Introduction to Modern Cryptography

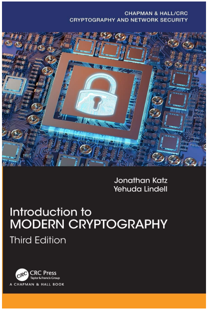
Anmoal Porwal

Technical University of Munich

20 September 2023
Finite Geometry and Friends, Brussels

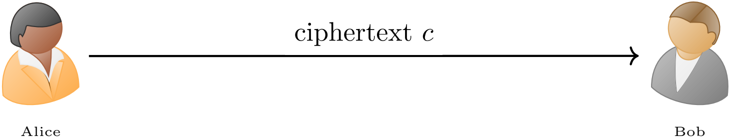
Reference

J. Katz and Y. Lindell, Introduction to Modern Cryptography. Boca Raton: CRC Press, 2021.



Background: Private-Key Encryption

Both parties share a common key k



$c = \mathrm{Enc}_k(m)$ $\qquad$ $m = \mathrm{Dec}_k(c)$

Historically...

- first rigorous definition of security was given by Shannon in 1949 called “perfect secrecy”

Historically...

- first rigorous definition of security was given by Shannon in 1949 called "perfect secrecy"
- however this definition had serious limitations

Historically...

- first rigorous definition of security was given by Shannon in 1949 called “perfect secrecy”
- however this definition had serious limitations
- these were finally overcome in the 1980s with the definition of semantic security

Modern Definition of Security

What was wrong with perfect secrecy?

Modern Definition of Security

What was wrong with perfect secrecy?

- adversary has *unbounded* computing power

Modern Definition of Security

What was wrong with perfect secrecy?

- adversary has *unbounded* computing power
- requires *zero* leak of information

Computational Security

A scheme is **secure** if all efficient adversaries succeed in breaking the scheme with small probability.

Computational Security

- efficient adversary

Computational Security

- efficient adversary
  $\rightarrow$ probabilistic polynomial-time (PPT) algorithm

Computational Security

- efficient adversary

  $\rightarrow$ probabilistic polynomial-time (PPT) algorithm
- small probability

Computational Security

- efficient adversary

  → probabilistic polynomial-time (PPT) algorithm
- small probability

  → negligible probability (e.g. 2^{-n})

Computational Security

- efficient adversary

  → probabilistic polynomial-time (PPT) algorithm
- small probability

  → negligible probability (e.g. 2^{-n})

note: all our algorithms will have a parameter n (think of it as the key length)

Computational Security

A scheme is **secure** if all efficient algorithms succeed in breaking the scheme with small probability.

Computational Security

A scheme is **secure** if all PPT algorithms succeed in breaking the scheme with negligible probability.

Definition of Security

Definition of Security

A scheme is **secure** if for every PPT algorithm $\mathcal{A}$,

$$\mathbb{P}[\mathcal{A}(\mathrm{Enc}_k(m)) = m] \leq \mathrm{negl}(n)$$

(probability taken over uniform message m and key k)

Definition of Security

A scheme is **secure** if for every PPT algorithm $\mathcal{A}$,

$$\mathbb{P}[\mathcal{A}(\mathrm{Enc}_k(m)) = m] \leq \mathrm{negl}(n)$$

(probability taken over uniform message m and key k) ✗

Definition of Security

A scheme is **secure** if for every PPT algorithm $\mathcal{A}$,
and for all i,

$$\mathbb{P}[\mathcal{A}(\mathrm{Enc}_k(m)) = \text{bit } i \text{ of } m] \leq \mathrm{negl}(n)$$

(probability taken over uniform message m and key k)

Definition of Security

A scheme is **secure** if for every PPT algorithm $\mathcal{A}$,
and for all i,

$$\mathbb{P}[\mathcal{A}(\text{Enc}_k(m)) = \text{bit } i \text{ of } m] \leq \text{negl}(n)$$

(probability taken over uniform message m and key k)

$\times$

Definition of Security

A scheme is **secure** if for every PPT algorithm $\mathcal{A}$,
and for all functions f,

$$\mathbb{P}[\mathcal{A}(\mathrm{Enc}_k(m)) = f(m)] \leq \mathrm{negl}(n)$$

(probability taken over uniform message m and key k)

Definition of Security

A scheme is **secure** if for every PPT algorithm $\mathcal{A}$,
and for all functions f,

$$\mathbb{P}[\mathcal{A}(\mathrm{Enc}_k(m)) = f(m)] \leq \mathrm{negl}(n)$$

(probability taken over uniform message m and key k) ✗

Semantic Security

Semantic Security

A scheme is **secure** if for every PPT algorithm $\mathcal{A}$,

Semantic Security

A scheme is **secure** if for every PPT algorithm $\mathcal{A}$,
... there is another PPT algorithm $\mathcal{A}'$,

