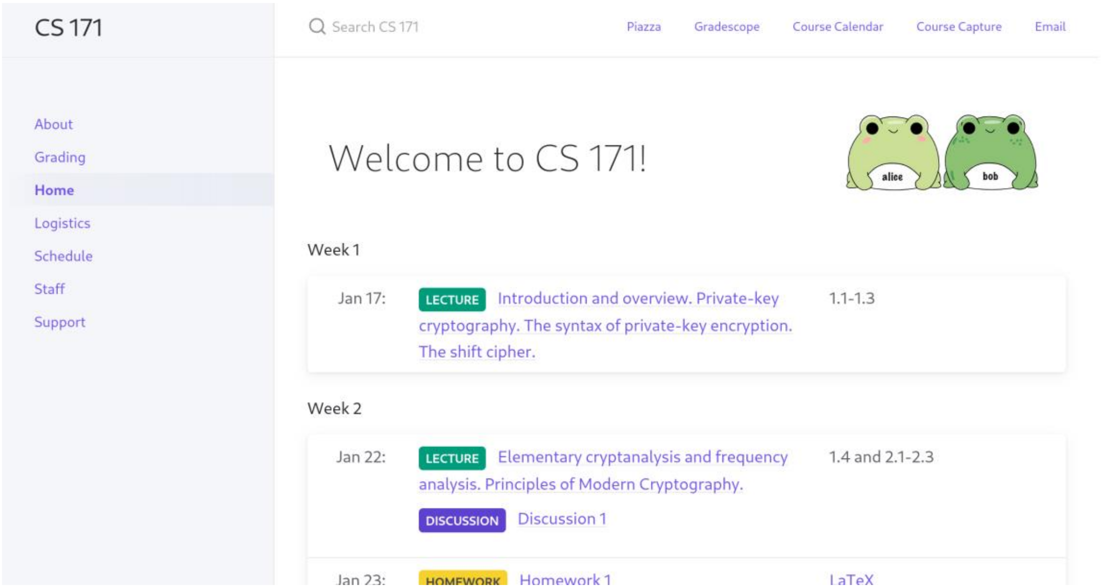
CS171: Cryptography

Lecture 3

Sanjam Garg

https://eecs171.com/



Email for Course Staff: cs171@berkeley.edu

Defining Secure Encryption: Formally

Definition 1: An encryption scheme (Gen, Enc, Dec) with message space $\mathcal{M}$ is *perfectly secret* if for every probability distribution over $\mathcal{M}$, every message $m \in \mathcal{M}$, and every ciphertext c for which $\Pr[C = c] > 0$:

$$\Pr[M = m \mid C = c] = \Pr[M = m]$$

Or, if for every two messages , $m, m' \in \mathcal{M}$, and every ciphertext c (in ciphertext space):

$$\Pr[Enc_K(m) = c] = \Pr[Enc_K(m') = c],$$

Definition 3 (Game Style)

eav is for Eavesdropper

$\text{PrivK}^{\text{eav}}_{A,\Pi}$

1. A outputs $m_0, m_1 \in \mathcal{M}$.
2. $\text{b} \leftarrow \{0,1\}$, $k \leftarrow \text{Gen}()$, $c \leftarrow Enc_k(m_b)$
3. c is given to A
4. A output b'
5. Output 1 if $b = b'$ and 0 otherwise

Challenge ciphertext

Encryption scheme $\Pi = (Gen, Enc, Dec)$ with message space $\mathcal{M}$

is **perfectly indistinguishable if**

$\forall A$ it holds that:

$$\Pr\left[\text{PrivK}^{\text{eav}}_{A,\Pi} = 1\right] = \frac{1}{2}$$

A can always succeed with probability ½. How?

Lemma (Prove on your own): Encryption scheme Π is *perfectly secret* if and only if it is *perfectly indistinguishable*.

