∾ CS6222 Cryptography ∾

Topic: Computational Indistinguishability
Date: Sep 03, 2024
Lecturer: Wei-Kai Lin (TA: Arup Sarker)
Scriber: Jinye He

Recall the perfectly secure encryption, OTP. That is, we bitwise-XOR our message with a uniform random string.

$$m \oplus k,\ |m| = |k|.$$

OTP is inefficient because the long random string must be shared between Alice and Bob in advance. We also have shown that OTP is optimal for perfect secure. Next, we will focus on how to improve efficiency while moderately relaxing security.

Suppose that we have a (mathematical, deterministic) function that can extends a short truly random string to a long ”random-looking” string. We can use the seemingly random to encrypt messages as in OTP, yet it is efficient.

1 Adversaries: Non-Uniform

To formalize ”seemingly random”, we want to model adversaries with a stronger capability than honest though bounded.

Definition 1 (Non-uniform PPT)**.** A *non-uniform probabilistic polynomial-time machine* (abbreviated NUPPT) $\mathcal{A}$ is a sequence of algorithms $\mathcal{A} = \{\mathcal{A}_1, \mathcal{A}_2, \dots\}$ such that:

- $\mathcal{A}_i$ computes on inputs of length i, and
- exists a polynomial d s.t. the description size $|A_i| \le d(i)$ and the time $\mathcal{A}_i$ is also less than $d(i)$.

Alternatively, an NUPPT algorithm can be defined as a uniform PPT $\mathcal{A}$ that takes an additional advice string of poly length $d(i)$ for each input length i. Non-uniform gives adversaries extra power and models many real scenario, e.g., Adversary may have a list of known (plain, cipher) pairs.

2 Computational Indistinguishability

Recall the Turing Test proposed by Turing in 1950. When a machine and a human is indistinguishable in every human's prompts, we call it AI. The key idea of ”seemingly random” is if we have no way to show the difference from truly random by NUPPT, then we are satisfied. We call it computational indistinguishability. We will formalize the concept asymptotically.

Definition 2 (Ensembles of Probability Distributions)**.** A sequence of distributions $(X_1, X_2, \dots)$ is called an *ensemble* if for each i, X_i is a probability distribution over $\{0,1\}^*$

Definition 3 (Negligible)**.** A function $\epsilon : \mathbb{N} \to \mathbb{R}^+ \cup \{0\}$ is *negligible* if for every $c > 0$, there exists some $n_0 \in \mathbb{N}$ such that for all $n > n_0$, $\epsilon(n) < \frac{1}{n^c}$.

Intuitively, a negligible function is asymptotically smaller than the inverse of any fixed polynomial. Examples of negligible includes 2^{-n}, $n^{100} \cdot 2^{-n}$ and $2^{-\sqrt{n}}$.

Definition 4 (Computational Indistinguishability)**.** Let $\mathcal{X} = (X_1, X_2, \dots)$ and $\mathcal{Y} = (Y_1, Y_2, \dots)$ be ensembles where X_n, Y_n are distributions over $\{0,1\}^{\ell(n)}$ for some polynomial $\ell(\cdot)$. We say that $\mathcal{X}$ and $\mathcal{Y}$ are *computationally indistinguishable* (denoted by $\mathcal{X} \approx_c \mathcal{Y}$) if for all NUPPT D (called the "distinguisher"), there exists a negligible function ϵ such that $\forall n \in \mathbb{N}$,

$$|\Pr[t \leftarrow X_n, D(1^n, t) = 1] - \Pr[t \leftarrow Y_n, D(1^n, t) = 1]| < \epsilon(n)$$

The 1^n here is an all one string with n bits which provides extra advice for the NUPPT about the length of the input.

Examples. Let $\mathcal{Z} = (Z_1, Z_2, \dots)$ be an ensemble of probability distributions where Z_n is the uniform distribution of $\{0,1\}^n$.

