Introduction

Yu Zhang

Harbin Institute of Technology

Cryptography, Autumn, 2018

Outline

Content

What is Cryptography?

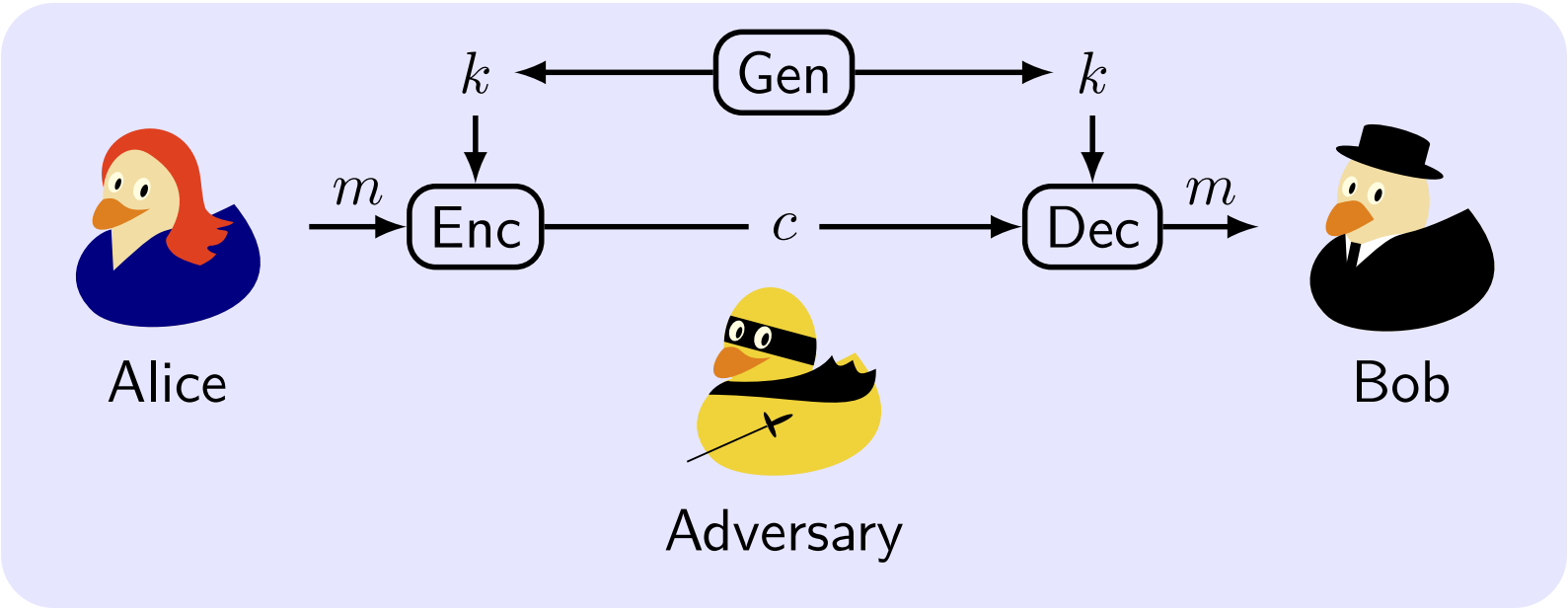
- **Cryptography**: from Greek *kryptós*, "hidden, secret"; and *gráphin*, "writing"
- **Cryptography**: the art of writing or solving codes. (Concise oxford dictionary 2006)
- **Codes**: a system of prearranged signals, especially used to ensure secrecy in transmitting messages. (*code word* in cryptography)
- **1980s**: from Classic to Modern; from Military to Everyone
- **Modern cryptography**: the scientific study of mathematical techniques for securing digital information, systems, and distributed computations against adversarial attacks

Content

Private-Key Encryption

- **Goal**: to construct a **ciphers** (encryption schemes) for providing secret communication between two parties sharing **private-key** (the symmetric-key) in advance
- **Implicit assumption**: there is some way of initially sharing a key in a secret manner
- **Disk encryption**: the same user at different points in time

The Syntax of Encryption



- key $k \in \mathcal{K}$, plaintext (or message) $m \in \mathcal{M}$, ciphertext $c \in \mathcal{C}$
- **Key-generation** algorithm $k \leftarrow \mathsf{Gen}$
- **Encryption** algorithm $c := \mathsf{Enc}_k(m)$
- **Decryption** algorithm $m := \mathsf{Dec}_k(c)$
- **Encryption scheme**: $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$
- **Basic correctness requirement**: $\mathsf{Dec}_k(\mathsf{Enc}_k(m)) = m$

Securing Key vs Obscuring Algorithm

- Easier to maintain secrecy of a short key
- In case the key is exposed, easier for the honest parties to change the key
- In case many pairs of people, easier to use the same algorithm, but different keys

Kerckhoffs's principle

The cipher method must not be required to be secret, and it must be able to fall into the hands of the enemy without inconvenience.

