrow \{0,1\}^{2n}$.
- $\mathcal{B}$ constructs the evaluation tree.
  - The first $i+1$ nodes can be ignored
  - The next i nodes are randomly generated
  - The next two nodes are made by splitting y sent by the challenger into two halves
  - Remaining nodes are generated using the algorithm from their parents.
- The adversary $\mathcal{A}$ makes polynomially many queries to $\mathcal{B}$. In response to a query $x \in \{0,1\}^{\log n}$, $\mathcal{B}$ sends the label of the leaf addressed by x (This traveral is shown in the HybridWorld0 of Fig. 3)
- After polynomially many queries, $\mathcal{B}$ forwards the response send by $\mathcal{A}$ (b') and wins if $b = b'$.

Figure 3: Reduction for Problem 4(a)

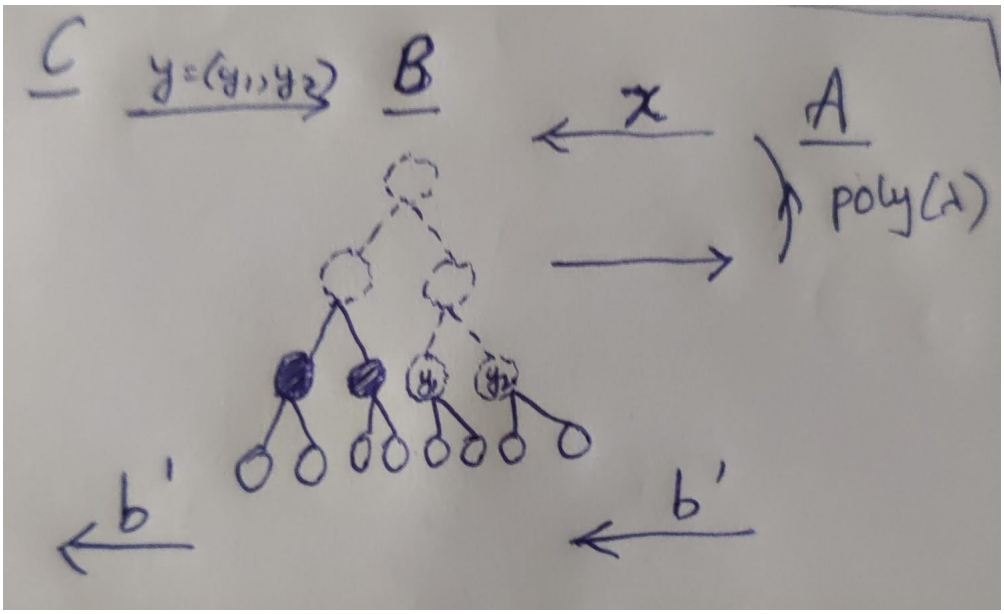

Figure 4: Reduction for distinguishing between HybridWorld2 and HybridWorld3

Proof: Consider the reduction Fig. 3

From the construction, it can be checked that:

$$\Pr[b' = 0 | b = 0] = \Pr[\mathcal{A} \text{ outputs 0 in HybridWorld } i] = p_i$$

$$\Pr[b' = 0 | b = 0] = \Pr[\mathcal{A} \text{ outputs 0 in HybridWorld } i+1] = p_{i+1}$$

Thus,

$$\mathsf{PRGAdv}[\mathcal{B}, \mathcal{G}] = |p_i - p_{i+1}|$$

Finally for the combined reduction, we choose any one of the reductions at random.

$$\mathsf{PRGAdv}[\mathcal{B}^c, \mathcal{G}^c] = \frac{|p_l - p_0|}{\log n}$$

(b) In the above construction, $\mathcal{B}$ will need to sample $O(2^d)$ random bitstrings in some hybrids, where d is the depth of the tree. If $d = \log n$ then $\mathcal{B}$ samples $O(n)$ bitstrings. However, if $d = n$ then he has to sample exponentially many strings, making him inefficient. So, we cannot use the same reduction when $x \in \{0,1\}^n$.

(c) The given construction is insecure. Consider an adversary $\mathcal{A}$ which plays the following game $\mathcal{G}$:

- $\mathcal{A}$ sends x to the Challenger and receives y_1
- $\mathcal{A}$ sends $x||1$ to the Challenger and receives y_2
- $\mathcal{A}$ finds $G(y_1) = (s_0, s_1)$ and checks if $s_1 = y_1$. If so it returns $b' = 0$ else $b' = 1$

Now,

$$\mathsf{PRFAdv}[\mathcal{A}, \mathcal{G}] = |\Pr[b' = 0|b = 0] - \Pr[b' = 0|b = 1]|$$

From construction we have,

$$\Pr[b' = 0|b = 0] = 1$$

and

$$\Pr[b' = 0|b = 1] = \Pr[G(y_1) = (s_0, s_1) \wedge s_1 = y_1] = 2^{-n}$$

Thus the $\mathsf{PRFAdv}[\mathcal{A}, \mathcal{G}] = 1 - 2^{-n}$

(d) The main problem in the above scheme occurs when one of the queries by the adversary is a proper prefix of the other. To avoid this, we modify the scheme by first applying a PRG on the query x. Let $\mathcal{G}' = \{G'_n : \{0,1\}^n \to \{0,1\}^{l(n)}\}_{n\in\mathbb{N}}$ be a secure PRG family where $l(n)$ is poly-bounded. In the scheme, the challenger first applies G'_n on the input x sent by the adversary and then uses the tree construction replacing $x \mapsto G'_n(x)$.

The main intuition behind this construction is that for an *efficient* adversary, it should be difficult to find $x_1, x_2, |x_1| = n_1 < n_2 = |x_2|$ such that $G'_{n_1}(x_1)$ is a proper prefix of $G'_{n_2}(x_2)$. Then we can use Theorem 4.11 of the book which states that *"If G is a secure PRG, then the variable length tree construction derived from G is a prefix-free secure PRF"*

Problem Part B : Coding Problems

1. **CRIME Attack:** The key observation for this attack is :

   " If cmsg contains a substring of the cookie, then the compressed string is shorter"

   In our code, we attempt a brute-force attack on the encryption scheme by iteratively guessing and refining the secret cookie character by character, exploring alternative paths when multiple characters result in the same encrypted text length. The basic idea is to iteratively build the cookie by finding the character that, when appended to the current partial key, produces the shortest encrypted text. If multiple such characters exist, then they are all stored in a list and their alternative paths are explored too.

2. **Attack on 2DES encryption:** Here, we implement a Meet-in-the-middle Attack on 2DES. Namely, we encrypt the messages using all possible keys and store them in a dictionary. We also decrypt the ciphertext using all possible keys and store them in a list. Finally, we search for the intermediary ciphertext by finding a common element in the dictionary and the list. The keys corresponding to this intermediary ciphertext are the keys we were searching for. Time complexity for the attack is $O(|\mathcal{K}|)$