CS 601.442/642 – Modern Cryptography Instructors: Aditya Hegde and Harry Eldridge

Homework 6

Deadline: March 12; 2026, 11:59pm ET

Instructions

- The solutions must be submitted via Canvas.
- You must typeset your solutions. We suggest using LaTeX or Typst.

Problems

1. Recall that the "Textbook RSA" encryption scheme we saw in class is not a secure public-key encryption scheme. In this problem, you will prove CPA-security of a public-key encryption scheme based on RSA.

   Specifically, let $\mathsf{GenRSA}(1^\lambda) \to (p, q)$ be a PPT algorithm that takes the security parameter as input and outputs distinct odd primes p and q. The RSA assumption is said to hold with respect to GenRSA if for all non-uniform PPT adversaries $\mathcal{A}$, there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$,

   $$\Pr\left[\mathcal{A}(N, e, x^e \bmod N) = x : \begin{array}{r} (p,q) \leftarrow \mathsf{GenRSA}(1^\lambda) \\ N := pq \\ x \leftarrow \mathbb{Z}_N^\times \\ e \leftarrow \mathbb{Z}_{\varphi(N)}^\times \end{array}\right] \leq \mathsf{negl}(\lambda),$$

   where $\varphi(N) = (p-1)(q-1)$.

   We saw in class that $\left\{f_{N,e}(x) := x^e \bmod N : (p,q) \leftarrow \mathsf{GenRSA}(1^\lambda), N := pq, e \leftarrow \mathbb{Z}_{\varphi(N)}^\times\right\}$ is a OWP from $\mathbb{Z}_N^\times \mapsto \mathbb{Z}_N^\times$ and that $\mathsf{hc}(x) :=$ Least-Significant-Bit(x) is a hard-core predicate for f.

   Consider the encryption algorithm Enc for 1-bit messages: $\mathsf{Enc}(\mathsf{pk}, m)$ first parses $\mathsf{pk} = (N, e)$. It then samples $r \leftarrow \mathbb{Z}_N^\times$ and outputs the ciphertext $\mathsf{ct} := (f_{N,e}(r), \mathsf{hc}(r) \oplus m)$.

   (a) (5 points) Construct appropriate $\mathsf{KeyGen}(1^\lambda)$ and $\mathsf{Dec}(\mathsf{sk}, \mathsf{ct})$ algorithms to complete the public-key encryption scheme $\Pi = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$.

   (b) (5 points) Prove that Π is correct, i.e., that decryption recovers the plaintext.

   (c) (10 points) Using the fact that $f_{N,e}$ is a OWP and hc is a hard-core predicate for f, prove that Π is CPA-secure.

2. (20 points) Let $G : \{0,1\}^\lambda \to \{0,1\}^{2\lambda}$ be a secure length-doubling PRG and write $G(s) = G_0(s) \| G_1(s)$, where $G_0(s), G_1(s) \in \{0,1\}^\lambda$ for all $s \in \{0,1\}^\lambda$. Consider the GGM construction applied to *variable-length inputs*. That is, let $\ell := \ell(\lambda)$ be a polynomially bounded integer. Define $F_k : \{0,1\}^{\leq \ell} \to \{0,1\}^\lambda$ as

   $$F_k(x) = G_{x_n}(G_{x_{n-1}}(\ldots G_{x_1}(k))),$$

   where $x = x_1 \| \ldots \| x_{n-1} \| x_n$ are the bits of x and $\{0,1\}^{\leq \ell} = \cup_{i=1}^{\ell} \{0,1\}^i$.

   Is F a secure PRF? Prove your answer.

3. (30 points) An efficiently-computable[1] family of functions $\left\{\mathsf{hc}_\lambda : \{0,1\}^\lambda \to \{0,1\}\right\}_\lambda$ is called a *universal* hard-core predicate if for every one-way function f, hc is a hard-core predicate for f.

   Prove that there is no universal hardcore predicate.

   Hint: Assume for the sake of contradiction that a universal hard-core predicate hc exists. Consider an arbitrary OWF f. Since hc is universal, it is a hard-core predicate for f. Can you construct a OWF g using f and hc such that hc cannot be the hard-core predicate for g? Make sure to argue both that g is a OWF and that hc is not a hard-core predicate for g.

4. (30 points) In class, we saw that if OWFs exist, then PRGs can be constructed from them. In this problem, you will prove the *converse*: if PRGs exist then OWFs exist. In fact, you will show that a PRG is already a OWF.

   Specifically, let $G : \{0,1\}^\lambda \to \{0,1\}^{\lambda+1}$ be a PRG. Prove that G is a OWF.

   Hint: Consider the image of G, i.e., $\left\{G(s) : s \in \{0,1\}^\lambda\right\} \subseteq \{0,1\}^{\lambda+1}$. What fraction of $\{0,1\}^{\lambda+1}$ does this set occupy? How does this help you when analyzing the behavior of the distinguisher you construct in your reduction on a uniformly random string versus a string in the image of G?

[1] Here efficiently-computable means that each function hc_λ can be computed in time polynomial in λ.