Lecture 1. Introduction

Definitions and one-time pad

Bruno Grenet

https://membres-ljk.imag.fr/Bruno.Grenet/IntroCrypto.html

Introduction to cryptology
Université Grenoble Alpes – IM²AG
M1 INFO, MOSIG & AM

What is cryptography?

Protecting secret data from adversaries

- Communications (email, web, credit card payment, …)
- Storage (encrypted hard drive, …)
- Computations (electronic voting, …)
- …

Used with various hardware

- High-end CPUs, mobile phones, microcontrollers, dedicated hardware
- Varying speed (throughput & latency), code/circuit size, energy consumption, …

"Doing crypto"

- Designing new primitives, constructions, protocols, …
- Analysing existing primitives, …
- Deploying crypto in products incl. implementation

What is this course about?

- Basic concepts of cryptography — *adversary model, security definition, …*
- Cryptographic constructions — *block cipher, key exchange, …*
- Some standard attacks — *birthday attack, …*
- Real-life usage — *What's inside TLS?*

But not (really) about

- Implementation
- Usage of existing standard cryptographic softwares, libraries, …

Historical ciphers

- Shift ciphers — *Caesar (50 BC); `rot13`*
- Substitution ciphers — *Atbash (600-500 BC)*
- Transposition ciphers — *Scytale (400 BC)*
- Polyalphabetic cipher — *Vigenère (1553); Enigma (1920s)*
- …

None of them is safe!

- Attacks: brute force, frequency analysis (1863), bombe (1938-40), …
- Some lessons drawn:
  - Large enough *key space* is needed
  - Designing an encryption system is *difficult*
  - Assessing its security is (even more?) difficult

Modern cryptography

What does it mean to be *secure*?

Defining security

Achieve some *goals*

- Confidentiality — *no adversary can read the data*
- Proof of identity — *that's me!*
- Authenticity — *no adversary can impersonate the sender*
- Integrity — *no adversary can modify the data*
- …

In the presence of *adversaries*

- Passive adversary — *eavesdropper*
- Active adversary — *can modify messages exchanged*

With or without *shared secret*

- Large — *one-time pad*
- Small — *symmetric cryptography*
- None — *public-key cryptography*

Example of a protocol: TLS

Goals

- Confidentiality — *no adversary can read the data*
- Authenticity — *no adversary can impersonate the sender*
- Integrity — *no adversary can modify the data*

Some ingredients

- Key exchange — *e.g.* Diffie-Hellman
  - *public-key (a.k.a. **a**symmetric) cryptography*
- Authenticated encryption — *e.g* using AES
  - *symmetric cryptography*
- Signatures — *e.g.* ECDSA
  - *public-key cryptography, using symmetric primitive*

The big picture of this course

	small shared secret symmetric cryptography	no shared secret public-key cryptography
passive adversaries confidentiality	*symmetric encryption*	*public-key encryption*
active adversaries authenticity	*message authentication codes*	*digital signatures*
active adversaries confidentiality+authenticity	*authenticated encryption*	*signcryption*

Other constructions

- Confidentiality using a large secret — *one-time pad & block ciphers*
- Sharing a secret in the presence of passive adversaries — *key exchange*
- Combine symmetric and public-key cryptography — *hybrid encryption*
- A very central tool — (cryptographic) *hash functions*

Course contents

Lectures

1.	Introduction	*One-time pad*
2.	Block ciphers	*AES*, *DES*
3.	Symmetric encryption	*CBC* & *CTR* modes of operation
4.	Hash functions	*SHA-2*, *SHA-3*
5.	Messages authentication codes & authenticated encryption	*CBC-MAC*, *HMAC*, *GCM*
6.	Key exchange	*Diffie-Hellman*
7.	Public-key encryption & key encapsulation	*ElGamal*
8.	Signatures	*Schnorr*, *DSA*
9.	RSA	
10.	Putting it all together	*TLS*

Common topics

- Definitions and security notions
- Proofs of security
- Examples & attacks

Contents

Contents

The one-time pad

Input: *Message* $m \in \{0,1\}^\ell$ (or *plaintext*)
Secret: Key $k \in \{0,1\}^\ell$
Output: *Ciphertext* $c \in \{0,1\}^\ell$

message space: $\mathcal{M} = \{0,1\}^\ell$
key space: $\mathcal{K} = \{0,1\}^\ell$
ciphertext space: $\mathcal{C} = \{0,1\}^\ell$

Encryption: $\mathsf{Enc}_k(m) = m \oplus k$
Decryption: $\mathsf{Dec}_k(c) = c \oplus k$

