Introduction to Modern Cryptography

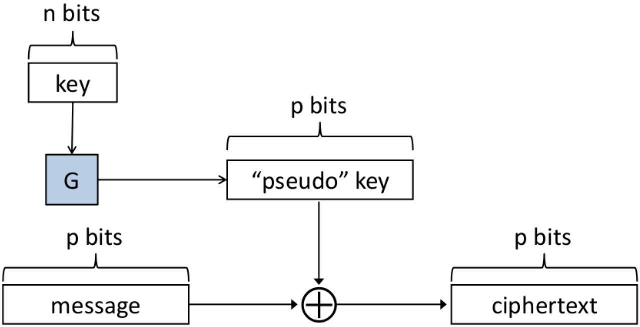
Michele Ciampi

(Slides courtesy of Prof. Jonathan Katz)

Lecture 7, Part 1

Security Against Chosen-Plaintext Attacks (CPA)

Pseudo One-time Pad (POTP) (previous lecture)



Security of POTP (previous lecture)

Theorem

If $\boldsymbol{G}$ is a pseudorandom generator, then the pseudo one-time pad $\mathbf{\Pi}$ is EAV-secure (i.e. computationally indistinguishable)

So far

- Proof that the pseudo OTP is secure...
- ...with some caveats
  - Assuming $\boldsymbol{G}$ is a pseudorandom generator
  - Relative to our definition
- The only ways the scheme can be broken are:
  - If a weakness is found in $\boldsymbol{G}$
  - **If the definition isn't sufficiently strong** (this lecture!)

Have we gained anything?

- Yes! The **POTP has a key shorter than the message**
  - $\boldsymbol{n}$ bits vs. $\boldsymbol{p(n)}$ bits
- $\Longrightarrow$ **Solved one of the limitations of the OTP**

- The fact that the parties internally generate a $\boldsymbol{p(n)}$-bit temporary string to encrypt/decrypt is irrelevant
- The **$\boldsymbol{n}$-bit key** is what the parties share in advance
- **Parties do not store the $\boldsymbol{p(n)}$-bit temporary value**

Stepping Back

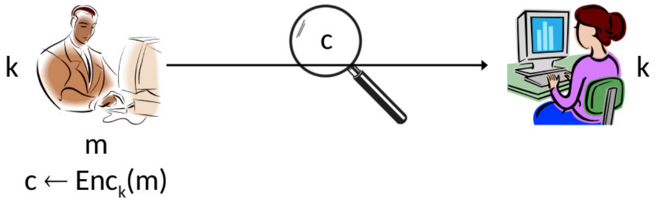
- Perfect secrecy has two limitations:
  1. Key as long as the message
  2. Key can only be used once
- We have seen how to circumvent the first (cf. POTP)
- **Does the POTP have the second limitation?**
- **How can we circumvent the second?**

But first...

- Develop an appropriate **security definition**
- Recall that security definitions have two parts
  - **Security goal:** what we want to prevent the attacker from doing
  - **Threat model:** the abilities the attacker is assumed to have
- Keep the security goal the same
  - as in indistinguishable encryption
- Strengthen the threat model

Single-message Secrecy (SMS)

SMS captures perfect secrecy and indistinguishability

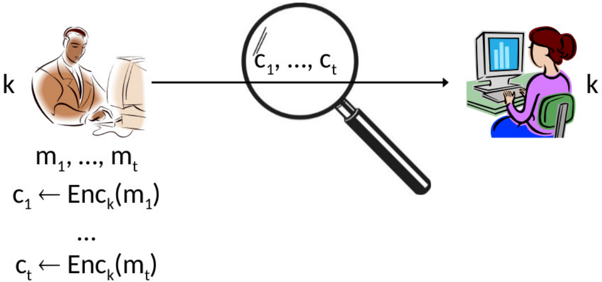


Parties share $\boldsymbol{k}$; single $\boldsymbol{m}$ encrypted under $\boldsymbol{k}$

- **Threat model:** attacker observes single ciphertext $\boldsymbol{c}$
- **Security goal:** given $\boldsymbol{c}$ attacker can not derive any information on $\boldsymbol{m}$

