ARIEL UNIVERSITY

אוניברסיטת אריאל בשומרון

Foundations of Modern Cryptography

Edward A. Hirsch

https://edwardahirsch.github.io/edwardahirsch

Neapolis University Pafos

Lecture 4: October 24, 2024

Lecture Plan

- Eventually, some proofs.
  - Goldreich–Levin Theorem proof.
  - Indistinguishability vs security proof.
- PRGs (pseudorandom generators).
  - . . . and security of our "more efficient" PKCS.

Lecture Plan

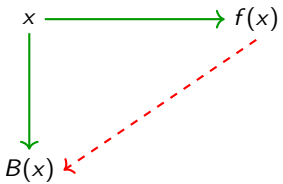
- Eventually, some proofs.
  - Goldreich–Levin Theorem proof.
  - Indistinguishability vs security proof.
- PRGs (pseudorandom generators).
  - . . . and security of our "more efficient" PKCS.

If the screen seems frozen and I do not respond,
please call me in Telegram.

Hardcore Predicate

Definition

$B\colon \{0,1\}^* \to \{0,1\}$ is a hardcore predicate for function $f\colon \{0,1\}^* \to \{0,1\}^*$ if
no adversary can guess $B(x)$ from $f(x)$ with probability non-negligibly better than 50%



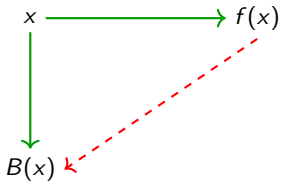
Oded Goldreich
Wikipedia

Leonid Levin
Taken by A.Smal

Hardcore Predicate

Definition

$B\colon \{0,1\}^* \to \{0,1\}$ is a hardcore predicate for function $f\colon \{0,1\}^* \to \{0,1\}^*$ if
no adversary can guess $B(x)$ from $f(x)$ with probability non-negligibly better than 50%



Oded Goldreich
Wikipedia

Leonid Levin
Taken by A.Smal

Theorem (Oded Goldreich, Leonid Levin)

If f is a strongly one-way function, then $\tilde{f}(x,r) = (f(x),r)$ is a strongly one-way function and $B(x,r) = \langle x,r\rangle \bmod 2 = x_1r_1 \oplus x_2r_2 \oplus \ldots \oplus x_nr_n$ is its hardcore predicate. *// Time to prove it!*

Goldreich–Levin Theorem: Proof

Computing an input bit x_i from hardcore bit

Adversary computes $B(x, r) = \langle x, r \rangle = \bigoplus_i x_i r_i$ from $f(x), r$

- Consider an adversary $\tilde{B}(f(x), r))$.
- Try inverting f (compute $x = x_1 \dots x_n$):

Goldreich–Levin Theorem: Proof

Computing an input bit x_i from hardcore bit

Adversary computes $B(x,r) = \langle x, r\rangle = \bigoplus_i x_i r_i$ from $f(x), r$

- Consider an adversary $\tilde{B}(f(x), r))$.
- Try inverting f (compute $x = x_1 \dots x_n$):
  - $e_i := (0, 0, \dots, 0, \underset{i}{1}, 0, \dots, 0)$
  - $r \oplus e_i = (r_1, r_2, \dots, r_i \oplus 1, \dots, r_n)$
  - $x_i = \langle x, r\rangle \oplus \langle x, r \oplus e_i\rangle = B(x,r) \oplus B(x, r \oplus e_i)$

Goldreich–Levin Theorem: Proof

Computing an input bit x_i from hardcore bit

Adversary computes $B(x,r) = \langle x, r\rangle = \bigoplus_i x_i r_i$ from $f(x), r$

- Consider an adversary $\tilde{B}(f(x), r))$.
- Try inverting f (compute $x = x_1 \dots x_n$):
  - $e_i := (0, 0, \dots, 0, \underset{i}{1}, 0, \dots, 0)$
  - $r \oplus e_i = (r_1, r_2, \dots, \bar{r}_i \quad , \dots, r_n)$
  - $x_i = \langle x, r\rangle \oplus \langle x, r \oplus e_i\rangle = B(x,r) \oplus B(x, r \oplus e_i)$

Goldreich–Levin Theorem: Proof

Computing an input bit x_i from hardcore bit

Adversary computes $B(x,r) = \langle x, r\rangle = \bigoplus_i x_i r_i$ from $f(x), r$

- Consider an adversary $\tilde{B}(f(x), r))$.
- Try inverting f (compute $x = x_1 \dots x_n$):
  - $e_i := (0, 0, \dots, 0, \underset{i}{1}, 0, \dots, 0)$
  - $r \oplus e_i = (r_1, r_2, \dots, \bar{r}_i \quad , \dots, r_n)$
  - $x_i = \langle x, r\rangle \oplus \langle x, r \oplus e_i\rangle = B(x,r) \oplus B(x, r \oplus e_i) \overset{?}{=} \underbrace{\tilde{B}(f(x), r)}_{\beta_r} \oplus \underbrace{\tilde{B}(f(x), r \oplus e_i)}_{\beta_{r\oplus e_i}}$

Goldreich–Levin Theorem: Proof

Computing an input bit x_i from hardcore bit

Adversary computes $B(x,r) = \langle x, r\rangle = \bigoplus_i x_i r_i$ from $f(x), r$

- Consider an adversary $\tilde{B}(f(x), r))$.
- Try inverting f (compute $x = x_1 \dots x_n$):
  - $e_i := (0, 0, \dots, 0, \underset{i}{1}, 0, \dots, 0)$
  - $r \oplus e_i = (r_1, r_2, \dots, \bar{r}_i \quad , \dots, r_n)$
  - $x_i = \langle x, r\rangle \oplus \langle x, r \oplus e_i\rangle = B(x,r) \oplus B(x, r \oplus e_i) \overset{?}{=} \underbrace{\tilde{B}(f(x), r)}_{\beta_r} \oplus \underbrace{\tilde{B}(f(x), r \oplus e_i)}_{\beta_{r\oplus e_i}}$
- $\tilde{B}$ may err, it only guesses with probability $\frac{1}{2} + \delta$

Goldreich–Levin Theorem: Proof

Computing an input bit x_i from hardcore bit

Adversary computes $B(x,r) = \langle x, r\rangle = \bigoplus_i x_i r_i$ from $f(x), r$

- Consider an adversary $\tilde{B}(f(x), r))$.
- Try inverting f (compute $x = x_1 \dots x_n$):
  - $e_i := (0, 0, \dots, 0, \underset{i}{1}, 0, \dots, 0)$
  - $r \oplus e_i = (r_1, r_2, \dots, \bar{r}_i \quad , \dots, r_n)$
  - $x_i = \langle x, r\rangle \oplus \langle x, r \oplus e_i\rangle = B(x,r) \oplus B(x, r \oplus e_i) \overset{?}{=} \underbrace{\tilde{B}(f(x), r)}_{\beta_r} \oplus \underbrace{\tilde{B}(f(x), r \oplus e_i)}_{\beta_{r\oplus e_i}}$
- $\tilde{B}$ may err, it only guesses with probability $\frac{1}{2} + \delta$
- The values β_s are not independent

