HW1: Symetric Cryptography

This homework is due before Wednesday, March 6th at 8am.
You can either hand your copy before class or submit it by email at:
`alain.passelegue@ens-lyon.fr;arthur.herledan_le_merdy@ens-lyon.fr`
Late submissions will receive a 2-point penalty for each day after the due date.

1 From a 1-bit stretch PRG to a full PRF

Let $G : \{0,1\}^k \to \{0,1\}^{k+1}$ be a secure pseudo-random generator.

1. Let $\ell < k+1$ and define $G_\ell : \{0,1\}^k \to \{0,1\}^\ell$ such that $G_\ell(x) = [G(x)]_{1\dots\ell}$, where this denotes the first ℓ bits of $G(x)$. Prove that G_ℓ satisfies the security notion of a PRG (pseudorandomness). Is G_ℓ a PRG?

2. Consider $G^{(1)} : \{0,1\}^k \to \{0,1\}^{k+2}$ defined as follows. On input $x \in \{0,1\}^k$, algorithm $G^{(1)}$ first evaluates $G(x)$ and obtains $(x^{(1)}, y^{(1)}) \in \{0,1\}^k \times \{0,1\}$ such that $G(x) = x^{(1)} \parallel y^{(1)}$. It then evaluates G on $x^{(1)}$ and eventually returns $G(x^{(1)}) \parallel y^{(1)}$. Show that if G is a secure PRG, then so is $G^{(1)}$.

3. (a) Let $n \geq 1$. Propose a construction of a PRG $G^{(n)} : \{0,1\}^k \to \{0,1\}^{k+n+1}$ based on G. Show that if G is a secure PRG, then so is $G^{(n)}$.

(b) Evaluate the cost of your construction.

As a result, we just showed that you can construct a length-doubling PRG from a PRG with 1-bit stretch.
Let $G : \{0,1\}^s \to \{0,1\}^{2s}$ be a secure length-doubling PRG. The Goldreich-Goldwasser-Micali construction shows how to build a secure Pseudo-Random Function for any input size from G.

4. Let us denote $G(k) =: G_0(k)||G_1(k)$ for any $k \in \{0,1\}^s$ where $G_0, G_1 : \{0,1\}^s \to \{0,1\}^s$. Define $F_0 : \{0,1\}^s \times \{0,1\} \to \{0,1\}^s$ such that:

$$\forall k \in \{0,1\}^s, \forall b \in \{0,1\}, F_0(k,b) := G_b(k).$$

Prove that F_0 is a secure PRF.

We now expand our construction to arbitrary input size n. Define the iterated PRF $F_n : \{0,1\}^s \times \{0,1\}^n \to \{0,1\}^s$ that does the following: on inputs k and $x = x_0 x_1 \dots x_{n-1}$, define $k_0 := k$ and compute recursively $k_i := G_{x_{i-1}}(k_{i-1})$ for $i = 1$ to n. Finally output k_n.
Remark: This can be seen as going down a binary tree.

5. Before proving the security of F_n, we prove that the distribution $(G(k_1), G(k_2), \dots, G(k_Q))$, where $k_i \hookleftarrow U(\{0,1\}^s)$ is indistinguishable from $U(\{0,1\}^{2Qs})$ for any $Q = \mathsf{poly}(s)$, under the security of G.

We use the hybrid argument by defining the following hybrid distributions:

$$\forall i \in [0,Q], D_i := (G(k_1), \dots, G(k_i), U(\{0,1\}^{2s(Q-i)}) \text{ where } k_j \hookleftarrow U(\{0,1\}^s) \forall j \leq i.$$

Notice that D_0 and D_Q correspond to the distributions defined previously.

Prove that D_0 and D_Q are indistinguishable under the security of G. Estimate the security loss.

We move on to the proof that F_n is secure.

6. To do so, we use the hybrid argument by introducing the following hybrid experiments. Let us first define

$$F_{n,i}^{(R_i)} : (x_0, \ldots, x_{n-1}) \mapsto G_{x_{n-1}}(\ldots(G_{x_i}(R_i(x_0, \ldots x_{i-1})))),$$

where $R_i : \{0,1\}^i \to \{0,1\}^s$ is a map.

(a) Prove that $F_{n,0}^{(U(\{\varepsilon\} \to \{0,1\}^s))}(\cdot)$ is actually the distribution $F_n(U(\{0,1\}^s), \cdot)$.

(b) Prove that $F_{n,n}^{(U(\{0,1\}^n \to \{0,1\}^s))}$ is actually the distribution $U(\{0,1\}^n \to \{0,1\}^s)$.

(c) We define the hybrid experiment Exp_i for $i \in [1,n]$ as: the challenger flips a coin b and samples R uniformly over $\{0,1\}^{i-b} \to \{0,1\}^n$. The adversary is then given access to an oracle, which on query $x \in \{0,1\}^n$ answers with $F_{n,i-b}^{(R)}(x)$. Eventually, the adversary outputs a guess b' and wins if and only if $b = b'$.
Prove that the PRF F_n is secure under the security of the PRG G and estimate the advantage loss.