- Let $\mathcal{X} = \mathcal{Z}$, $\mathcal{Y} = \mathcal{Z}$. Then $\mathcal{X}$ and $\mathcal{Y}$ are computational indistinguishable.
- Let $\mathcal{X} = \mathcal{Z}$, $\mathcal{Y} = (Z_1 \oplus 1^n, Z_2 \oplus 1^n, \dots)$. Then $\mathcal{X}$ and $\mathcal{Y}$ are computational indistinguishable.
- Let $\mathcal{X} = \mathcal{Z}$, $\mathcal{Y} = (Z_1 \vee 1^n, Z_2 \vee 1^n, \dots)$. Then $\mathcal{X}$ and $\mathcal{Y}$ are **not** computational indistinguishable. A NUPPT D can be construct to distinguish these two ensembles with non-negligible probability (for example: D output 1 when input is an all one string and output 0 otherwise).

Now we introduce our relaxed definition of secure encryption. We say an encryption scheme $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ is secure if for all $m_0, m_1 \in \mathcal{M}$, the ensembles $\mathcal{X} := \{\mathsf{Enc}_k(m_0) : k \leftarrow \mathsf{Gen}(1^n)\}_{n \in \mathbb{N}}$ and $\mathcal{Y} := \{\mathsf{Enc}_k(m_1) : k \leftarrow \mathsf{Gen}(1^n)\}_{n \in \mathbb{N}}$ are computational indistinguishable.

2.1 Properties of Computational Indistinguishability

Lemma 5 (Closure under NUPPT)**.** *If the pair of ensembles $\mathcal{X} \approx_c \mathcal{Y}$, then for any NUPPT M, $M(\mathcal{X}) \approx_c M(\mathcal{Y})$, where $M(\mathcal{X}) := \{\{M(t) : t \leftarrow X_n\}\}_{n \in \mathbb{N}}$ and $M(\mathcal{Y}) := \{\{M(t) : t \leftarrow Y_n\}\}_{n \in \mathbb{N}}$.*

Proof. We prove this lemma by reduction. Assume for contradiction that there exists an NUPPT $\mathcal{A}$ and a polynomial (positive when n tends to positive infinity) p such that for infinitely many $n \in \mathbb{N}$,

$$\Pr[t \leftarrow M(X_n), \mathcal{A}(1^n, t) = 1] - \Pr[t \leftarrow M(Y_n), \mathcal{A}(1^n, t) = 1] \geq \frac{1}{p(n)}.$$

We can construct a NUPPT $\mathcal{B}$ such that $\mathcal{B}(1^n, u)$ outputs $\mathcal{A}(1^n, M(u))$. Then we show that the NUPPT $\mathcal{B}$ can distinguish $\mathcal{X}$ and $\mathcal{Y}$.

$$\begin{aligned} &\Pr[u \leftarrow X_n, \mathcal{B}(1^n, u)] - \Pr[u \leftarrow Y_n, \mathcal{B}(1^n, u)] \\ =&\Pr[u \leftarrow X_n, \mathcal{A}(1^n, M(u))] - \Pr[u \leftarrow Y_n, \mathcal{A}(1^n, M(u))] \\ =&\Pr[t \leftarrow M(X_n), \mathcal{A}(1^n, t)] - \Pr[t \leftarrow M(Y_n), \mathcal{A}(1^n, t)]. \end{aligned}$$

Thus, the difference between $\Pr[u \leftarrow X_n, \mathcal{B}(1^n, u)]$ and $\Pr[u \leftarrow Y_n, \mathcal{B}(1^n, u)]$ is greater than $\frac{1}{p(n)}$ for infinitely many $n \in \mathbb{N}$ which is contradict with $\mathcal{X} \approx_c \mathcal{Y}$. □

References

[KL21] Jonathan Katz and Yehuda Lindell. *Introduction to Modern Cryptography.* CRC Press, Massachusetts, 3 edition, 2021.

[LS] Wei-kai Lin and Arup Sarker. weikailin/cs6222: Cryptography.

[PS10] Rafael Pass and Abhi Shelat. *A Course In Cryptography*. Lecture notes available atcs. cornell/courses/cs4830/2010fa/lecnotes.pdf, 2010.

[Wik] Wikipedia. Monty hall problem.