Goldreich–Levin Theorem: Proof

Computing an input bit x_i from hardcore bit

Adversary computes $B(x,r) = \langle x, r\rangle = \bigoplus_i x_i r_i$ from $f(x), r$

- Consider an adversary $\tilde{B}(f(x), r))$.
- Try inverting f (compute $x = x_1 \dots x_n$):
  - $e_i := (0, 0, \dots, 0, \underset{i}{1}, 0, \dots, 0)$
  - $r \oplus e_i = (r_1, r_2, \dots, \bar{r}_i \quad , \dots, r_n)$
  - $x_i = \langle x, r\rangle \oplus \langle x, r \oplus e_i\rangle = B(x,r) \oplus B(x, r \oplus e_i) \stackrel{?}{=} \underbrace{\tilde{B}(f(x), r)}_{\beta_r} \oplus \underbrace{\tilde{B}(f(x), r \oplus e_i)}_{\beta_{r\oplus e_i}}$
- $\tilde{B}$ may err, it only guesses with probability $\frac{1}{2} + \delta$
- The values β_s are not independent
- Choose $\beta_{r\oplus e_i}$ (try 0 and 1), get β_r from $\tilde{B}$
  - cannot check the correctness without all the x_i's
  - too many cases for $i = 1, 2, \dots, n$

Goldreich–Levin Theorem: Proof

Computing an input bit x_i from hardcore bit

Adversary computes $B(x,r) = \langle x, r\rangle = \bigoplus_i x_i r_i$ from $f(x), r$

- Consider an adversary $\tilde{B}(f(x), r))$.
- Try inverting f (compute $x = x_1 \dots x_n$):
  - $e_i := (0, 0, \dots, 0, \underset{i}{1}, 0, \dots, 0)$
  - $r \oplus e_i = (r_1, r_2, \dots, \bar{r}_i \quad , \dots, r_n)$
  - $x_i = \langle x, r\rangle \oplus \langle x, r \oplus e_i\rangle = B(x,r) \oplus B(x, r \oplus e_i) \overset{?}{=} \underbrace{\tilde{B}(f(x), r)}_{\beta_r} \oplus \underbrace{\tilde{B}(f(x), r \oplus e_i)}_{\beta_{r\oplus e_i}}$
- $\tilde{B}$ may err, it only guesses with probability $\frac{1}{2} + \delta$
- The values β_s are not independent
- Choose $\beta_{r\oplus e_i}$ (try 0 and 1), get β_r from $\tilde{B}$
  - cannot check the correctness without all the x_i's
  - too many cases for $i = 1, 2, \dots, n$
- **We will choose β_s for logarithmically many random s's and clone**

Goldreich–Levin Theorem: Proof

Small sample space

- $\tilde{B}$ guesses B with probability $\frac{1}{2} + \delta$

- Sample $\ell = O(\log n)$ vectors $r^j \leftarrow U_n$
- Try all possible bits β_j for $B(x, r^j)$ (and assume we selected the correct ones)

Goldreich–Levin Theorem: Proof

Small sample space

- $\tilde{B}$ guesses B with probability $\frac{1}{2} + \delta$

- Sample $\ell = O(\log n)$ vectors $r^j \leftarrow U_n$
- Try all possible bits β_j for $B(x, r^j)$ (and assume we selected the correct ones)
- Compute many vectors: for each nonempty $\mathsf{J} \subseteq \{1, \ldots, \ell\}$

$$r^{\mathsf{J}} = \bigoplus_{j \in \mathsf{J}} r^j \qquad \beta^{\mathsf{J}} = \bigoplus_{j \in \mathsf{J}} \beta^j$$

These β's are also correct as $\langle x, s \rangle \oplus \langle x, t \rangle = \bigoplus_i x_i(s_i \oplus t_i) = \langle x, s \oplus t \rangle$!

Goldreich–Levin Theorem: Proof

Small sample space

- $\tilde{B}$ guesses B with probability $\frac{1}{2} + \delta$

- Sample $\ell = O(\log n)$ vectors $r^j \leftarrow U_n$
- Try all possible bits β_j for $B(x, r^j)$ (and assume we selected the correct ones)
- Compute many vectors: for each nonempty $\mathsf{J} \subseteq \{1, \ldots, \ell\}$

$$r^{\mathsf{J}} = \bigoplus_{j \in \mathsf{J}} r^j \qquad \beta^{\mathsf{J}} = \bigoplus_{j \in \mathsf{J}} \beta^j$$

  These β's are also correct as $\langle x, s \rangle \oplus \langle x, t \rangle = \bigoplus_i x_i(s_i \oplus t_i) = \langle x, s \oplus t \rangle$!

- Final version: Instead of

$$\tilde{B}(f(x), r) \oplus \tilde{B}(f(x), r \oplus e_i)$$

  compute

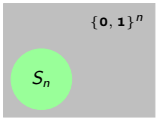
$$\begin{aligned} x_i^{\mathsf{J}} &= \beta^{\mathsf{J}} \oplus \tilde{B}(f(x), r^{\mathsf{J}} \oplus e_i) \\ \tilde{x}_i &= \operatorname*{maj}_{\mathsf{J}} x_i^{\mathsf{J}} \end{aligned}$$

Goldreich–Levin Theorem: Proof

Bird's eye view

- Lemma 1: There are many inputs where $\tilde{B}$ is correct whp:

$$\left| \left\{ x \mid \mathbf{Pr}\{ \tilde{B}(f(x), r) = B(x, r) \} \geqslant \frac{1}{2} + \frac{\delta}{2} \right\} \right| \geqslant \delta \cdot 2^n$$

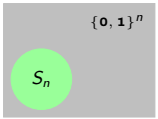


Goldreich–Levin Theorem: Proof

Bird's eye view

- Lemma 1: There are many inputs where $\tilde{B}$ is correct whp:

$$\left| \left\{ x \mid \mathbf{Pr}\{ \tilde{B}(f(x), r) = B(x, r) \} \geqslant \frac{1}{2} + \frac{\delta}{2} \right\} \right| \geqslant \delta \cdot 2^n$$



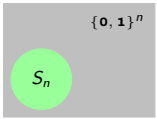
- Lemma 2: $r^{\mathsf{J}} \in U_n$ and $r^{\mathsf{J}}, r^{\mathsf{K}}$ are independent

Goldreich–Levin Theorem: Proof

Bird's eye view

- Lemma 1: There are many inputs where $\tilde{B}$ is correct whp:

$$\left|\left\{x \mid \mathbf{Pr}\{\tilde{B}(f(x), r) = B(x, r)\} \geqslant \frac{1}{2} + \frac{\delta}{2}\right\}\right| \geqslant \delta \cdot 2^n$$