The One-Time Pad

Fix an integer ℓ, then let $\mathcal{M}, \mathcal{K}, \mathcal{C} = \{0,1\}^{\ell}$

- Gen: output a uniform value from $\mathcal{K}$
- $Enc_k(m)$: where $m \in \{0,1\}^{\ell}$, output $c := k \oplus m$
- $Dec_k(c)$: output $m := k \oplus c$
- Correctness: $Dec_k\big(Enc_k(m)\big) = k \oplus k \oplus m = m$
- Security: $\forall\, m, c, \Pr[Enc_K(m) = c] = 2^{-\ell}$. Or,
  $\forall\, m, m', c, \Pr[Enc_K(m) = c] = \Pr[Enc_K(m') = c]$

One-Time Pad: Good and Bad

- One-Time Pad achieves perfect security
  - Been used in the past

- Not used anymore, why not?
  1. The key is as long as the message
  2. Can’t reuse the key
  3. Broken under known-plaintext attack

Can we make $|\mathcal{M}| > |\mathcal{K}|$?

Optimality of One-Time Pad

Theorem: If $\Pi = (Gen, Enc, Dec)$ is a perfectly secret encryption scheme with message space $\mathcal{M}$ and key space $\mathcal{K}$, then $|\mathcal{M}| \leq |\mathcal{K}|$.

1. Assume $|\mathcal{K}| < |\mathcal{M}|$ (will show that Π cannot be perfectly secret)
2. $\mathcal{M}(c) = \{m \mid m = Dec_k(c) \, for \; some \; k \in \mathcal{K}\}$
3. $|\mathcal{M}(c)| \leq \mathcal{K}$
4. $\exists m' \in \mathcal{M}, \; m' \notin \mathcal{M}(c)$
5. $\Pr[M = m' \mid C = c] = 0 \neq \Pr[M = m']$

Computational Security

- Relaxation of perfect security
  - Security only against efficient adversaries
  - Security can fail with some very small probability

- Two approaches
  - Concrete security
  - Asymptotic security

Concrete Security

- A scheme is *(t, ϵ)-secure* if for any adversary running for time at most t succeeds in breaking the scheme with probability at most ϵ.
- Example: Consider an encryption scheme that is $(2^{128}, 2^{-60})$ −secure.
- 2^{80} is the computation that can be performed by super-computers in one year or so.
- 2^{-60} is the probability that an event happens roughly once every 100 billion years

What’s wrong?

- Concrete security is essential in choosing scheme parameters in practice.

- However, it doesn’t yield clean theory
  - Depends on the computational model
  - Need to change schemes as (t, ϵ) need to be updated

- Need schemes that allow tuning (t, ϵ) as desired

Asymptotic Security

- Introduce a security parameter n (known to adversary)
- All honest parties run in polynomial time in n

- Security can be tuned by changing n
  - t and ϵ are now functions of n
  - t -> probabilistic polynomial time (PPT) in n
  - ϵ -> a negligible function in n

Polynomial and Negligible

- A function $f\colon Z^{+} \to Z^{+}$ is *polynomial* if there exists c such that $f(n) < n^c$ for large enough n
- A function $f\colon Z^{+} \to [0,1]$ is *negligible* if ∀ polynomial p it holds that $f(n) < 1/p(n)$ for large enough n
  - Typical example: $f(n) = poly(n) \cdot 2^{-\alpha n}$

Negligible Function (formally)

- A function $f\colon Z^{+} \rightarrow [0,1]$ is *negligible* if $\forall$ polynomial p it holds that $\exists\ N \in Z^{+}\ \forall\ n > N$ (for large enough n) we have $f(n) < 1/p(n)$
  - $\forall\, p\ \exists\ N \in Z^{+}\ \forall\ n > N, f(n) < 1/p(n)$

- Prove that 2^{-n} is a *negligible* function

Is this a negligible function?