Correctness: $\mathsf{Dec}_k(\mathsf{Enc}_k(m)) = (m \oplus k) \oplus k = m \oplus (k \oplus k) = m \oplus 0 = m$

Pros

- Used during the cold war
- Used for small plaintexts/secrets
- *Perfectly secret/indistinguishable*

Cons & caveats

- Key as long as the message
- Can be used only once
- The key must be uniformly sampled

Perfect indistinguishability

a.k.a information-theoretic security, *a.k.a.* unconditional security

The ciphertext provides no (extra) information about the message to an adversary

Indistinguishability **game** for an encryption Enc

Adversary	chooses two messages $m_0, m_1 \in \mathcal{M}$
Challenger	samples $k \twoheadleftarrow \mathcal{K}$ and $b \twoheadleftarrow \{0,1\}$ and computes $c \leftarrow \mathsf{Enc}_k(m_b)$
Adversary	receives c and outputs $\hat{b}$ (*tries to guess* b)

Definition

Enc is *perfectly indistinguishable* if whatever the adversary does, $\Pr\left[\hat{b} = b\right] = \frac{1}{2}$

(probability taken over the random choices for k and b *and whatever randomness is used by the Adversary*)

Theorem

One-time pad encryption is perfectly indistinguishable

Proof of the theorem

$M = C = K = \{0,1\}^\ell$

Theorem restated

No matter what $\mathcal{A}$ does, $\Pr\left[\hat{b} = b\right] = \frac{1}{2}$

Proof

- Lemma: Fix $m \in M$ and $c \in C$. Take $k \xleftarrow{R} K$. Then $\Pr_k[m \oplus k = c] = 1/2^\ell$

  Pf. $\Pr_k[m \oplus k = c] = \Pr_k[k = m \oplus c] = 1/2^\ell$ ∎

- $\Pr[\hat{b} = b] = \Pr[\hat{b} = 0 \mid b = 0]\,\Pr[b = 0] + \Pr[\hat{b} = 1 \mid b = 1]\,\Pr[b = 1]$

  $\Pr[b = 0] = 1/2$; $\Pr[b = 1] = 1/2$; $\mathcal{A}_b \to 0$; $\mathcal{A}_b \to 1$

- Assume that $\mathcal{A}_b$ is deterministic $\rightsquigarrow$ $C = C_0 \sqcup C_1$

  $\Pr[\hat{b} = 0 \mid b = 0] = \Pr_k[c \in C_0 \mid b = 0] = \#C_0 / 2^\ell$ and $\Pr[\hat{b} = 1 \mid b = 1] = \#C_1 / 2^\ell$

  $\Rightarrow \Pr[\hat{b} = b] = \frac{1}{2}\left(\#C_0 + \#C_1\right)/2^\ell = \frac{1}{2}\,2^\ell/2^\ell = \frac{1}{2}$.

Proof of the theorem

Theorem restated

No matter what $\mathcal{A}$ does, $\Pr\left[\hat{b} = b\right] = \frac{1}{2}$

Proof

- Assume now that A_b is randomized $\rightsquigarrow$ it uses random bits $r \in \{0,1\}^*$

For each choice of random bits, we have a deterministic alg. $A_{b,r}$.

$$\Pr_{b,r}\left[\hat{b} = b\right] = \sum_{r \in \{0,1\}^*} \underbrace{\Pr\left[\hat{b} = b \mid A_b \text{ uses } r \text{ as random bits}\right]}_{1/2} \Pr\left[A_b \text{ uses } r \dots\right]$$

$$= 1/2 \sum_{r \in \{0,1\}^*} \Pr\left[A_b \text{ uses } r \text{ as random bits}\right] = 1/2$$

Perfect indistinguishability implies no information leak

Examples

Given $c \leftarrow \mathrm{Enc}_k(m)$, no adversary can learn

- the least significant bit $m_{[0]}$ of m
- the parity $\bigoplus_{i=0}^{\ell-1} m_{[i]}$ of m
- whether m contains more ones than zeroes
- …

Equivalent definition: the distribution of m does not depend on c

- For any fixed $m^* \in \mathcal{M}$ and $c^* \in \mathcal{C}$ s.t. $\Pr[c = c^*] > 0$,

$$\Pr_{m,k}\left[m = m^* \middle| c = c^*\right] = \Pr_{m,c,k}\left[m = m^*\right]$$

Limitations of perfect secrecy: Shannon's theorem

Theorem

A *perfectly indistinguishable* encryption scheme must satisfy:

(i) $\#\mathcal{K} \geq \#\mathcal{M}$

(ii) if $\#\mathcal{K} = \#\mathcal{M}$, k must be uniformly sampled from $\mathcal{K}$

Proof of (i)

- We assume $\#K < \#M$ and build A_0 s.t. $Pr[\hat{b} = b] > 1/2$
- For $c \in C$, $M_c = \{ m \in M : \exists k, Dec_k(c) = m \}$

  Since Dec is det., $\#M_c \leq \#K < \#M$
- Take $c^* \in C$ s.t. $M_{c^*} \neq \phi$, and $m_0 \in M_{c^*}$ and $m_1 \notin M_{c^*}$.
- A_0 receives c : - if $c = c^*$, A_0 outputs $\hat{b} = 0$ (c^* cannot decrypt to m_1)
  - otherwise it outputs a random bit $\hat{b}$.
- $Pr[\hat{b} = b] = Pr[\hat{b} = b \mid c = c^*] Pr[c = c^*] + Pr[\hat{b} = b \mid c \neq c^*] Pr[c \neq c^*] = 1/2 + 1/2\,p > 1/2$ (with $Pr[\hat{b} = b \mid c = c^*] = 1$, $Pr[c = c^*] = p$, $Pr[\hat{b} = b \mid c \neq c^*] = 1/2$, $Pr[c \neq c^*] = 1 - p$)

Conclusion on one-time pad

- One-time pad: perfectly indistinguishable but…
- … perfect indistinguishability impossible with *small* keys

Relaxation of the security notion

- Allow to recover *some* (very little!) information — *statistical security*
- Put a limit on the computational power of an adversary — *computational security*

Another problem

An adversary can modify any message — *malleability*

- $c = m \oplus k \implies c \oplus m' = (m \oplus m') \oplus k$

Contents

Principles of modern cryptography

Formal definitions

- What does *secure encryption* mean?
  - An adversary cannot recover the key
  - An adversary cannot recover the message from the ciphertext
  - An adversary cannot retrieve any character of the message from the ciphertext
  - …

Principles of modern cryptography

Formal definitions

- What does *secure encryption* mean?
  - (good definition) Whatever information an adversary has about the message, the ciphertext only provides them with *very little* additional information

Principles of modern cryptography

Formal definitions

- What does *secure encryption* mean?
  - (good definition) Whatever information an adversary has about the message, the ciphertext only provides them with *very little* additional information
- What is an *adversary*?
  - Ciphertext-only attack COA
  - Known-plaintext attack KPA
  - Chosen-plaintext attack CPA
  - Chosen-ciphertext attack CCA

Principles of modern cryptography

Formal definit

- What does
  - (good
    cipher
- What is an
  - Ciphe
  - Known
  - Chose
  - Chose

message, the

COA
KPA
CPA
CCA

https://xkcd.com/257/

Principles of modern cryptography

Formal definitions

- What does *secure encryption* mean?
  - (good definition) Whatever information an adversary has about the message, the ciphertext only provides them with *very little* additional information
- What is an *adversary*?
  - ~~Ciphertext-only attack~~ ~~COA~~
  - ~~Known-plaintext attack~~ ~~KPA~~
  - Chosen-plaintext attack CPA
  - Chosen-ciphertext attack CCA

Principles of modern cryptography

Formal definitions

- What does *secure encryption* mean?
  - (good definition) Whatever information an adversary has about the message, the ciphertext only provides them with *very little* additional information
- What is an *adversary*?
  - ~~Ciphertext-only attack~~ ~~COA~~
  - ~~Known-plaintext attack~~ ~~KPA~~
  - Chosen-plaintext attack CPA
  - Chosen-ciphertext attack CCA

Specific assumptions

- Computational power of an adversary (complexity theory)
- Validity of assumptions, comparison between them and *necessary* assumptions

Provable security

Proving that a protocol satisfies a *security definition*, assuming *assumptions*.

Back to perfect indistinguishability

Reminder: perfect indistinguishability for Enc using a game

Adversary chooses two messages $m_0, m_1 \in \mathcal{M}$

Challenger samples $k \twoheadleftarrow \mathcal{K}$, $b \twoheadleftarrow \{0,1\}$ and computes $c \leftarrow \mathsf{Enc}_k(m_b)$

Adversary outputs a bit $\hat{b}$ (*tries to guess* b)

- Enc is perfectly indistinguishable if $\Pr\left[\hat{b} = b\right] = \frac{1}{2}$

Characteristics

- Weak model of adversary: COA
  - Adversary only sees c
  - Even with access to many other ciphertexts, no difference!
  - But not KPA $\rightarrow$ one pair $(m, \mathsf{Enc}_k(m))$ reveals k *one-time* pad
- Strong guarantees: no matter how powerful, the adversary learns nothing