- Lemma 2: $r^{\mathsf{J}} \in U_n$ and $r^{\mathsf{J}}, r^{\mathsf{K}}$ are independent
- Boolean (0/1) variable $\zeta_i^{\mathsf{J}} := \{x_i = x_i^{\mathsf{J}}\}$, the i-th bit is computed correctly for J
- Lemma 3: Most answers are computed correctly whp for $x \in S_n$

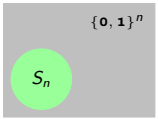
$$\mathbf{Pr}\left\{\sum_{\mathsf{J}} \zeta_i^{\mathsf{J}} \leqslant \frac{2^\ell - 1}{2}\right\} < \frac{1}{2n}$$

Goldreich–Levin Theorem: Proof

Bird's eye view

- Lemma 1: There are many inputs where $\tilde{B}$ is correct whp:

$$\left|\left\{x \mid \mathbf{Pr}\{\tilde{B}(f(x), r) = B(x, r)\} \geqslant \frac{1}{2} + \frac{\delta}{2}\right\}\right| \geqslant \delta \cdot 2^n$$



- Lemma 2: $r^{\mathsf{J}} \in U_n$ and $r^{\mathsf{J}}, r^{\mathsf{K}}$ are independent
- Boolean (0/1) variable $\zeta_i^{\mathsf{J}} := \{x_i = x_i^{\mathsf{J}}\}$, the i-th bit is computed correctly for J
- Lemma 3: Most answers are computed correctly whp for $x \in S_n$

$$\mathbf{Pr}\left\{\sum_{\mathsf{J}} \zeta_i^{\mathsf{J}} \leqslant \frac{2^\ell - 1}{2}\right\} < \frac{1}{2n}$$

- Summary:
  - We try all possible β^j for the basic vectors r^j (a polynomial number)
  - For correct β's most answers x_i^{J} are correct whp
  - Therefore we compute correct x_i by majority with prob. $\geqslant \frac{1}{2}$ for $x \in S_n$ and wsp in total
  - Given all x_i's, check the answer using f

Goldreich–Levin Theorem: Proof

Lemma 1

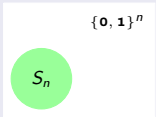
Lemma (1)

$\tilde{B}$ computes B with prob. $\geqslant \frac{1}{2} + \delta$, $\quad S_n = \{x \in \{0,1\}^n \mid \mathbf{Pr}_{\tilde{B},r}\{\underbrace{\tilde{B}(f(x), r) = B(x, r)}_{S(x)}\} \geqslant \frac{1}{2} + \frac{\delta}{2}\}$.

Then $|S_n| \geqslant \delta \cdot 2^n$.

Proof (counting argument).

$$|\overline{S_n}| = 2^n \cdot \mathbf{Pr}_x\left\{S(x) < \tfrac{1}{2} + \tfrac{\delta}{2}\right\} = 2^n \cdot \mathbf{Pr}_x\{1 - S(x) > \underbrace{\tfrac{1}{2} - \tfrac{\delta}{2}}_{\alpha_*}\},$$



Lemma (1)

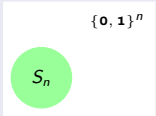
$\tilde{B}$ computes B with prob. $\geqslant \frac{1}{2} + \delta$, $\quad S_n = \{x \in \{0,1\}^n \mid \mathbf{Pr}_{\tilde{B},r}\{\underbrace{\tilde{B}(f(x), r) = B(x, r)}_{S(x)}\} \geqslant \frac{1}{2} + \frac{\delta}{2}\}$.

Then $|S_n| \geqslant \delta \cdot 2^n$.

Proof (counting argument).

$|\overline{S_n}| = 2^n \cdot \mathbf{Pr}_x\left\{S(x) < \frac{1}{2} + \frac{\delta}{2}\right\} = 2^n \cdot \mathbf{Pr}_x\{1 - S(x) > \underbrace{\frac{1}{2} - \frac{\delta}{2}}_{\alpha_*}\}$,

$\mathbf{E}\,(1 - S(x)) = 1 - \left(\frac{1}{2} + \delta\right) = \frac{1}{2} - \delta$.



Goldreich–Levin Theorem: Proof

Lemma (1)

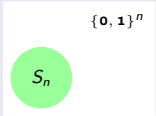
$\tilde{B}$ computes B with prob. $\geqslant \frac{1}{2} + \delta$, $\quad S_n = \{x \in \{0,1\}^n \mid \mathbf{Pr}_{\tilde{B},r}\{\underbrace{\tilde{B}(f(x), r) = B(x, r)}_{S(x)}\} \geqslant \frac{1}{2} + \frac{\delta}{2}\}$.

Then $|S_n| \geqslant \delta \cdot 2^n$.

Proof (counting argument).

$|\overline{S_n}| = 2^n \cdot \mathbf{Pr}_x\left\{S(x) < \frac{1}{2} + \frac{\delta}{2}\right\} = 2^n \cdot \mathbf{Pr}_x\{1 - S(x) > \underbrace{\frac{1}{2} - \frac{\delta}{2}}_{\alpha_*}\}$,

$\mathbf{E}\,(1 - S(x)) = 1 - \left(\frac{1}{2} + \delta\right) = \frac{1}{2} - \delta$.



Markov's inequality: $\mathbf{Pr}\{\alpha > \alpha_*\} \leqslant \frac{\mathbf{E}\,\alpha}{\alpha_*}$, where $\alpha \geqslant 0$.

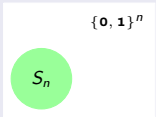
Lemma (1)

$\tilde{B}$ computes B with prob. $\geqslant \frac{1}{2} + \delta$, $\quad S_n = \{x \in \{0,1\}^n \mid \mathbf{Pr}_{\tilde{B},r}\{\underbrace{\tilde{B}(f(x), r) = B(x, r)}_{S(x)}\} \geqslant \frac{1}{2} + \frac{\delta}{2}\}$.

Then $|S_n| \geqslant \delta \cdot 2^n$.

Proof (counting argument).

$|\overline{S_n}| = 2^n \cdot \mathbf{Pr}_x\left\{S(x) < \frac{1}{2} + \frac{\delta}{2}\right\} = 2^n \cdot \mathbf{Pr}_x\{1 - S(x) > \underbrace{\frac{1}{2} - \frac{\delta}{2}}_{\alpha_*}\}$,



$\mathbf{E}\,(1 - S(x)) = 1 - \left(\frac{1}{2} + \delta\right) = \frac{1}{2} - \delta$.

Markov's inequality: $\mathbf{Pr}\{\alpha > \alpha_*\} \leqslant \frac{\mathbf{E}\,\alpha}{\alpha_*}$, where $\alpha \geqslant 0$.

