Cryptography

Lecture 5

Announcements

- HW1 due Monday, 2/12
- HW2 posted, Monday, 2/26

Agenda

- Last time:
  - Indistinguishability in the presence of an eavesdropper (K/L 3.2)
  - Defining PRG (K/L 3.3)
- This time:
  - Constructing computationally secure SKE from PRG (K/L 3.3)
  - Security Proof (K/L 3.3)
  - Class Exercise on PRG's

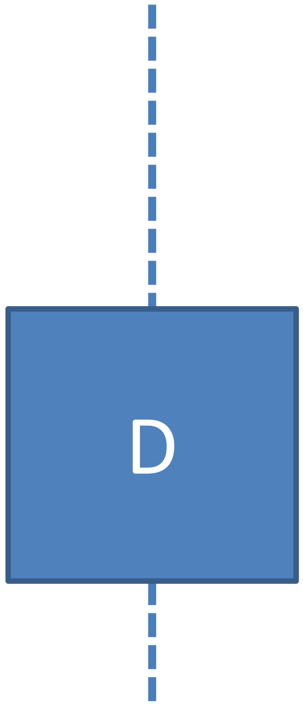
Pseudorandom Generator

- Functionality
  - Deterministic algorithm G
  - Takes as input a short random seed s
  - Ouputs a long string $G(s)$
- Security
  - No efficient algorithm can "distinguish" $G(s)$ from a truly random string r.
  - i.e. passes all "statistical tests."
- Intuition:
  - Stretches a small amount of true randomness to a larger amount of pseudorandomness.
- Why is this useful?
  - We will see that pseudorandom generators will allow us to beat the Shannon bound of $|\boldsymbol{K}| \geq |\boldsymbol{M}|$.
  - I.e. we will build a computationally secure encryption scheme with $|\boldsymbol{K}| < |\boldsymbol{M}|$

Pseudorandom Generator (PRG)

$r \in \{0,1\}^{\ell(n)}$

Truly random bit string r of length $\ell(n)$ is sampled and given to D.



$s \in \{0,1\}^n, G(s)$

Truly random bit string s of length n is sampled. $G(s)$ is given to D.

PRF: Any efficient D cannot tell which world it is in.

$$|\Pr[D(r) = 1] - \Pr[D(G(s)) = 1]| \leq negligible$$

Pseudorandom Generators

Definition: Let $\ell(\cdot)$ be a polynomial and let G be a deterministic poly-time algorithm such that for any input $s \in \{0,1\}^n$, algorithm G outputs a string of length $\ell(n)$. We say that G is a pseudorandom generator if the following two conditions hold:

1. (Expansion:) For every n it holds that $\ell(n) > n$.
2. (Pseudorandomness:) For all ppt distinguishers D, there exists a negligible function $negl$ such that:

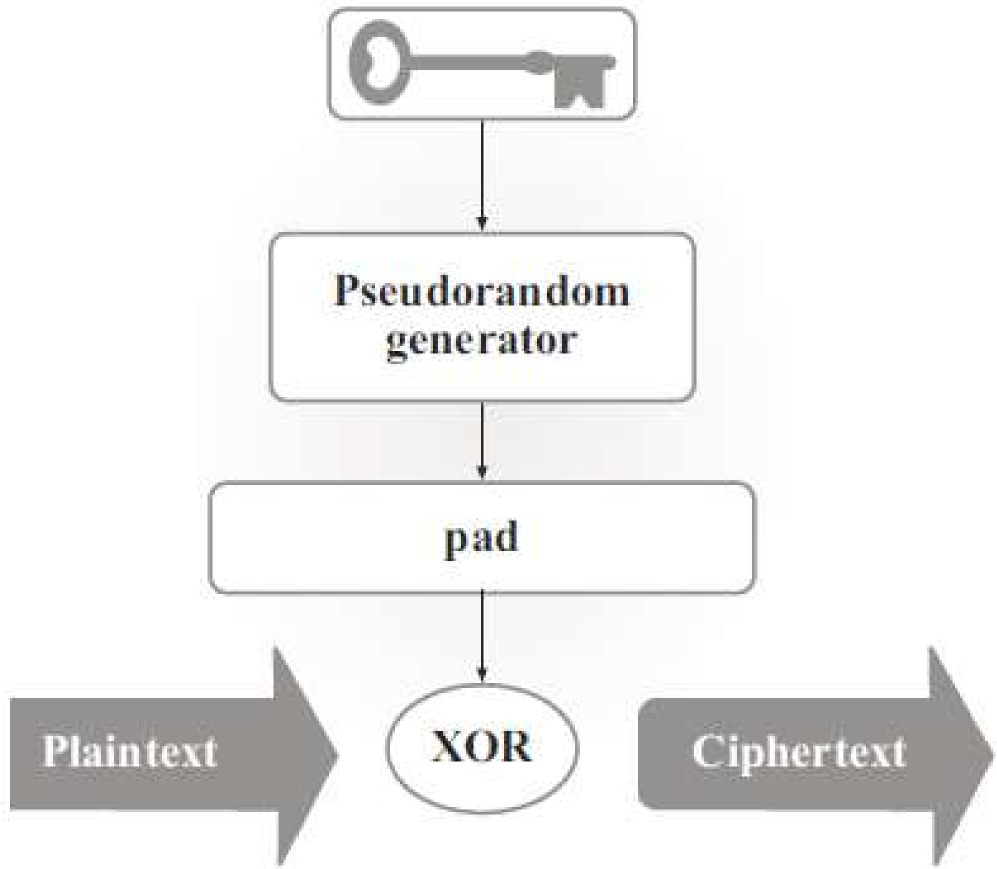
$$\left|\Pr[D(r) = 1] - \Pr\left[D\big(G(s)\big) = 1\right]\right| \leq negl(n),$$