- $f(n) = 2^{-\sqrt{n}}$
- $f(n) = n^{-\log n}$

- $f(n) = 2^{-n}$ for n mod 2 = 0

  $= n^{-c}$ for n mod 2 = 1

Choice of Polynomial and Negligible

- Using PPT for efficient machines is borrowed from complexity theory

- Also some nice closure properties:
  - $poly(n) \cdot\ poly(n)$ is still $poly(n)$
  - $poly(n) \cdot negl(n)$ is still $negl(n)$

Concrete vs Asymptotic

A scheme is (t, ϵ)-*secure* if for any adversary running for time at most t succeeds in breaking the scheme with probability at most ϵ.

↓

A scheme is *secure* if any PPT adversary succeeds in breaking the scheme with probability at most negligible.

Defining Computationally Secure Encryption (syntax)

- A *private-key encryption scheme* is a tuple of algorithms (Gen, Enc, Dec):
  - $Gen(1^n)$: outputs a key k (assume $|k| > n$)
  - Enc_k(m): takes key k and message $m \in \{0,1\}^*$ as input; outputs ciphertext c
    $$c \leftarrow Enc_k(m)$$
  - Dec_k (c): takes key k and ciphertext c as input; outputs m or "error"
    $$m := Deck(c)$$

Correctness: For all n, k output by $Gen(1^n)$, $m \in \{0,1\}^*$ it holds that $Dec_k(Enc_k(m)) = m$

Computational Indistinguishability

$\mathrm{PrivK}^{\mathrm{eav}}_{\mathrm{A},\Pi}$ (n)

1. A outputs $m_0, m_1 \in$ ~~$\mathcal{M}$.~~ $\{0,1\}^*, |m_0| = |m_1|$
2. b ← {0,1}, $k \leftarrow$ Gen(1^n), $c \leftarrow Enc_k(m_b)$
3. c is given to A
4. A output b'
5. Output 1 if $b = b$' and 0 otherwise

Encryption scheme $\Pi = (Gen, Enc, Dec)$ with ~~message space $\mathcal{M}$~~ is ~~perfectly~~ computationally indistinguishable if $\forall$ PPT A it holds that:

$$\Pr\left[\mathrm{PrivK}^{\mathrm{eav}}_{\mathrm{A},\Pi}(n) = 1\right] \leq \frac{1}{2} + \mathrm{negl}(n)$$

Does not hide message length! A scheme that only supports messages of fixed length is called a fixed-length encryption scheme.

Distinguishing variant

$\mathrm{PrivK}^{\mathrm{eav}}_{\mathrm{A},\Pi}$ (n, d)

1. A outputs $m_0, m_1 \in \{0,1\}^*, |m_0| = |m_1|$.
2. b = d , $k \leftarrow \mathrm{Gen}(1^n)$, $c \leftarrow Enc_k(m_b)$
3. c is given to A
4. A output b'
5. Output 1 if $b = b'$ and 0 otherwise

The output of A is $\mathrm{out}_{\mathrm{A}}\left(\mathrm{PrivK}^{\mathrm{eav}}_{\mathrm{A},\Pi}(1^n, d)\right)$

Π is computationally indistinguishable if

$\forall$ $PPT\ A$ it holds that:

$\left|\Pr\left[\mathrm{out}_{\mathrm{A}}\left(\mathrm{PrivK}^{\mathrm{eav}}_{\mathrm{A},\Pi}(1^n, 1)\right) = 1\right] - \Pr\left[\mathrm{out}_{\mathrm{A}}\left(\mathrm{PrivK}^{\mathrm{eav}}_{\mathrm{A},\Pi}(1^n, 0)\right) = 1\right]\right| \leq$ negl(n).

- Here, $\mathrm{PrivK}^{\mathrm{eav}}_{\mathrm{A},\Pi}(1^n, d)$ is same as $\mathrm{PrivK}^{\mathrm{eav}}_{\mathrm{A},\Pi}(1^n)$ except that we set $b = d$.

Thank You!