Thus

$$\mathbf{Pr}_x\{1 - S(x) > \tfrac{1}{2} - \tfrac{\delta}{2}\} \leqslant \frac{\frac{1}{2} - \delta}{\frac{1}{2} - \frac{\delta}{2}} = \frac{1 - 2\delta}{1 - \delta} = 1 - \frac{\delta}{1 - \delta} \leqslant 1 - \delta.$$

□

Lemma (2)

$r^{\mathsf{J}} \in U_n$ (trivial) and also $r^{\mathsf{J}}, r^{\mathsf{K}}$ are independent

Proof.

If $K \subseteq \mathsf{J}$, then

$$\mathbf{Pr}\{r^{\mathsf{J}} = t,\ r^{K} = t'\} = \mathbf{Pr}\{r^{\mathsf{J}\setminus K} = t \oplus t',\ r^{K} = t'\} \overset{(\mathsf{J}\setminus K)\cap K=\emptyset}{=}$$

$$\mathbf{Pr}\{r^{\mathsf{J}\setminus K} = t \oplus t'\} \cdot \mathbf{Pr}\{r^{K} = t'\} \overset{\text{uniformly distributed!}}{=} \mathbf{Pr}\{r^{\mathsf{J}} = t\} \cdot \mathbf{Pr}\{r^{K} = t'\}.$$

Lemma (2)

$r^{\mathsf{J}} \in U_n$ (trivial) and also $r^{\mathsf{J}}, r^{\mathsf{K}}$ are independent

Proof.

If $K \subseteq \mathsf{J}$, then

$$\mathbf{Pr}\{r^{\mathsf{J}} = t,\ r^K = t'\} = \mathbf{Pr}\{r^{\mathsf{J}\setminus K} = t \oplus t',\ r^K = t'\} \overset{(\mathsf{J}\setminus K)\cap K = \emptyset}{=}$$

$$\mathbf{Pr}\{r^{\mathsf{J}\setminus K} = t \oplus t'\} \cdot \mathbf{Pr}\{r^K = t'\} \overset{\text{uniformly distributed!}}{=} \mathbf{Pr}\{r^{\mathsf{J}} = t\} \cdot \mathbf{Pr}\{r^K = t'\}.$$

Otherwise, $\mathsf{J} \setminus K \neq \emptyset$ and $K \setminus \mathsf{J} \neq \emptyset$. Then

$$\mathbf{Pr}\{r^{\mathsf{J}} = t,\ r^K = t'\} = \sum_{t''} \mathbf{Pr}\{r^{\mathsf{J}} = t,\ r^K = t',\ r^{\mathsf{J}\cap K} = t''\} =$$

$$\sum_{t''} \mathbf{Pr}\{r^{\mathsf{J}\setminus K} = t \oplus t'',\ r^{K\setminus \mathsf{J}} = t' \oplus t'',\ r^{\mathsf{J}\cap K} = t''\} =$$

$$\mathbf{Pr}\{r^{\mathsf{J}\setminus K} = t \oplus t''\} \cdot \mathbf{Pr}\{r^{K\setminus \mathsf{J}} = t' \oplus t''\} \cdot \underbrace{\sum_{t''} \mathbf{Pr}\{r^{\mathsf{J}\cap K} = t''\}}_{1} \overset{\text{uniform}}{=} \mathbf{Pr}\{r^{\mathsf{J}} = t\} \cdot \mathbf{Pr}\{r^K = t'\}. \quad \square$$

Lemma (3)

Let $\delta = \frac{1}{n^k}$, $\quad \ell = (2k+2)\lceil \log_2 n \rceil$, $\quad m = 2^\ell - 1$.
Boolean (0/1) variable $\zeta_i^J := \{x_i = x_i^J\}$ *// the i-th bit is computed correctly for J*
Then for n big enough, for $x \in S_n$,

$$\mathbf{Pr}\left\{\sum_J \zeta_i^J \leqslant \frac{m}{2}\right\} < \frac{1}{2n}$$

Lemma (3)

Let $\delta = \frac{1}{n^k}$, $\quad \ell = (2k+2)\lceil \log_2 n \rceil$, $\quad m = 2^\ell - 1$.
Boolean (0/1) variable $\zeta_i^J := \{x_i = x_i^J\}$ *// the i-th bit is computed correctly for J*
Then for n big enough, for $x \in S_n$,

$$\mathbf{Pr}\left\{\sum_J \zeta_i^J \leqslant \frac{m}{2}\right\} < \frac{1}{2n}$$

Proof.

- For $x \in S_n$, success prob. $\geqslant \frac{1}{2} + \frac{\delta}{2}$ *// and S_n is "large", Lemma 1*

Lemma (3)

Let $\delta = \frac{1}{n^k}$, $\quad \ell = (2k+2)\lceil \log_2 n \rceil$, $\quad m = 2^\ell - 1$.
Boolean (0/1) variable $\zeta_i^{\mathsf{J}} := \{x_i = x_i^{\mathsf{J}}\}$ *// the i-th bit is computed correctly for J*
Then for n big enough, for $x \in S_n$,

$$\mathbf{Pr}\left\{\sum_{\mathsf{J}} \zeta_i^{\mathsf{J}} \leqslant \frac{m}{2}\right\} < \frac{1}{2n}$$

Proof.

- For $x \in S_n$, success prob. $\geqslant \frac{1}{2} + \frac{\delta}{2}$ *// and S_n is "large", Lemma 1*
- $\mathbf{E} \sum \zeta_i^{\mathsf{J}} = m\left(\frac{1}{2} + \frac{\delta}{2}\right)$

Lemma (3)

Let $\delta = \frac{1}{n^k}$, $\quad \ell = (2k+2)\lceil \log_2 n \rceil$, $\quad m = 2^\ell - 1$.
Boolean (0/1) variable $\zeta_i^{\mathsf{J}} := \{x_i = x_i^{\mathsf{J}}\}$ *// the i-th bit is computed correctly for J*
Then for n big enough, for $x \in S_n$,

$$\mathbf{Pr}\left\{\sum_{\mathsf{J}} \zeta_i^{\mathsf{J}} \leqslant \frac{m}{2}\right\} < \frac{1}{2n}$$

Proof.

- For $x \in S_n$, success prob. $\geqslant \frac{1}{2} + \frac{\delta}{2}$ *// and S_n is "large", Lemma 1*
- $\mathbf{E} \sum \zeta_i^{\mathsf{J}} = m\left(\frac{1}{2} + \frac{\delta}{2}\right)$
- Pairwise independence (Lemma 2) yields $\mathrm{Var}(\sum \zeta_i^{\mathsf{J}}) = m \cdot \mathrm{Var}(\zeta_i^{\mathsf{J}})$

Lemma (3)

Let $\delta = \frac{1}{n^k}$, $\quad \ell = (2k+2)\lceil \log_2 n \rceil$, $\quad m = 2^\ell - 1$.
Boolean (0/1) variable $\zeta_i^{\mathsf{J}} := \{x_i = x_i^{\mathsf{J}}\}$ *// the i-th bit is computed correctly for J*
Then for n big enough, for $x \in S_n$,

$$\mathbf{Pr}\left\{\sum_{\mathsf{J}} \zeta_i^{\mathsf{J}} \leqslant \frac{m}{2}\right\} < \frac{1}{2n}$$

Proof.

