Introduction to Modern Cryptography

Michele Ciampi

(Slides courtesy of Prof. Jonathan Katz)

Lecture 5, part 2

Pseudorandomness

Terminology: random vs. uniform

Random

Sample a **random element** according to **some distribution**

Uniform

Sample an element **uniformly at random** means to sample according to the **uniform distribution**

Informally

random $\approx$ **uniform**

Pseudorandom (informally)

pseudorandom $\approx$ *"looks like random"*

Pseudorandomness

- Important building block for **computationally secure** encryption
- Important concept in cryptography

What does *random* mean?

Uniform

- What does **uniform** mean?
- Which of the following is a uniform string?
  - `0101010101010101`
  - `0010111011100110`
  - `0000000000000000`

What does *random* mean?

Uniform

- What does **uniform** mean?
- Which of the following is a uniform string?
  - `0101010101010101`
  - `0010111011100110`
  - `0000000000000000`

If we generate a uniform $\mathbf{16}$-bit string, each of the above occurs with probability $\mathbf{2^{-16}}$

What does *uniform* mean?

Uniformity

- **Uniformity** is not a property of a string, but a property of a distribution
- A distribution on n-bit strings is a function $D : \{0,1\}^n \to [0,1]$ such that $\sum_x D(x) = 1$
- The **uniform distribution** on n-bit strings, denoted U_n, assigns probability 2^{-n} to every $x \in \{0,1\}^n$

What does *pseudorandom* mean?

Pseudorandom

- Cannot be distinguished from **uniform** (i.e. random)
- Which of the following is **pseudorandom**?
  - `0101010101010101`
  - `0010111011100110`
  - `0000000000000000`
- Pseudorandomness is a property of a distribution, not a string

Pseudorandomness (heuristic)

- Fix some distribution $\boldsymbol{D}$ on $\boldsymbol{n}$-bit strings
  - $\boldsymbol{x} \leftarrow \boldsymbol{D}$ means *sample x according to D*
- Historically, $\boldsymbol{D}$ was considered pseudorandom if it *passed a bunch of statistical tests*
  - $\mathbf{Pr}_{\boldsymbol{x} \leftarrow \boldsymbol{D}}[\textbf{1st bit of x is 1}] \approx \mathbf{1/2}$
  - $\mathbf{Pr}_{\boldsymbol{x} \leftarrow \boldsymbol{D}}[\textbf{parity of of x is 1}] \approx \mathbf{1/2}$
  - $\mathbf{Pr}_{\boldsymbol{x} \leftarrow \boldsymbol{D}}[\mathbf{Test}_{\boldsymbol{i}}(\boldsymbol{x}) = \mathbf{1}] \approx \mathbf{Pr}_{\boldsymbol{x} \leftarrow \boldsymbol{U_n}}[\mathbf{Test}_{\boldsymbol{i}}(\boldsymbol{x}) = \mathbf{1}]$ for all $\boldsymbol{i} = \mathbf{1, 2, \ldots}$

- This is not sufficient in an adversarial setting!
- Who knows what statistical test an attacker will use?

Pseudorandomness

Cryptographic definition

$\boldsymbol{D}$ is pseudorandom if it passes **all** efficient statistical tests

Pseudorandomness (concrete)

Definition

Let D be a distribution on p-bit strings. D is **(t, ϵ)-pseudorandom** if for all A running in time at most t it holds that:

$$|\mathbf{Pr}_{x \leftarrow D}[A(x) = 1] - \mathbf{Pr}_{x \leftarrow U_p}[A(x) = 1]| \leq \epsilon$$

Pseudorandomness (asymptotic)

- Security parameter $\boldsymbol{n}$, polynomial $\boldsymbol{p}$
- Let $\boldsymbol{D_n}$ be a distribution over $\boldsymbol{p(n)}$-bit strings
- **Pseudorandomness is a property of a sequence of distributions:**

$$\{\boldsymbol{D_n}\} = \{\boldsymbol{D_1}, \boldsymbol{D_2}, \ldots\}$$

Pseudorandomness (asymptotic)

Definition

$\{D_n\}$ is **pseudorandom** if for all probabilistic, polynomial-time distinguishers A, there is a negligible function ϵ such that

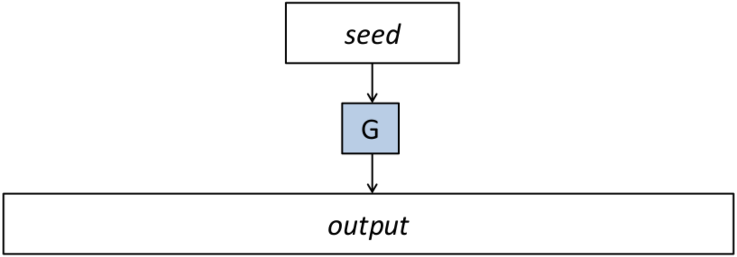
$$|\mathbf{Pr}_{x \leftarrow D_n}[A(x) = 1] - \mathbf{Pr}_{x \leftarrow U_{p(n)}}[A(x) = 1]| \leq \epsilon(n)$$

