Homework 3

David Miller & Hannah McLaughlin
CIS 5371: Cryptography

October 22, 2018

Problem 1. *In class weve learned the following security notions for encryption:*

- *Left-or-right (LR) security: Here the adversary has an oracle Enc(,) and is dropped into either a left world, or a right world. For each query Enc(m_0, m_1) with $|m0| = |m1|$, in the left world, the oracle encrypts m_0, and in the right world, it encrypts m1. The adversary has to tell which world (left or right) it is in.*
- *Real-or-random (RR) security: Here the adversary has an oracle Enc() and is dropped into either a real world, or a random world. For each query Enc(m), in the real world, the oracle encrypts m, and in the random world, it encrypts a random message of the same length as m. The adversary has to tell which world (real or random) it is in.*

Note that in both notions above, an adversary may make multiple queries. Prove that RR security implies LR security.

We want to prove that Real-or-Random security implies Left-or-Right security, therefore we take the contrapositive and prove that Left-or-Right insecurity implies Real-or-Random insecurity. That is, given an efficient adversary A that breaks LR security, we can construct an efficient adversary B that breaks RR security. Essentially we want to show that if $\mathbf{Adv}^{LR}(A) = \epsilon$, then we can construct an adversary B such that $\mathbf{Adv}^{RR}(B) = \alpha\epsilon$, where $\alpha \in \mathbb{R}_{>0}$ so that we keep a "high" advantage. Let $A : x \Rightarrow B : y$ mean A submits x to B which produces output y (omission of y means no output at that stage), then

Algorithm 1 Construction of B

1: $b' \xleftarrow{\$} \{0,1\}$ ▷ Randomly choose a bit
2: **for** $i = 1$ to q **do** ▷ Iterate over q queries
3: $\quad A : (m_0, m_1) \Rightarrow B : c_i \leftarrow Enc(m_{b'})$ ▷ B encrypts $m_{b'}$ via oracle
4: $\quad B : c_i \Rightarrow A$ ▷ B tells A resulting ciphertext
5: **end for**
6: $A : b^* \leftarrow \{0,1\} \Rightarrow B$ ▷ A guesses b' with value b^*
7: **if** $b' = b^\star$ **then** ▷ A guessed cprrectly
8: $\quad$ **return** 0
9: **else** ▷ A guessed incorrectly
10: $\quad$ **return** 1

Now we compute the advantage of B:

$$\begin{aligned}\mathbf{Adv}^{RR}(B) = \Pr[B \text{ guessed } 0 \mid b = 0]\Pr[b = 0] + \Pr[B \text{ guessed } 1 \mid b = 1]\Pr[b = 1] \\ - \Pr[B \text{ guessed } 0 \mid b = 1]\Pr[b = 1] - \Pr[B \text{ guessed } 1 \mid b = 0]\Pr[b = 0]\end{aligned} \tag{1}$$

From algorithm 1 we have that

$$B \text{ guessed } b = \begin{cases} b^\star = b' & \text{if } B \text{ thinks real world} \\ b^\star \neq b' & \text{if } B \text{ thinks random world} \end{cases}$$

which allows us to rewrite equation (1) as

$$\begin{aligned} \mathbf{Adv}^{RR}(B) = \Pr[b^\star = b' \,|\, b = 0]\Pr[b = 0] + \Pr[b^\star = b' \,|\, b = 1]\Pr[b = 1] \\ - \Pr[b^\star \neq b' \,|\, b = 1]\Pr[b = 1] - \Pr[b^\star \neq b' \,|\, b = 0]\Pr[b = 0] \end{aligned} \tag{2}$$

Since b is a uniformly random distributed variable over $\{0, 1\}$ we have that $\Pr[b = 0] = \Pr[b = 1] = \frac{1}{2}$. We also have that $\Pr[b^* = b' \,|\, b = 1] = \Pr[b^* \neq b' \,|\, b = 1] = \frac{1}{2}$. This is because the random oracle is instantiated ($b = 1$) and therefore A only sees random strings in response to its queries independent of what value B has for b'. This implies the advantage of A is bounded as $\frac{1}{2} \leq \mathbf{Adv}(A) \leq \frac{1}{2}$. From this we have

$$\begin{aligned} \mathbf{Adv}^{RR}(B) &= \frac{1}{2}\Bigg(\Pr[b^\star = b' \,|\, b = 0] + \overbrace{\Pr[b^\star = b' \,|\, b = 1] - \Pr[b^\star \neq b' \,|\, b = 1]}^{=0} - \Pr[b^\star \neq b' \,|\, b = 0]\Bigg) \\ &= \frac{1}{2}\Bigg(\Pr[b^\star = b' \,|\, b = 0] - \Pr[b^\star \neq b' \,|\, b = 0]\Bigg) \end{aligned} \tag{3}$$

When $b = 0$ then the game is essentially just a LR game. Then equation (3) can just be recast as

$$\begin{aligned} \mathbf{Adv}^{RR}(B) &= \frac{1}{2}\Bigg(\quad \Pr[b^\star = b' \,|\, b = 0] \quad - \quad \Pr[b^\star \neq b' \,|\, b = 0] \quad \Bigg) \\ &\qquad\qquad \Big\downarrow \qquad\qquad\qquad \Big\downarrow \\ &= \frac{1}{2}\Bigg(\Pr[A \text{ guesses correctly}] - \Pr[A \text{ guesses incorrectly}]\Bigg) \end{aligned} \tag{4}$$

where equation (4) is just $\frac{1}{2}\mathbf{Adv}^{LR}(A)$. It follows directly from (4) that $\mathbf{Adv}^{RR}(B) = \frac{1}{2}\epsilon$ which proves that RR security implies LR security.