- For $x \in S_n$, success prob. $\geqslant \frac{1}{2} + \frac{\delta}{2}$ *// and S_n is "large", Lemma 1*
- $\mathbf{E} \sum \zeta_i^{\mathsf{J}} = m\left(\frac{1}{2} + \frac{\delta}{2}\right)$
- Pairwise independence (Lemma 2) yields $\mathrm{Var}(\sum \zeta_i^{\mathsf{J}}) = m \cdot \mathrm{Var}(\zeta_i^{\mathsf{J}})$
- Chebyshev's inequality $\mathbf{Pr}\{\alpha < \mathbf{E}\,\alpha - k\sqrt{\mathrm{Var}(\alpha)}\} < \frac{1}{k^2}$ $\quad // \Delta = k\sqrt{\mathrm{Var}(\alpha)}$

Lemma (3)

Let $\delta = \frac{1}{n^k}$, $\quad \ell = (2k+2)\lceil \log_2 n \rceil$, $\quad m = 2^\ell - 1$.
Boolean (0/1) variable $\zeta_i^{\mathsf{J}} := \{x_i = x_i^{\mathsf{J}}\}$ *// the i-th bit is computed correctly for J*
Then for n big enough, for $x \in S_n$,

$$\mathbf{Pr}\left\{\sum_{\mathsf{J}} \zeta_i^{\mathsf{J}} \leqslant \frac{m}{2}\right\} < \frac{1}{2n}$$

Proof.

- For $x \in S_n$, success prob. $\geqslant \frac{1}{2} + \frac{\delta}{2}$ *// and S_n is "large", Lemma 1*
- $\mathbf{E} \sum \zeta_i^{\mathsf{J}} = m\left(\frac{1}{2} + \frac{\delta}{2}\right)$
- Pairwise independence (Lemma 2) yields $\mathrm{Var}(\sum \zeta_i^{\mathsf{J}}) = m \cdot \mathrm{Var}(\zeta_i^{\mathsf{J}})$
- Chebyshev's inequality $\mathbf{Pr}\{\alpha < \mathbf{E}\,\alpha - \Delta\} < \frac{\mathrm{Var}(\alpha)}{\Delta^2}$ *// $k = \Delta/\sqrt{\mathrm{Var}(\alpha)}$*

Lemma (3)

Let $\delta = \frac{1}{n^k}$, $\quad \ell = (2k+2)\lceil \log_2 n \rceil$, $\quad m = 2^\ell - 1$.
Boolean (0/1) variable $\zeta_i^{\mathsf{J}} := \{x_i = x_i^{\mathsf{J}}\}$ // *the i-th bit is computed correctly for J*
Then for n big enough, for $x \in S_n$,

$$\mathbf{Pr}\left\{\sum_{\mathsf{J}} \zeta_i^{\mathsf{J}} \leqslant \frac{m}{2}\right\} < \frac{1}{2n}$$

Proof.

- For $x \in S_n$, success prob. $\geqslant \frac{1}{2} + \frac{\delta}{2}$ // *and S_n is "large", Lemma 1*
- $\mathbf{E}\sum \zeta_i^{\mathsf{J}} = m\left(\frac{1}{2} + \frac{\delta}{2}\right) \Longrightarrow \frac{m}{2} = \mathbf{E}\ldots - \frac{m\delta}{2}$, we need more
- Pairwise independence (Lemma 2) yields $\mathrm{Var}(\sum \zeta_i^{\mathsf{J}}) = m \cdot \mathrm{Var}(\zeta_i^{\mathsf{J}})$
- Chebyshev's inequality $\mathbf{Pr}\{\alpha < \mathbf{E}\,\alpha - \Delta\} < \frac{\mathrm{Var}(\alpha)}{\Delta^2}$ // $k = \Delta/\sqrt{\mathrm{Var}(\alpha)}$
- $\mathbf{Pr}\{\sum_{\mathsf{J}} \zeta_i^{\mathsf{J}} < \mathbf{E}\ldots - \frac{m\delta}{2}\} < \frac{4 \cdot \mathrm{Var}(\sum \zeta_i^{\mathsf{J}})}{m^2\delta^2} < \frac{4}{m\delta^2} \leqslant \frac{4}{n^2}$ // $\mathrm{Var}(\zeta) = \boldsymbol{E}(\zeta - \boldsymbol{E}\zeta)^2 < 1$

Exercises

Exercise

Do we really need length preserving owf ($|x| = |x'| \Rightarrow |f(x)| = |f(x')|$)?

Exercise

We need to know the success probability of $\tilde{B}$ (where?). Get rid of it.

Exercise

What if f is not injective?

Pseudorandom generators (PRG) and our "more efficient" PKCS

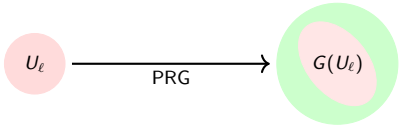
Definition

$G : \{0,1\}^{\ell} \to \{0,1\}^{k(\ell)}$ is an $k(\ell)$-PRG if for every adversary A,
the distributions $G(U_\ell)$ and $U_{k(\ell)}$ are computationally indistinguishable.

Pseudorandom generators (PRG) and our "more efficient" PKCS

Definition

$G : \{0,1\}^{\ell} \to \{0,1\}^{k(\ell)}$ is an $k(\ell)$-PRG if for every adversary A,
the distributions $G(U_\ell)$ and $U_{k(\ell)}$ are computationally indistinguishable.



Theorem

$E^{**}(b_1 \dots b_m, e, r) = (e^m(r), B(r) \oplus b_1, B(e(r)) \oplus b_2, \dots)$,

If one breaks E^{**} (distinguishes $00\dots0$ from a random message), then we can break a PRG.

Lemma

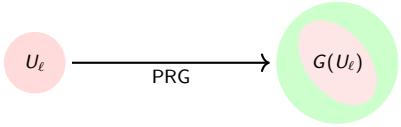
If f is a length-preserving owp and B is its hardcore predicate, then there is an $k(\ell)$-PRG, namely

$$G_{k(\ell)}(x) = \left(f^{k(\ell)-\ell}(x)\ ,\ B(x)\ ,\ B(g(x))\ ,\ \dots\ ,\ B(f^{k(\ell)-\ell-1}(x))\right)$$

Pseudorandom generators (PRG) and our "more efficient" PKCS

Definition

$G : \{0,1\}^{\ell} \to \{0,1\}^{k(\ell)}$ is an $k(\ell)$-PRG if for every adversary A,
the distributions $G(U_\ell)$ and $U_{k(\ell)}$ are computationally indistinguishable.



Theorem

$E^{**}(b_1 \dots b_m, e, r) = (e^m(r), B(r) \oplus b_1, B(e(r)) \oplus b_2, \dots)$,

If one breaks E^{**} (distinguishes $00\dots0$ from a random message), then we can break a PRG.