Need a stronger model of adversary, but weaker guarantees is sufficient

IND-CPA: Indistinguishability under Chosen-Plaintext Attack

IND-CPA game for Enc

Challenger	samples $k \leftarrow\!\!\$\ \mathcal{K}$
Adversary	has *oracle access* to $\mathrm{Enc}_k(\cdot)$: on *query* x_i, gets $c_i \leftarrow \mathrm{Enc}_k(x_i)$
	creates two messages $m_0, m_1 \in \mathcal{M}$ of same length
Challenger	samples $b \leftarrow\!\!\$\ \{0,1\}$ and computes $c \leftarrow \mathrm{Enc}_k(m_b)$
Adversary	outputs a bit $\hat{b}$ (*tries to guess* b)

Remarks

- The adversary can query $\mathrm{Enc}_k(m_0)$ and $\mathrm{Enc}_k(m_1)$
  - $\mathrm{Enc}_k(.)$ must be *randomized*
- Messages of same length
  - IND-CPA encryption may reveal the length of the message
  - Necessary for efficiency in general
  - *Ad-hoc* solution if the length is a sensitive information

Relaxing the guarantees: IND-CPA *advantage*

What is the *advantage* of an adversary $\mathcal{A}$ compared to choosing $\hat{b}$ at random?

Advantage of adversary $\mathcal{A}$

$$\mathsf{Adv}^{\mathsf{IND-CPA}}_{\mathsf{Enc}}(\mathcal{A}) = 2\left|\Pr\left[\mathcal{A}^{\mathsf{Enc}_k} \to b\right] - \frac{1}{2}\right|$$

Remarks

- Advantage between 0 (perfectly indistinguishable) and 1
- $\mathsf{Adv}^{\mathsf{IND-CPA}}_{\mathsf{Enc}}(\mathcal{A}) = \left|\Pr\left[\mathcal{A}^{\mathsf{Enc}_k} \to 1 \middle| b = 1\right] - \Pr\left[\mathcal{A}^{\mathsf{Enc}_k} \to 1 \middle| b = 0\right]\right|$

The IND-CPA security

IND-CPA secure: an adversary with *bounded resources* has *negligible* advantage

Asymptotic security

complexity-theoretic definition

- Fix a *security parameter* n
- Bounded resources: randomized polynomial-time adversaries $\mathcal{A}$ poly. in n
- Negligible advantage: if $< 1/p(n)$ for every polynomial p

Concrete security

chosen in this course

- *Advantage function*: $\mathrm{Adv}^{\mathrm{IND-CPA}}_{\mathrm{Enc}}(q, t) = \max_{\mathcal{A}_{q,t}} \mathrm{Adv}^{\mathrm{IND-CPA}}_{\mathrm{Enc}}(\mathcal{A}_{q,t})$
  where $\mathcal{A}_{q,t}$ ranges over all (randomized) algorithms that run in time $\leq t$ and make $\leq q$ *queries*
- No formal definition of *bounded resources* or *negligible*:
  - Compare two schemes by comparing their advantage functions
  - Bound the advantage for some fixed explicit bounds q and t

Orders of magnitude (time)

Computational time

- $t \simeq 2^{40}$: $\sim$ 1 day on my laptop
- $t \simeq 2^{60}$: possible on a large CPU/GPU cluster — done in academia
- $t \simeq 2^{80}$: possible with an ASIC cluster — Bitcoin mining
- $t \simeq 2^{128}$: seems hard enough

Example: perform 2^{128} operations within 34 years ($\simeq 2^{30}$ seconds)

- Hypotheses:
  - Hardware at 2^{50} op/s — *quite fast*
  - Hugely parallelizable — *not always true*
  - 1000 W per device — *quite good*
- Results:
  - Require $\simeq 2^{128}/(2^{50} \cdot 2^{30}) = 2^{48}$ machines — $> 280 \cdot 10^{12}$
  - Require $\simeq 280\,000$ TW — $> 1.7 \cdot 10^{9}$ EPR

Orders of magnitude (probabilities)

Probabilities

- $p = \frac{1}{2}$: get a TAIL with a fair coin
- $p = \frac{1}{6}$: get a 6 with a fair die
- $p \simeq 2^{-24}$: probability to win at French lottery
- $p \simeq 2^{-72}$: probability to win 3 times in a row at French lottery

Example

- An attack that takes 1 second and has a probability of success of 2^{-60} is expected to have succeeded less than once since the Big Bang
- Errors in CPU due to cosmic rays happen with much larger probabilities!