Pseudorandom Generators (PRG)

- A PRG is an efficient, deterministic algorithm that expands a **short, uniform seed** into a **longer, pseudorandom output**
- Useful whenever you have a *small* number of true random bits, and want lots of *random-looking* bits

PRGs

$\boldsymbol{G}$ is a deterministic, poly-time algorithm that is **expanding** i.e. $|\boldsymbol{G}(\boldsymbol{x})| = \boldsymbol{p}(|\boldsymbol{x}|) > |\boldsymbol{x}|$



PRGs

- G defines a sequence of distributions $\{D_n\}$
- D_n: the distribution on $p(n)$-bit strings defined by choosing $x \leftarrow U_n$ and outputting $G(x)$.
- The distribution on the output of G is far from uniform.
- Assume $U_n = \{0, 1\}^{2n}$ and consider G that takes inputs from $\{0, 1\}^n$.
- What is, at most, the size of the range of G?
- In the range of G there is only a small fraction of the strings samplable from U_n: $2^n/2^{2n} = 2^{-n}$
- Hence, most elements of U_n occur with probability $\mathbf{0}$ in the output fo G.
  - i.e. **Far from uniform**

PRGs

- G is a PRG $\iff$ $\{D_n\}$ is pseudorandom
- i.e. for all efficient distinguishers A, there is a negligible function ϵ such that

$$\left|\mathbf{Pr}_{x \leftarrow U_n}[A(G(x)) = 1] - \mathbf{Pr}_{y \leftarrow U_{p(n)}}[A(y) = 1]\right| \leq \epsilon(n)$$

- i.e. no efficient A can distinguish whether it is given $G(x)$ (for uniform x) or a uniform string y

Is the Following PRG Secure?

PRG
$\boldsymbol{G(x) = 0 \ldots 0}$

Is the Following PRG Secure?

PRG

$$\boldsymbol{G(x) = 0 \ldots 0}$$

Distinguisher

$\boldsymbol{A} =$ [all bits equal to $\mathbf{0}$]

Is the Following PRG Secure?

PRG

$G(x) = 0 \ldots 0$

Distinguisher

$A =$ [all bits equal to $\mathbf{0}$]

Analysis

$$\mathbf{Pr}_{x \leftarrow U_n}[A(G(x)) = 1] = 1$$

$$\mathbf{Pr}_{y \leftarrow U_{p(n)}}[A(y) = 1] = \frac{1}{2^{p(n)}}$$

$$1 - \frac{1}{2^{p(n)}} \approx 1 \not\lesssim \mathbf{negl}$$

Is the Following PRG Secure?

PRG

$\boldsymbol{G(x) = x \mid \texttt{OR}(\textbf{bits of } x)}$

Is the Following PRG Secure?

PRG

$\boldsymbol{G(x) = x \mid \texttt{OR}(\textbf{bits of } x)}$

Distinguisher

$\boldsymbol{A} = [\text{ least-significant bit} \neq \mathbf{0}]$

Is the Following PRG Secure?

PRG

$\boldsymbol{G(x) = x \mid \texttt{OR}(\textbf{bits of } x)}$

Distinguisher

$\boldsymbol{A} = [$ least-significant bit $\neq \mathbf{0}]$

Analysis

$$\mathbf{Pr}_{x \leftarrow U_n}[A(G(x)) = 1] = 1 - \frac{1}{2^n} \approx 1$$

$$\mathbf{Pr}_{y \leftarrow U_{p(n)}}[A(y) = 1] = \frac{1}{2}$$

$$1 - \frac{1}{2} \approx \frac{1}{2} \not\lesssim \mathbf{negl}$$

Do PRGs Exist?

- We don't know...
- Most of cryptography requires the unproven assumption that $\mathcal{P} \neq \mathcal{NP}$
- We will assume certain algorithms are PRGs
  - This is what is done in practice
- Can construct PRGs from weaker assumptions

So far

- We saw that there are some inherent limitations if we want **perfect secrecy**
- In particular, **key must be as long as the message**
- We defined **computational secrecy**, a relaxed notion of security
- We defined **PRG**
- Can we use **computational secrecy** + **PRG** to overcome prior limitations?

End

References: from Page 60 until the last paragraph of Page 64