Semantic Security

A scheme is **secure** if for every PPT algorithm $\mathcal{A}$,
... there is another PPT algorithm $\mathcal{A}'$,
... such that for all (efficiently sampleable) message distributions

Semantic Security

A scheme is **secure** if for every PPT algorithm $\mathcal{A}$,
... there is another PPT algorithm $\mathcal{A}'$,
... such that for all (efficiently sampleable) message distributions
... and all (polynomial-time computable) functions f and h,

Semantic Security

A scheme is **secure** if for every PPT algorithm $\mathcal{A}$,
... there is another PPT algorithm $\mathcal{A}'$,
... such that for all (efficiently sampleable) message distributions
... and all (polynomial-time computable) functions f and h,

$$|\mathbb{P}[\mathcal{A}(\mathrm{Enc}_k(m), h(m)) = f(m)] - \mathbb{P}[\mathcal{A}'(h(m)) = f(m)]|$$

is negligible

Semantic Security

A scheme is **secure** if for every PPT algorithm $\mathcal{A}$,
... there is another PPT algorithm $\mathcal{A}'$,
... such that for all (efficiently sampleable) message distributions
... and all (polynomial-time computable) functions f and h,

$$|\mathbb{P}[\mathcal{A}(\mathrm{Enc}_k(m), h(m)) = f(m)] - \mathbb{P}[\mathcal{A}'(h(m)) = f(m)]|$$

is negligible ✓

Indistinguishability

Indistinguishability

- a simpler but equivalent definition exists

Indistinguishability

- a simpler but equivalent definition exists
  define the experiment:

Indistinguishability

- a simpler but equivalent definition exists
  define the experiment:
  1. the algorithm $\mathcal{A}$ outputs two (distinct) messages m_0, m_1

Indistinguishability

- a simpler but equivalent definition exists

  define the experiment:

  1. the algorithm $\mathcal{A}$ outputs two (distinct) messages m_0, m_1
  2. the challenger encrypts one of the messages and gives it to $\mathcal{A}$

Indistinguishability

- a simpler but equivalent definition exists

  define the experiment:

  1. the algorithm $\mathcal{A}$ outputs two (distinct) messages m_0, m_1
  2. the challenger encrypts one of the messages and gives it to $\mathcal{A}$
  3. $\mathcal{A}$ outputs a guess which message was encrypted

Indistinguishability

- a simpler but equivalent definition exists

  define the experiment:

  1. the algorithm $\mathcal{A}$ outputs two (distinct) messages m_0, m_1
  2. the challenger encrypts one of the messages and gives it to $\mathcal{A}$
  3. $\mathcal{A}$ outputs a guess which message was encrypted
  4. success if guess is correct

Indistinguishability

- a simpler but equivalent definition exists

  define the experiment:

  1. the algorithm $\mathcal{A}$ outputs two (distinct) messages m_0, m_1
  2. the challenger encrypts one of the messages and gives it to $\mathcal{A}$
  3. $\mathcal{A}$ outputs a guess which message was encrypted
  4. success if guess is correct
- A scheme is **secure** if every PPT algorithm $\mathcal{A}$ succeeds with probability at most $\frac{1}{2} + \text{negl}(n)$

Provable Security?

Provable Security?

- currently we cannot *unconditionally* prove any scheme to be secure

Provable Security?

- currently we cannot *unconditionally* prove any scheme to be secure
- but it is possible to prove security based on *weaker* assumptions

Provable Security?

- currently we cannot *unconditionally* prove any scheme to be secure
- but it is possible to prove security based on *weaker* assumptions
- e.g. we construct provably secure private-key schemes from just one-way functions

Provable Security?

- currently we cannot *unconditionally* prove any scheme to be secure
- but it is possible to prove security based on *weaker* assumptions
- e.g. we construct provably secure private-key schemes from just one-way functions
- however the schemes in use today generally rely on more stronger assumptions since that yields more efficient schemes

Going further...

- stronger security notions: CPA, CCA, ...

Going further...

- stronger security notions: CPA, CCA, ...
- other schemes: authentication, public-key encryption, digital signatures

Going further...

- stronger security notions: CPA, CCA, ...
- other schemes: authentication, public-key encryption, digital signatures

Going further...

- stronger security notions: CPA, CCA, ...
- other schemes: authentication, public-key encryption, digital signatures

Thank you!