Combining orders of magnitude

If $\mathsf{Adv}^{\mathsf{IND-CPA}}_{\mathsf{Enc}}(2^{128}) < 2^{-60}$, Enc can be considered (IND-CPA) secure

Orders of magnitude (probabilities)

Probabilities

- $p = \frac{1}{2}$: get a TAIL with a fa
- $p = \frac{1}{6}$: get a 6 with a fair
- $p \simeq 2^{-24}$: probability to w
- $p \simeq 2^{-72}$: probability to w

h lottery

Example

- An attack that takes 1 sec
- have succeeded less than
- Errors in CPU due to cosm

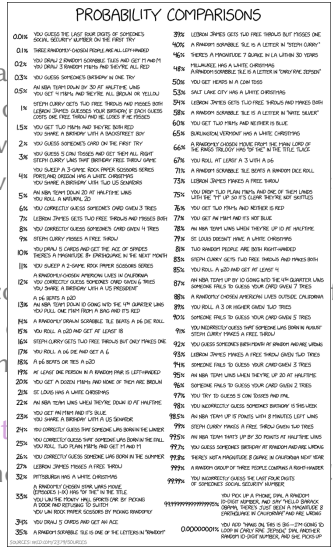
f success of 2^{-60} is expected to

larger probabilities!

Combining orders of magnit

If $\mathrm{Adv}^{\mathrm{IND-CPA}}_{\mathrm{Enc}}(2^{128}) < 2^{-60}$, En

PA) secure



https://xkcd.com/2379/

Conclusion

One-time pad

- First example of encryption scheme
- Strong security… in a very weak model!
- Vastly insufficient in practice

Computational security

- Game + advantage $\rightarrow$ security notion
- Various security models, depending on the experiment
  - Fix goals & means

What's next?

- Symmetric and public-key encryption
- Authentication and integrity
- Each time:
  - What is the suitable security notion
  - How to achieve this security notion?

Recap on probability theory

Notations

- $r \twoheadleftarrow \mathcal{S}$: *r uniformly sampled from* $\mathcal{S}$ $\qquad \forall s \in \mathcal{S}, \Pr[r = s] = 1/\#\mathcal{S}$
- $\Pr_x[\texttt{event}]$: probability of $\texttt{event}$, insisting on where the randomness lies
- $\Pr[E|F]$ ("prob. of E given F"): probability that E happens *knowing that* F happens

Law of total probabilities *proof by exhaustion/case*

- If E, F are events, $\Pr[E] = \Pr[E|F]\Pr[F] + \Pr[E|\neg F]\Pr[\neg F]$
- Generalization for $\Omega = \bigsqcup_i F_i$, $\Pr[E] = \sum_i \Pr[E|F_i]\Pr[F_i]$

Other useful tools

- $\Pr[\neg E] = 1 - \Pr[E]$
- Union bound: $\Pr[E \vee F] \leq \Pr[E] + \Pr[F]$ $\qquad \Pr[\bigvee_i E_i] \leq \sum_i \Pr[E_i]$
- If E and F are independent, $\Pr[E \wedge F] = \Pr[E]\Pr[F]$

For self-learning / refresh

- B. Barak, *An Intensive Introduction to Cryptography*.
  Chapter 0. Mathematical background *(also available from the course webpage)*

Recap on (symmetric) encryption

Encryption scheme

- Three spaces: messages $\mathcal{M}$, keys $\mathcal{K}$, ciphertexts $\mathcal{C}$
- Two functions: $\mathsf{Enc} : \mathcal{K} \times \mathcal{M} \to \mathcal{C}$ and $\mathsf{Dec} : \mathcal{K} \times \mathcal{C} \to \mathcal{M}$

Notations: $c \leftarrow \mathsf{Enc}_k(m)$ and $\hat{m} \leftarrow \mathsf{Dec}_k(c)$

Correctness: for all $m \in \mathcal{M}$ and $k \in \mathcal{K}$, $\mathsf{Dec}_k(\mathsf{Enc}_k(m)) = m$

IND-CPA security

Game: an Adversary

- has *oracle access* to $\mathsf{Enc}_k(\cdot)$
- submits two *challenges* m_0 and m_1 of same length and gets c
- must decide whether c encrypts m_0 or m_1

Advantage: *distance* from the success probability of a random guess, *normalized* to $[0, 1]$

$$2\left|\Pr[\text{success}] - \frac{1}{2}\right|$$

Security: *resource-bounded* adversaries must have *negligible* advantage