Lemma

If f is a length-preserving owp and B is its hardcore predicate, then there is an $k(\ell)$-PRG, namely

$$G_{k(\ell)}(x) = \left(f^{k(\ell)-\ell}(x) \circ B(x) \circ B(f(x)) \circ \dots \circ B(f^{k(\ell)-\ell-1}(x))\right)$$

If we distinguish U from $G_{k(\ell)}(U)$, then we distinguish $G_i(U)$ and $G_{i+1}(U)$ for some i (perhaps $i = 0$, for U).

U from $G_1(U)$

$G(x) = f(x) \circ B(x)$ is broken by A:

$$|\underbrace{\mathbf{Pr}\{A(f(x)) = 1\}}_{\gamma} - \underbrace{\mathbf{Pr}\{A(y) = 1\}}_{\nu}| \geq \frac{1}{\ell^k}.$$

U from $G_1(U)$

$G(x) = f(x) \circ B(x)$ is broken by A:

$$|\underbrace{\mathbf{Pr}\{A(f(x)) = 1\}}_{\gamma} - \underbrace{\mathbf{Pr}\{A(y) = 1\}}_{\nu}| \geq \frac{1}{\ell^k}.$$

The first n bits are distributed as U_n anyway (both $f(x)$ and $y_{1..n}$).
The last bit b is a random variable that is either $B(x)$ or independent of the first bits.

U from $G_1(U)$

$G(x) = f(x) \circ B(x)$ is broken by A:

$$|\underbrace{\mathbf{Pr}\{A(f(x)) = 1\}}_{\gamma} - \underbrace{\mathbf{Pr}\{A(y) = 1\}}_{\nu}| \geq \frac{1}{\ell^k}.$$

The first n bits are distributed as U_n anyway (both $f(x)$ and $y_{1..n}$).
The last bit b is a random variable that is either $B(x)$ or independent of the first bits.

$\alpha := \mathbf{Pr}\{A(f(x) \circ b) = 1 | b = B(x)\} = \mathbf{Pr}\{A(f(x) \circ B(x)) = 1\} = \gamma$,
$\beta := \mathbf{Pr}\{A(f(x) \circ b) = 1 | b = \overline{B(x)}\} = \mathbf{Pr}\{A(f(x) \circ \overline{B(x)}) = 1\}$.

$\nu = \mathbf{Pr}\{A(f(x) \circ b) = 1\} = \mathbf{Pr}\{b = B(x)\} \cdot \alpha + \mathbf{Pr}\{b = \overline{B(x)}\} \cdot \beta = \frac{\alpha+\beta}{2}$.

W.l.o.g. $\frac{1}{\ell^k} \leqslant \gamma - \nu = \alpha - \frac{\alpha+\beta}{2} = \frac{\alpha-\beta}{2}$.

U from $G_1(U)$

$G(x) = f(x) \circ B(x)$ is broken by A:

$$|\underbrace{\mathbf{Pr}\{A(f(x)) = 1\}}_{\gamma} - \underbrace{\mathbf{Pr}\{A(y) = 1\}}_{\nu}| \geq \frac{1}{\ell^k}.$$

The first n bits are distributed as U_n anyway (both $f(x)$ and $y_{1..n}$).
The last bit b is a random variable that is either $B(x)$ or independent of the first bits.

$\alpha := \mathbf{Pr}\{A(f(x) \circ b) = 1 | b = B(x)\} = \mathbf{Pr}\{A(f(x) \circ B(x)) = 1\} = \gamma$,
$\beta := \mathbf{Pr}\{A(f(x) \circ b) = 1 | b = \overline{B(x)}\} = \mathbf{Pr}\{A(f(x) \circ \overline{B(x)}) = 1\}$.

$\nu = \mathbf{Pr}\{A(f(x) \circ b) = 1\} = \mathbf{Pr}\{b = B(x)\} \cdot \alpha + \mathbf{Pr}\{b = \overline{B(x)}\} \cdot \beta = \frac{\alpha+\beta}{2}$.

W.l.o.g. $\frac{1}{\ell^k} \leqslant \gamma - \nu = \alpha - \frac{\alpha+\beta}{2} = \frac{\alpha-\beta}{2}$.

Our new adversary A' computes $B(x)$ from its input w $(= f(x))$:

- pick $b \leftarrow U_1$,
- if $A(w \circ b) = 1$, then say b, otherwise say $\overline{b}$.

U from $G_1(U)$

$G(x) = f(x) \circ B(x)$ is broken by A:

$$|\underbrace{\mathbf{Pr}\{A(f(x)) = 1\}}_{\gamma} - \underbrace{\mathbf{Pr}\{A(y) = 1\}}_{\nu}| \geq \frac{1}{\ell^k}.$$

The first n bits are distributed as U_n anyway (both $f(x)$ and $y_{1..n}$).
The last bit b is a random variable that is either $B(x)$ or independent of the first bits.

$\alpha := \mathbf{Pr}\{A(f(x) \circ b) = 1 | b = B(x)\} = \mathbf{Pr}\{A(f(x) \circ B(x)) = 1\} = \gamma$,
$\beta := \mathbf{Pr}\{A(f(x) \circ b) = 1 | b = \overline{B(x)}\} = \mathbf{Pr}\{A(f(x) \circ \overline{B(x)}) = 1\}$.

$\nu = \mathbf{Pr}\{A(f(x) \circ b) = 1\} = \mathbf{Pr}\{b = B(x)\} \cdot \alpha + \mathbf{Pr}\{b = \overline{B(x)}\} \cdot \beta = \frac{\alpha+\beta}{2}$.

W.l.o.g. $\frac{1}{\ell^k} \leqslant \gamma - \nu = \alpha - \frac{\alpha+\beta}{2} = \frac{\alpha-\beta}{2}$.

Our new adversary A' computes $B(x)$ from its input w $(= f(x))$:

- pick $b \leftarrow U_1$,
- if $A(w \circ b) = 1$, then say b, otherwise say $\overline{b}$.

$$\begin{aligned}\mathbf{Pr}\{A'(f(x)) = B(x)\} = \quad & \mathbf{Pr}\{b = B(x)\} \cdot \mathbf{Pr}\{A(f(x) \circ B(x)) = 1\} \quad + \\ & \mathbf{Pr}\{b = \overline{B(x)}\} \cdot \mathbf{Pr}\{A(f(x) \circ \overline{B(x)}) \neq 1\} \quad = \tfrac{1}{2}\alpha + \tfrac{1}{2}(1-\beta) = \tfrac{1}{2} + \tfrac{\alpha-\beta}{2} \geq \tfrac{1}{2} + \tfrac{1}{\ell^k}\end{aligned}$$