Note that the efficiency of B is just $\mathcal{O}(A) + \mathcal{O}(1)$ which makes B an efficient adversary.

Problem 2. *Let $E : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ be a good blockcipher (meaning that you can model it as a PRF). Alice wants to build a PRF $F : \{0,1\}^k \times \{0,1\}^{n-1} \to \{0,1\}^{2n}$ on top of E. Bob suggests the following constructions:*

- $G_K(x) = E_K(1\|x)\|E_K(0\|x)$, *or*
- $H_K(x) = E_K(1\|x)\|E_K(x\|0)$.

Note that among the two constructions above, one is actually secure, but the other is trivially broken. Inform Alice which one to choose. If you think a construction is secure, prove it; otherwise, show an attack.

Attack of $\mathbf{H_k(x)}$:

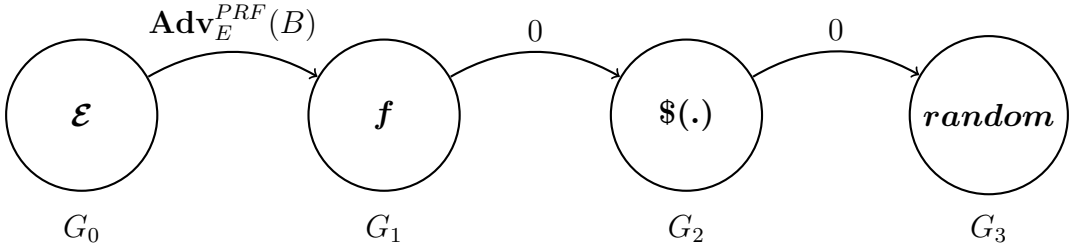
Let $x_1 = 1^{n-2}0$ and $x_2 = 1^{n-1}$ then we have that

$$H_K(x_1) = E_K(1\|x_1)E_K(x_1\|0) = E_K(1^{n-1}0)E_K(1^{n-2}0^2) = c_1^1 c_2^1 \quad (1)$$
$$H_K(x_2) = E_K(1\|x_2)E_K(x_2\|0) = E_K(1^n)E_K(1^{n-1}0) = c_1^2 c_2^2 \quad (2)$$

From this we have $\Pr[c_1^1 = c_2^2 \mid \text{Enc is } H_K(x)] = 1$ and $\Pr[c_1^1 = c_2^2 \mid \text{Enc is random}] = 2^{-(n-1)}$ Therefore, $H_K(x)$ is trivially broken and not a good PRF.

Proof of $\mathbf{G_k(x)}$ security:

We want to prove that the PRF security of G implies the PRF security of E, therefore we take the contrapositive and prove that the given an efficient adversary A breaking the PRF security of E, we will construct an efficient adversary B that breaks the PRF security of G.



B:
Run A to get b'
output b'

$G_{0,1}$:
$k \xleftarrow{\$} \mathcal{K}$; $f \xleftarrow{\$} F(n)$
$b' \leftarrow A^{Enc}$
Procedure Enc(x)
$c_1 \leftarrow Ev(1||x)$
$c_2 \leftarrow Ev(0||x)$
return $c_1||c_2$
Procedure $\boldsymbol{Ev(y)}$
return $\mathcal{E}_k(y)$
return $f(y)$

G_2 :
$b' \leftarrow A^{Enc}$
Procedure Enc(x)
$c_1 \xleftarrow{\$} (1||x)$
$c_2 \xleftarrow{\$} (0||x)$
return $c_1||c_2$

G_3 :
$b' \leftarrow A^{Enc}$
Procedure Enc(x)
$y \xleftarrow{\$} \{0,1\}^{2n}$
return y

Since Game 0 is an implementation of the Real Room we have that $\Pr[G_0 \Rightarrow 1] = 1$. Furthermore, since Game 1 is an implementation of the Random Room and function f outputs a random string, but remembers previous inputs, $\Pr[G_1 \Rightarrow 1] = 2^{-2n}$. Therefore, the advantage between the two games is $\Pr[G_0 \Rightarrow 1]$ - $\Pr[G_1 \Rightarrow 1] = 1 - 2^{-2n} \leq \mathrm{Adv}_E^{PRF}(B)$. The $\Pr[G_2 \Rightarrow 1] = 0$ because \$(.) outputs a random string on every query. Finally, $\Pr[G_3 \Rightarrow 1] = 0$ because Game 3 just outputs a random string. It follows that given an adversary that breaks E, we can break the PRF security of G, thus proving G is secure.