Multiple-message Secrecy (MMS)

MMS strenghtens the threat model of SMS



Parties share $\boldsymbol{k}$; multiple $\boldsymbol{m_i}$ encrypted under $\boldsymbol{k}$

- **Threat model:** attacker observes multiple ciphertexts $\boldsymbol{c_i}$
- **Security goal:** given $\boldsymbol{c_i}$ attacker can not derive any information on any $\boldsymbol{m_i}$

A Formal Definition

Experiment $\mathsf{PrivK}^{\mathbf{mult}}_{\boldsymbol{A},\boldsymbol{\Pi}}$

Fix $\boldsymbol{\Pi}, \boldsymbol{A}$. Define a randomized experiment $\mathsf{PrivK}^{\mathbf{mult}}_{\boldsymbol{A},\boldsymbol{\Pi}}(\boldsymbol{n})$:

1. $\boldsymbol{A(1^n)}$ outputs two vectors $\boldsymbol{(m_{0,1} \ldots m_{0,t})}$ and $\boldsymbol{(m_{1,1} \ldots m_{1,t})}$
   - Require that $\forall \boldsymbol{i}:\ |\boldsymbol{m_{0,i}}| = |\boldsymbol{m_{1,i}}|$
2. $\boldsymbol{k} \leftarrow \mathsf{Gen}(\boldsymbol{1^n})$, $\boldsymbol{b} \leftarrow \{\boldsymbol{0}, \boldsymbol{1}\}$, $\forall \boldsymbol{i}:\ \boldsymbol{c_i} = \mathsf{Enc}_{\boldsymbol{k}}(\boldsymbol{m_{b,i}})$
3. $\boldsymbol{b' = A(c_1 \ldots c_t)}$; $\boldsymbol{A}$ succeeds if $\boldsymbol{b = b'}$, and experiment evaluates to $\boldsymbol{1}$ in this case

A Formal Definition

Multiple-message Indistinguishability

$\mathbf{\Pi}$ is **multiple-message indistinguishable** if for all PPT attackers $\boldsymbol{A}$, there is a negligible function $\boldsymbol{\epsilon}$ such that

$$\mathbf{Pr}[\mathsf{PrivK}^{\mathbf{mult}}_{\boldsymbol{A},\mathbf{\Pi}}(\boldsymbol{n}) = \mathbf{1}] \leq \frac{\mathbf{1}}{\mathbf{2}} + \boldsymbol{\epsilon}(\boldsymbol{n})$$

A Formal Definition

Claim

POTP is not multiple-message indistinguishable

Attack

$\boldsymbol{A}$ outputs $(\boldsymbol{m_{0,0}, m_{0,1}})$ and $(\boldsymbol{m_{1,0}, m_{1,1}})$ s.t.

$$\boldsymbol{m_{0,0} = m_{0,1} = m_{1,0} \neq m_{1,1}}$$

If $\boldsymbol{c_0 = c_1}$ then $\boldsymbol{A}$ outputs $\boldsymbol{b' = 0}$; otherwise $\boldsymbol{b' = 1}$ i.e. $\boldsymbol{A}$ wins the $\mathsf{PrivK}^{\mathbf{mult}}_{\boldsymbol{A,\Pi}}(\boldsymbol{n})$ game with $\mathbf{Pr = 1}$

Multiple-message Secrecy

Fact

No **deterministic** encryption scheme is multiple-message indistinguishable

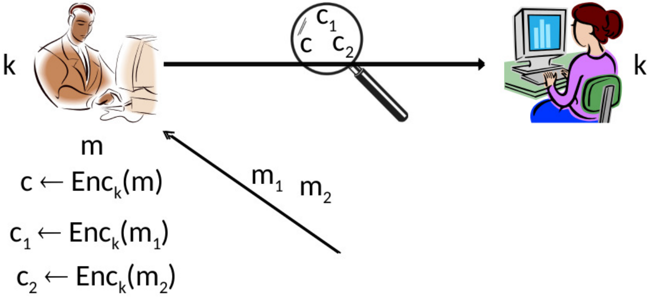
- The issue is not an artefact of our definition
- It is a problem in practise if an attacker can tell when **the same message is encrypted twice**
- Need to consider **randomized** schemes!