$G_i(U)$ from $G_{i+1}(U)$

A distinguishes

$G_{i+1}(x)$	$f^{i+1}(x)$	,	$B(x)$	,	$B(f(x))$	,	$\ldots$	,	$B(f^i(x))$
$G_i(z)$	$f^i(z)$	,	b_i	,	$B(z)$	,	$\ldots$	,	$B(f^{i-1}(z))$

$G_i(U)$ from $G_{i+1}(U)$

A distinguishes

$$\begin{array}{lccccccc} G_{i+1}(x) & f^{i+1}(x) & , & B(x) & , & B(f(x)) & , \quad \ldots \quad , & B(f^i(x)) \\ G_i(z) & f^i(z) & , & b_i & , & B(z) & , \quad \ldots \quad , & B(f^{i-1}(z)) \end{array}$$

Someone gave us y and b and asks to distinguish $G_1(U)$ from U :

$$\begin{array}{lll} y = f(x), & b = B(x), & \text{where } x \leftarrow U_n \\ y = z \quad , & b = b_i \quad , & \text{where } z \leftarrow U_n,\ b_i \leftarrow U_1 \end{array}$$

$G_i(U)$ from $G_{i+1}(U)$

A distinguishes

$$\begin{array}{lccccccc} G_{i+1}(x) & f^{i+1}(x) & , & B(x) & , & B(f(x)) & , \ \ldots \ , & B(f^i(x)) \\ G_i(z) & f^i(z) & , & b_i & , & B(z) & , \ \ldots \ , & B(f^{i-1}(z)) \end{array}$$

Someone gave us y and b and asks to distinguish $G_1(U)$ from U :

$$\begin{array}{lll} y = f(x), & b = B(x), & \text{where } x \leftarrow U_n \\ y = z \quad , & b = b_i \quad , & \text{where } z \leftarrow U_n,\ b_i \leftarrow U_1 \end{array}$$

Give it to A as

$$f^i(y) \quad , \quad b \quad , \quad B(y) \quad , \ \ldots \ , \quad B(f^{i-1}(y))$$

Then A will distinguish $G_{i+1}(U)$ from $G_i(U)$ for us.

$G_i(U)$ from $G_{i+1}(U)$

A distinguishes

$$\begin{array}{lccccccc} G_{i+1}(x) & f^{i+1}(x) & , & B(x) & , & B(f(x)) & , \ \ldots \ , & B(f^i(x)) \\ G_i(z) & f^i(z) & , & b_i & , & B(z) & , \ \ldots \ , & B(f^{i-1}(z)) \end{array}$$

Someone gave us y and b and asks to distinguish $G_1(U)$ from U :

$$\begin{array}{lll} y = f(x), & b = B(x), & \text{where } x \leftarrow U_n \\ y = z \quad , & b = b_i \quad , & \text{where } z \leftarrow U_n,\ b_i \leftarrow U_1 \end{array}$$

Give it to A as

$$\begin{array}{lccccccc} G_{i+1}(x) & f^{i+1}(x) & , & B(x) & , & B(f(x)) & , \ \ldots \ , & B(f^i(x)) \\ & f^i(y) & , & b & , & B(y) & , \ \ldots \ , & B(f^{i-1}(y)) \end{array}$$

Then A will distinguish $G_{i+1}(U)$ from $G_i(U)$ for us.

$G_i(U)$ from $G_{i+1}(U)$

A distinguishes

$$\begin{array}{lccccccc} G_{i+1}(x) & f^{i+1}(x) & , & B(x) & , & B(f(x)) & , \ \ldots \ , & B(f^i(x)) \\ G_i(z) & f^i(z) & , & b_i & , & B(z) & , \ \ldots \ , & B(f^{i-1}(z)) \end{array}$$

Someone gave us y and b and asks to distinguish $G_1(U)$ from U :

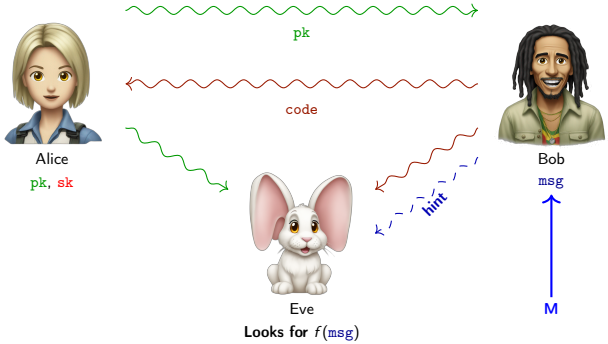
$$\begin{array}{lll} y = f(x), & b = B(x), & \text{where } x \leftarrow U_n \\ y = z \quad , & b = b_i \quad , & \text{where } z \leftarrow U_n,\ b_i \leftarrow U_1 \end{array}$$

Give it to A as

$$\begin{array}{lccccccc} G_{i+1}(x) & f^{i+1}(x) & , & B(x) & , & B(f(x)) & , \ \ldots \ , & B(f^i(x)) \\ & f^i(y) & , & b & , & B(y) & , \ \ldots \ , & B(f^{i-1}(y)) \\ G_i(z) & f^i(z) & , & b_i & , & B(z) & , \ \ldots \ , & B(f^{i-1}(z)) \end{array}$$

Then A will distinguish $G_{i+1}(U)$ from $G_i(U)$ for us.

Computational indistinguishability vs Semantic security



Definition

A PKCS is semantically secure if $\forall f\ \forall h\ \forall A\ \forall M$

$$\mathbf{Pr}\{A(E(e,m),e,h(m))=f(m)\} \leqslant \mathbf{Pr}\{\widetilde{A}(e,h(m))=f(m)\}+\varepsilon(n),$$

where the hint h and the "sense" f are polynomial-time computable,
M is a message generator, A is an adversary trying to guess f from code,
$\widetilde{A}$ is an "adversary" guessing f without code.

Computational indistinguishability vs Semantic security

Definition

Probability distributions P and Q are computationally indistinguishable if $\forall$ adversary A

$$|\mathbf{Pr}_{x \leftarrow P}\{A(x) = 1\} - \mathbf{Pr}_{x \leftarrow Q}\{A(x) = 1\}| < \varepsilon(n)$$

Definition

A PKCS is semantically secure if $\forall f \ \forall h \ \forall A \ \forall M$

$$\mathbf{Pr}\{A(E(e, m), e, h(m)) = f(m)\} \leqslant \mathbf{Pr}\{\widetilde{A}(e, h(m)) = f(m)\} + \varepsilon(n),$$

where the hint h and the "sense" f are polynomial-time computable,
M is a message generator, A is an adversary trying to guess f from `code`,
$\widetilde{A}$ is an "adversary" guessing f without `code`.

Computational indistinguishability vs Semantic security

Definition

Probability distributions P and Q are computationally indistinguishable if $\forall$ adversary A

$$|\mathbf{Pr}_{x \leftarrow P}\{A(x) = 1\} - \mathbf{Pr}_{x \leftarrow Q}\{A(x) = 1\}| < \varepsilon(n)$$

Definition

or generated by the adversary?

