CS783: Theoretical Foundations of Cryptography Fall 2024

Assignment 1

Instructor: Chethan Kamath

Exercise 1 (Classical ciphers [KL14])**.** *Let's understand the conditions under which some of the classical ciphers we discussed in Lecture 2 become perfectly secure.*

1. *Show that (monoalphabetic) shift cipher is perfectly secure for messages of length one, i.e., message-space* $\{\mathsf{a}, \cdots, \mathsf{z}\}$.

2. *What is the maximum message-space for which (monoalphabetic) substitution cipher is perfectly secure?*

Exercise 2 (Statistical secrecy)**.** *An SKE* $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *is said to be statistically-secret if for* every *eavesdropper* Eve

$$\delta(n) := \left| \Pr_{\substack{(m_0,m_1)\leftarrow\mathsf{Eve}(1^n)\\ k\leftarrow\mathsf{Gen}(1^n)\\ c\leftarrow\mathsf{Enc}(k,m_0)}}[\mathsf{Eve}(c)=0] - \Pr_{\substack{(m_0,m_1)\leftarrow\mathsf{Eve}(1^n)\\ k\leftarrow\mathsf{Gen}(1^n)\\ c\leftarrow\mathsf{Enc}(k,m_1)}}[\mathsf{Eve}(c)=0] \right|$$

is negligible (as defined in Lecture 3). Since we allow a slack, this is a weaker requirement than perfect secrecy. Does Shannon's impossibility still extend to statistically-secret SKE schemes?

Exercise 3 (One-time pad (OTP))**.** *Recall the definition of OTP from Lecture 2.*

1. *The goal of this exercise is to help you understand more about randomness in encryption algorithm. Recall that the encryption algorithm of OTP is deterministic. Modify OTP to come up with* two *perfectly-secure SKE schemes* PS_1 *and* PS_2 *that have* randomised *encryption algorithm,* and *such that leaking the random coins used in encryption leads (a)* PS_1 *to become insecure (b)* PS_2 *to remain secure.*

2. *Let's consider OTP against a tampering adversary* Tam *who can modify a ciphertext* c *of some message* $m = m_0 \cdots m_{\ell-1} \in \{0,1\}^\ell$ *before it reaches the recipient, Caeser's general. Can* Tam *tamper* c *to some ciphertext* c' *such that Caeser's general decrypts* c' *to the following. If your answer is 'yes', then describe* Tam*; if it is 'no', justify.*

   (a) $m \oplus (110^{\ell-2})$*, i.e.,* m *with first two bit flipped (asssume* $\ell \geq 2$*)*

   (b) 0^n

   (c) $m_1 m_0 m_2 \cdots m_{\ell-1}$*, i.e., the first two bits of* m *swapped*

3. *Suppose an OTP key is used to encrypt two messages* m_0 *and* m_1 *of your choice. Is it possible to recover the key* with certainty*?*

Exercise 4 (Negligible functions). *Recall the definition of negligible functions from Lecture 3.*

1. *Are f_p and f_M negligible, and why? Here, a Mersenne prime is a prime of the form $M_n := 2^n - 1$.*

$$f_p(n) := \begin{cases} 1/n^{314159} & \text{if } n \text{ is a prime} \\ 1/2^n & \text{otherwise} \end{cases}$$

$$f_M(n) := \begin{cases} 1/n^{314159} & \text{if } M_n \text{ is a Mersenne prime} \\ 1/n^{\log(n)} & \text{otherwise} \end{cases}$$

2. *If ν_1 and ν_2 are negligible function, which of these following functions are also (always) negligible? In case the function is negligible, provide a security reduction; in case not, provide a counter-example.*

   (a) $f_+(n) := \nu_1(n) + \nu_2(n)$

   (b) $f_\times(n) := \nu_1(n) \times \nu_2(n)$

   (c) $f_\div(n) := \nu_1(n) \div \nu_2(n)$

Exercise 5 (Message length in definitions [KL14]). *Recall the definition of adversarial indistinguishability (Definition 4) from Lecture 3. I didn't stress during the lecture that the messages m_0, m_1 that* Eve *outputs* must *be of same length (i.e., $|m_0| = |m_1|$). You will try to understand why through this exercise. Prove that a Π that supports arbitrary length messages (i.e., the message-space is $\{0,1\}^*$)* cannot *satisfy adversarial indistinguishability if* Eve *is not restricted to challenging on equal length messages.*

Exercise 6 (PRGs). *Recall the definition of PRGs from Lecture 3.*

1. *Let G be a PRG that stretches from n bits to $n+1$ bits. Which of the following candidates based on G are also (always) PRGs? In case your claim is that a candidate is a PRG, provide a proof; in case not, provide a counter-example and the efficient distinguisher.*

   (a) *Duplicating PRG: $\mathsf{G}_d(s) := s \| s$, where $\|$ denotes string concatenation*

   (b) *Leaky PRG: $\mathsf{G}_\ell(s \| b) := \mathsf{G}(s) \| b$, where $b \in \{0,1\}$*

   (c) *Complementary PRGs $\mathsf{G}_1(s) := \mathsf{G}(\overline{s})$ and $\mathsf{G}_2(s) := \overline{\mathsf{G}(s)}$, where for a bit-string s, $\overline{s}$ denotes bit-complement.*

   (d) *Singly punctured PRG:*

$$\mathsf{G}_p(s) := \begin{cases} 0^{|s|+1} & \text{if } s \text{ of the form } 0^{|s|} \\ \mathsf{G}(s) & \text{otherwise} \end{cases}$$

   (e) *Mildly punctured PRG:*

$$\mathsf{G}_m(s) := \begin{cases} 0^{|s|+1} & \text{if } s \text{ of the form } 0^{\lceil\sqrt{|s|}\rceil} \| \{0,1\}^* \\ \mathsf{G}(s) & \text{otherwise} \end{cases}$$

(f) Heavily punctured PRG:

$$\mathsf{G}_h(s) := \begin{cases} 0^{|s|+1} & \text{if } s \text{ of the form } 0^{\lfloor \log(|s|) \rfloor} \| \{0,1\}^* \\ \mathsf{G}(s) & \text{otherwise} \end{cases}$$

(g) Prefixing PRG: $\mathsf{G}_f(s) := \mathsf{G}(0^{|s|} \| s)$

2. *Recall the definition of computational indistinguishability (CI) at the end of Lecture 3. Show formally that CI is a transitive property. That is, if X_1 is CI from X_2, and X_2 is CI from X_3, then X_1 is CI also from X_3.*

References

[KL14] Jonathan Katz and Yehuda Lindell. *Introduction to Modern Cryptography (3rd ed.)*. Chapman and Hall/CRC, 2014.