Multiple-message Secrecy

- We shall not work with **multiple-message indistinguishability**
- Instead, define something stronger:
- **Security against chosen-plaintext attacks (CPA-security)**
- CPA is **the minimal notion of security** an encryption scheme should satisfy

If $\mathbf{\Pi}$ is CPA-secure $\implies$ $\mathbf{\Pi}$ is multiple-message indist.

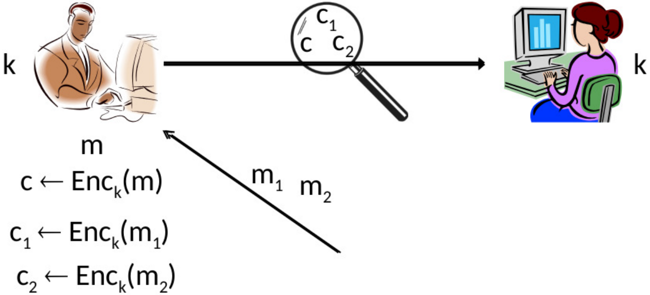
CPA-security



Threat model

- Attacker $\boldsymbol{A}$ can request encryption of any $\boldsymbol{m_i}$ of his choice
- i.e. $\boldsymbol{A}$ is given **access to an encryption oracle** $\boldsymbol{E_k}$

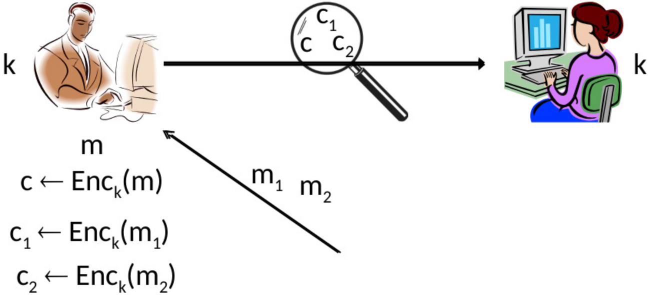
CPA-security



Threat model

$\boldsymbol{A}$ submits $\boldsymbol{m_1} \implies$ obtains $\boldsymbol{c_1 = E_k(m_1)}$

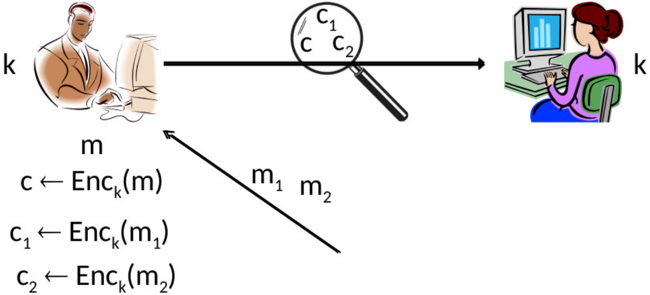
CPA-security



Threat model

A submits $m_2 \implies$ obtains $c_2 = E_k(m_2)$

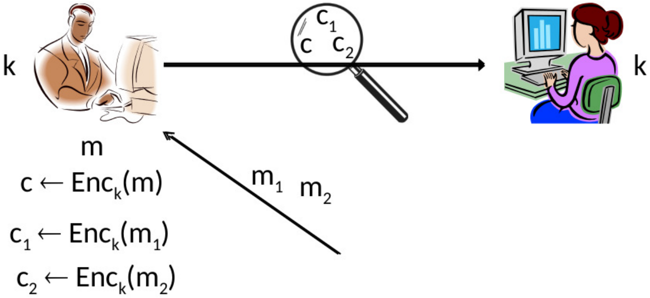
CPA-security



Threat model

$\boldsymbol{A}$ submits $\boldsymbol{m_i} \implies$ obtains $\boldsymbol{c_i = E_k(m_i)}$: $\boldsymbol{i = 1, 2, \ldots}$