A PKCS is computationally indistinguishable if $\forall$ messages (m_0, m_1) of polynomial length $\forall$ adversary A

$$\left| \begin{array}{ll} \mathbf{Pr}\{A(E(m_0, e, r_e), e, 1^n, m_0, m_1) = 1\} & - \\ \mathbf{Pr}\{A(E(m_1, e, r_e), e, 1^n, m_0, m_1) = 1\} & \end{array} \right| < \varepsilon(n).$$

The probability is taken over r_g, r_e, and A's randomness.

Definition

A PKCS is semantically secure if $\forall f\ \forall h\ \forall A\ \forall M$

$$\mathbf{Pr}\{A(E(e, m), e, h(m)) = f(m)\} \leqslant \mathbf{Pr}\{\widetilde{A}(e, h(m)) = f(m)\} + \varepsilon(n),$$

where the hint h and the "sense" f are polynomial-time computable,
M is a message generator, A is an adversary trying to guess f from `code`,
$\widetilde{A}$ is an "adversary" guessing f without `code`.

Computational indistinguishability vs Semantic security

Definition

Probability distributions P and Q are computationally indistinguishable if $\forall$ adversary A

$$|\mathbf{Pr}_{x\leftarrow P}\{A(x)=1\} - \mathbf{Pr}_{x\leftarrow Q}\{A(x)=1\}| < \varepsilon(n)$$

Definition

or generated by the adversary?

A PKCS is computationally indistinguishable if $\forall$ messages (m_0, m_1) of polynomial length $\forall$ adversary A

$$\left|\begin{array}{l} \mathbf{Pr}\{A(E(m_0,e,r_e),e,1^n,m_0,m_1)=1\} \\ \mathbf{Pr}\{A(E(m_1,e,r_e),e,1^n,m_0,m_1)=1\} \end{array}\right| < \varepsilon(n).$$

The probability is taken over r_g, r_e, and A's randomness.

Definition

A PKCS is semantically secure if $\forall f\ \forall h\ \forall A\ \forall M$

$$\mathbf{Pr}\{A(E(e,m),e,h(m))=f(m)\} \leqslant \mathbf{Pr}\{\widetilde{A}(e,h(m))=f(m)\} + \varepsilon(n),$$

where the hint h and the "sense" f are polynomial-time computable,
M is a message generator, A is an adversary trying to guess f from `code`,
$\widetilde{A}$ is an "adversary" guessing f without `code`.

Semantic security $\Leftrightarrow$ computational indistinguishability *// Time to prove it!*

Semantic security to indistinguishability

Definition

A PKCS is semantically secure if $\forall f\ \forall h\ \forall A\ \forall M$

$$\mathbf{Pr}\{A(E(e,m),e,h(m)) = f(m)\} \leqslant \mathbf{Pr}\{\widetilde{A}(e,h(m)) = f(m)\} + \varepsilon(n),$$

- Adversary A_I distinguishes encrypted m_0, m_1 // *Take the pair with the best probability of success*

Semantic security to indistinguishability

Definition

A PKCS is semantically secure if $\forall f\ \forall h\ \forall A\ \forall M$

$$\mathbf{Pr}\{A(E(e,m),e,h(m)) = f(m)\} \leqslant \mathbf{Pr}\{\widetilde{A}(e,h(m)) = f(m)\} + \varepsilon(n),$$

- Adversary A_I distinguishes encrypted m_0, m_1 *// Take the pair with the best probability of success*
- Let $f(m_i) := i$

Semantic security to indistinguishability

Definition

A PKCS is semantically secure if $\forall f\ \forall h\ \forall A\ \forall M$

$$\mathbf{Pr\{}A(E(e,m),e,h(m))=f(m)\mathbf{\}} \leqslant \mathbf{Pr\{}\widetilde{A}(e,h(m))=f(m)\mathbf{\}}+\varepsilon(n),$$

- Adversary A_I distinguishes encrypted m_0, m_1 // *Take the pair with the best probability of success*
- Let $f(m_i) := i$
- New adversary A_S uses A_I for m_0, m_1 ***(hardwired into the circuit M_S)***, which has success prob. $> \frac{1}{2}+\delta$

Semantic security to indistinguishability

Definition

A PKCS is semantically secure if $\forall f\ \forall h\ \forall A\ \forall M$

$$\mathbf{Pr\{}A(E(e,m),e,h(m))=f(m)\mathbf{\}} \leqslant \mathbf{Pr\{}\widetilde{A}(e,h(m))=f(m)\mathbf{\}}+\varepsilon(n),$$

- Adversary A_I distinguishes encrypted m_0, m_1 // *Take the pair with the best probability of success*
- Let $f(m_i) := i$
- New adversary A_S uses A_I for m_0, m_1 ***(hardwired into the circuit M_S)***, which has success prob. $> \frac{1}{2}+\delta$
- Can $\tilde{A}$ do better as well? Nope, it's $\frac{1}{2}$.

Semantic security to indistinguishability

Definition

A PKCS is semantically secure if $\forall f\ \forall h\ \forall A\ \forall M$

$$\mathbf{Pr\{}A(E(e,m),e,h(m))=f(m)\mathbf{\}} \leqslant \mathbf{Pr\{}\widetilde{A}(e,h(m))=f(m)\mathbf{\}}+\varepsilon(n),$$

- Adversary A_I distinguishes encrypted m_0, m_1 // *Take the pair with the best probability of success*
- Let $f(m_i) := i$
- New adversary A_S uses A_I for m_0, m_1 ***(hardwired into the circuit*** M_S***)***, which has success prob. $> \frac{1}{2}+\delta$
- Can $\tilde{A}$ do better as well? Nope, it's $\frac{1}{2}$.

- Think about generated vs any m_0, m_1.

Semantic security to indistinguishability

Definition

A PKCS is semantically secure if $\forall f\ \forall h\ \forall A\ \forall M$

$$\mathbf{Pr}\{A(E(e,m),e,h(m)) = f(m)\} \leqslant \mathbf{Pr}\{\widetilde{A}(e,h(m)) = f(m)\} + \varepsilon(n),$$

- Adversary A_I distinguishes encrypted m_0, m_1 *// Take the pair with the best probability of success*
- Let $f(m_i) := i$
- New adversary A_S uses A_I for m_0, m_1 ***(hardwired into the circuit M_S)***, which has success prob. $> \frac{1}{2} + \delta$
- Can $\tilde{A}$ do better as well? Nope, it's $\frac{1}{2}$.

- Think about generated vs any m_0, m_1.
- Think about uniform adversaries (algorithms, not circuits).

- *The opposite direction: next lecture.*

Takeout

- Hardcore predicate — now with a proof.
- Pseudorandom generators and how they show indistinguishability for E^{**}.
- Semantic security implies indistinguishability — now with a proof.

Coming next:

- *Indistinguishability implies semantic security — a proof.*
- *Private-key cryptography: not so simple.*
- *Digital signatures.*