where r is chosen uniformly at random from $\{0,1\}^{\ell(n)}$, the seed s is chosen uniformly at random from $\{0,1\}^n$, and the probabilities are taken over the random coins used by D and the choice of r and s.

The function $\ell(\cdot)$ is called the expansion factor of G.

Constructing Secure Encryption Schemes

A Secure Fixed-Length Encryption Scheme



The Encryption Scheme

Let G be a pseudorandom generator with expansion factor ℓ. Define a private-key encryption scheme for messages of length ℓ as follows:

- Gen: on input 1^n, choose $k \leftarrow \{0,1\}^n$ uniformly at random and output it as the key.
- Enc: on input a key $k \in \{0,1\}^n$ and a message $m \in \{0,1\}^{\ell(n)}$, output the ciphertext
$$c \coloneqq G(k) \oplus m.$$
- Dec: on input a key $k \in \{0,1\}^n$ and a ciphertext $c \in \{0,1\}^{\ell(n)}$, output the plaintext message
$$m \coloneqq G(k) \oplus c.$$

Security Analysis

Theorem: If G is a pseudorandom generator, then the Construction above is a fixed-length private-key encryption scheme that has indistinguishable encryptions in the presence of an eavesdropper.

Indistinguishability in the presence of an eavesdropper

Definition: A private key encryption scheme $\Pi = (Gen, Enc, Dec)$ has indistinguishable encryptions in the presence of an eavesdropper if for all probabilistic polynomial-time adversaries A there exists a negligible function $negl$ such that

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

Where the prob. Is taken over the random coins used by A, as well as the random coins used in the experiment.

Pseudorandom Generators

Definition: Let $\ell(\cdot)$ be a polynomial and let G be a deterministic poly-time algorithm such that for any input $s \in \{0,1\}^n$, algorithm G outputs a string of length $\ell(n)$. We say that G is a pseudorandom generator if the following two conditions hold:

1. (Expansion:) For every n it holds that $\ell(n) > n$.
2. (Pseudorandomness:) For all ppt distinguishers D, there exists a negligible function $negl$ such that:

$$\left|\Pr[D(r) = 1] - \Pr\left[D\left(G(s)\right) = 1\right]\right| \leq negl(n),$$

where r is chosen uniformly at random from $\{0,1\}^{\ell(n)}$, the seed s is chosen uniformly at random from $\{0,1\}^n$, and the probabilities are taken over the random coins used by D and the choice of r and s.

The function $\ell(\cdot)$ is called the expansion factor of G.

Security Analysis

- Proof by reduction method.

Security Analysis

Proof: Let A be a ppt adversary trying to break the security of the construction. We construct a distinguisher D that uses A as a subroutine to break the security of the PRG.

Distinguisher D:

D is given as input a string $w \in \{0,1\}^{\ell(n)}$.

1. Run $A(1^n)$ to obtain messages $m_0, m_1 \in \{0,1\}^{\ell(n)}$.
2. Choose a uniform bit $b \in \{0,1\}$. Set $c := w \oplus m_b$.
3. Give c to A and obtain output b'. Output 1 if $b' = b$, and output 0 otherwise.

Security Analysis

Consider the probability D outputs 1 in the case that w is random string r vs. w is a pseudorandom string $G(s)$.

- When w is random, D outputs 1 with probability exactly ½. Why?
- When w is pseudorandom, D outputs 1 with probability $\Pr\left[PrivK^{eav}{}_{A,\Pi}(n) = 1\right] = \frac{1}{2} + \rho(n)$, where ρ is non-negligible.

Security Analysis

D's distinguishing probability is:

$$\left|\frac{1}{2} - \left(\frac{1}{2} + \rho(n)\right)\right| = \rho(n).$$

This is a contradiction to the security of the PRG, since ρ is non-negligible.