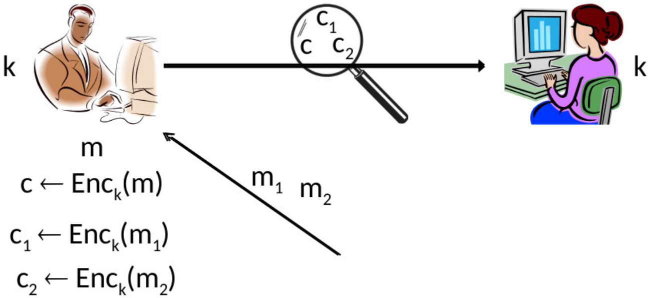
CPA-security



Threat model

- At some point an unknown (to A) message m is encrypted
- Attacker observes $c = E_k(m)$

CPA-security



Security goal

- Given $\boldsymbol{c}$ attacker can not derive any information on $\boldsymbol{m}$

Is the threat model too strong?

- In practice, there are many ways an attacker can influence what gets encrypted
- Not clear how best to model this
- **Chosen-plaintext attacks encompass any such influence**
- In some cases an attacker may have complete control over what gets encrypted

CPA-security

Experiment $\mathsf{PrivK}^{\mathbf{cpa}}_{A,\Pi}(n)$

Fix Π, A. Define a randomized experiment $\mathsf{PrivK}^{\mathbf{cpa}}_{A,\Pi}(n)$:

- $k \leftarrow \mathsf{Gen}(1^n)$
- $A(1^n)$ **interacts with an encryption oracle** $\mathsf{Enc}_k(\cdot)$, and then outputs m_0, m_1 of the same length
- $b \leftarrow \{0, 1\}$, $c \leftarrow \mathsf{Enc}_k(m_b)$, give c to A
- A can **continue to interact with** $\mathsf{Enc}_k(\cdot)$
- A outputs b'; A succeeds if $b = b'$, and the experiment evaluates to 1 in this case

CPA-security

Security Against Chosen-plaintext Attacks

$\mathbf{\Pi}$ is secure against chosen-plaintext attacks (CPA-secure) if for all PPT attackers $\boldsymbol{A}$, there is a negligible function $\boldsymbol{\epsilon}$ such that

$$\mathbf{Pr}[\mathsf{PrivK}^{\mathbf{cpa}}_{\boldsymbol{A},\mathbf{\Pi}}(\boldsymbol{n}) = \mathbf{1}] \leq \frac{\mathbf{1}}{\mathbf{2}} + \boldsymbol{\epsilon}(\boldsymbol{n})$$

Relation with Previous Definitions

- CPA-security is stronger than multiple-message indistingiushability
- i.e. if $\mathbf{\Pi}$ is CPA-secure then it is also multiple-message indistinguishable

Fact

No **deterministic** encryption scheme is multiple-message indistinguishable

Corollary

No **deterministic encryption** scheme can be CPA-secure

CPA against Deterministic Encryption Schemes

Attacker $\boldsymbol{A}$ attacks deterministic scheme $\boldsymbol{\Pi} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$

1. Query the ecnryption oracle on $\boldsymbol{m_0}$ and $\boldsymbol{m_1}$
2. Obtain $\boldsymbol{c_0} = \mathsf{Enc}_{\boldsymbol{k}}(\boldsymbol{m_0})$, $\boldsymbol{c_1} = \mathsf{Enc}_{\boldsymbol{k}}(\boldsymbol{m_1})$
3. Output $\boldsymbol{m_0}, \boldsymbol{m_1}$; get challenge $\boldsymbol{c}$
4. If $\boldsymbol{c} = \boldsymbol{c_0}$ output $\mathbf{0}$; if $\boldsymbol{c} = \boldsymbol{c_1}$ output $\mathbf{1}$

- $\boldsymbol{A}$ succeeds with $\mathbf{Pr} = \mathbf{1}$
- Is CPA-security impossible to achieve?

End

References: Section 3.4 until Pag. 76