2 On weak PRFs and the DDH problem

In the PRF security game, the adversary may adaptively make function evaluation queries: for $i = 1, 2, \ldots$, it sends x_i of its choice, and gets $F_k(x_i)$ (resp. $f(x_i)$) from the challenger, where F_k is the PRF (resp. f is the uniformly chosen function). A weak-PRF consists of the same algorithms as a PRF, but the queries are modified as follows: the adversary does not get to see $F_k(x_i)$ (resp. $f(x_i)$) for **an input x_i of its choice**, but instead every time the adversary requests a new pair, **the challenger samples a fresh uniform** x_i and sends $(x_i, F_k(x_i))$ (resp. $(x_i, f(x_i))$) to the adversary.

7. Give a formal definition of a weak-PRF, based on a security game.

8. Show that a PRF is a weak-PRF, by providing a security reduction.

9. Assuming that a weak-PRF exists, build a weak-PRF that is not a PRF.

10. What is the difference between a PRG and a weak-PRF?

Let $G = (g)$ be a cyclic group of known prime order p. We recall that the DDH hardness assumption states that the distributions (g^a, g^b, g^{ab}) and (g^a, g^b, g^c) are computationally indistinguishable when a, b and c are independently and uniformly distributed in $\mathbb{Z}/p\mathbb{Z}$. Let $k \in \mathbb{Z}/p\mathbb{Z}$ a uniformly chosen key. We consider the function $F_k : h \in G \mapsto h^k \in G$.

11. Let $Q \geq 1$. Consider the (randomized) map ϕ that takes $(g_1, g_2, g_3) \in G^3$ as input, samples $(x_i, y_i) \in (\mathbb{Z}/p\mathbb{Z})^2$ uniformly and independently for $i \leq Q$ and returns $(g_1^{x_i} g^{y_i}, g_3^{x_i} g_2^{y_i})_{i \leq Q}$.

- Show that if $(g_1, g_2, g_3) = (g^a, g^b, g^{ab})$, then the output is distributed as $(g^{r_i}, g^{br_i})_{i \leq Q}$ for r_i's in $\mathbb{Z}/p\mathbb{Z}$ uniform and independent.
- Show that if $(g_1, g_2, g_3) = (g^a, g^b, g^c)$ for $c \neq ab$, then the output is distributed as $(g^{r_i}, g^{s_i})_{i \leq Q}$ for (r_i, s_i)'s in $(\mathbb{Z}/p\mathbb{Z})^2$ uniform and independent.

12. Show that F_k is a weak-PRF under the DDH hardness assumption.
Hint: set "$k = b$" and use the previous question to build the weak PRF challenger.

13. Is F_k a secure PRF? Justify your answer.

3 A CCA-secure symmetric encryption scheme

Consider the following construction of symmetric encryption, where $\Pi = (\mathsf{Gen}, \mathsf{Mac}, \mathsf{Verify})$ is a MAC.

Gen(1^λ)**:** Choose a random key $K_1 \leftarrow \mathsf{Gen}'(1^\lambda)$ for an IND-CPA secure symmetric encryption scheme $(\mathsf{Gen}', \mathsf{Enc}', \mathsf{Dec}')$. Choose a random key $K_0 \leftarrow \Pi.\mathsf{Gen}(1^\lambda)$ for the MAC Π. The secret key is $K = (K_0, K_1)$.

Enc(K, M)**:** To encrypt M, do the following.

1. Compute $c = \mathsf{Enc}'(K_1, M)$.
2. Compute $t = \Pi.\mathsf{Mac}(K_0, c)$.

Return $C = (t, c)$.

Dec(K, C)**:** Return $\perp$ if $\Pi.\mathsf{Verify}(K_0, c, t) = 0$. Otherwise, return $M = \mathsf{Dec}'(K_1, c)$.

14. Assume that the MAC is weakly unforgeable. Assume however that there exists an algorithm $\mathcal{F}$, which on input a valid message for the MAC and a tag (M, t), outputs a forgery (M, t') such that $t \neq t'$. In particular, the MAC is not strongly unforgeable. Show that the scheme is not IND-CCA secure.

15. We assume that: (i) $(\mathsf{Gen}', \mathsf{Enc}', \mathsf{Dec}')$ is IND-CPA-secure; (ii) Π is strongly unforgeable under chosen-message attacks. We will prove in this question the IND-CCA security of the new encryption scheme under these assumptions. Let $\mathcal{A}$ be an adversary against the IND-CCA security of the scheme.

(a) Define the event Valid as the event where $\mathcal{A}$ makes a valid (i.e. accepted by the MAC) decryption query for (c, t) where the ciphertext c was not encrypted by the encryption oracle nor is (c, t) the challenge ciphertext. Prove that if $\Pr(\mathsf{Valid})$ is non-negligible then there exists an adversary with non-negligible advantage against the strong unforgeability of the MAC.

The intuition is that since this event has negligible probability, the decryption oracle is useless to an attacker $\mathcal{A}$.

(b) Prove that if $|\Pr(\mathcal{A} \text{ wins} \wedge \overline{\mathsf{Valid}}) - 1/2|$ is non-negligible, then there exists an efficient adversary against the IND-CPA security of the encryption scheme $(\mathsf{Gen}', \mathsf{Enc}', \mathsf{Dec}')$.

(c) Conclude.