Why "Open Cryptographic Design"

- Published designs undergo public scrutiny are to be stronger
- Better for security flaws to be revealed by "ethical hackers"
- Reverse engineering of the code (or leakage by industrial espionage) poses a serious threat to security
- Enable the establishment of standards.

Attack Scenarios

- **Ciphertext-only**: the adversary just observes ciphertext
- **Known-plaintext**: the adversary learns pairs of plaintexts/ciphertexts under the same key
- **Chosen-plaintext**: the adversary has the ability to obtain the encryption of plaintexts of its choice
- **Chosen-ciphertext**: the adversary has the ability to obtain the decryption of **other** ciphertexts of its choice
- **Passive attack**: COA KPA
  - because not all ciphertext are confidential
- **Active attack**: CPA CCA
  - when to encrypt/decrypt whatever an adversary wishes?

Content

Caesar's Cipher

If he had anything confidential to say, he wrote it in cipher, that is, by so changing the order of the letters of the alphabet, that not a word could be made out. If anyone wishes to decipher these, and get at their meaning, he must substitute the fourth letter of the alphabet, namely D, for A, and so with the others

–Suetonius, "Life of Julius Caesar"

- $\text{Enc}(m) = m + 3 \mod 26$ [1]
- **Weakness**: What is the key?

Example

`begintheattacknow`

[1] In fact the quote indicates that decryption involved rotating letters of the alphabet forward 3 positions, $\text{Dec}(c) = c + 3 \mod 26$

Shift Cipher

- $\text{Enc}_k(m) = m + k \mod 26$
- $\text{Dec}_k(c) = c - k \mod 26$
- **Weakness**: Fragile under **Brute-force attack** (exhaustive search)

Example: Decipher the string

`EHJLQWKHDWWDFNQRZ`

Sufficient Key Space Principle

Any secure encryption scheme must have a key space that is not vulnerable to exhaustive search.[2]

[2] If the plaintext space is larger than the key space.

Index of Coincidence (IC) Method (to find k)

Index of Coincidence (IC): the probability that two randomly selected letters (pick-then-return) will be identical.
Let p_i denote the probability of ith letter in English text.

$$I \stackrel{\text{def}}{=} \sum_{i=0}^{25} p_i^2$$

Example

What's the IC of 'apple'?

For a long English text, the IC is ≈ 0.065. For $j = 0, 1, \ldots, 25$, q_j is the probability of jth letter in the ciphertext.

$$I_j \stackrel{\text{def}}{=} \sum_{i=0}^{25} p_i \cdot q_{i+j}$$

Q: For shift cipher, if $j = k$, then $I_j \approx$?

Mono-Alphabetic Substitution

- **Idea**: To map each character to a different one in an arbitrary manner
- **Strength**: Key space is large $\approx 2^{88}$. Q: how to count?
- **Weakness**: The mapping of each letter is fixed

Example

`abcdefghijklmnopqrstuvwxyz`
`XEUADNBKVMROCQFSYHWGLZIJPT`
Plaintext: `tellhimaboutme`
Ciphertext: `????????????`

Attack with Statistical Patterns

1. Tabulate the frequency of letters in the ciphertext
2. Compare it to those in English text
3. Guess the most frequent letter corresponds to e, and so on
4. Choose the plaintext that does "make sense" (Not trivial)

Table: Average letter frequencies for English-language text

e	12.7%	t	9.1%	a	8.2%	o	7.5%	i	7.0%
n	6.7%	_	6.4%	s	6.3%	h	6.1%	r	6.0%
d	4.3%	l	4.0%	c	2.8%	u	2.8%	m	2.4%
w	2.4%	f	2.2%	g	2.0%	y	2.0%	p	1.9%
b	1.5%	v	1.0%	k	0.8%	j	0.2%	x	0.2%
q	0.1%	z	0.1%						

Example of Frequency Analysis (Ciphertext)

LIVITCSWPIYVEWHEVSRIQMXLEYVEOIEWHRXEXIPFEMVEWHKVS
TYLXZIXLIKIIXPIJVSZEYPERRGERIMWQLMGLMXQERIWGPSRIH
MXQEREKIETXMJTPRGEVEKEITREWHEXXLEXXMZITWAWSQWXSWE
XTVEPMRXRSJGSTVRIEYVIEXCVMUIMWERGMIWXMJMGCSMWXSJO
MIQXLIVIQIVIXQSVSTWHKPEGARCSXRWIEVSWIIBXVIZMXFSJX
LIKEGAEWHEPSWYSWIWIEVXLISXLIVXLIRGEPIRQIVIIBGIIHM
WYPFLEVHEWHYPSRRFQMXLEPPXLIECCIEVEWGISJKTVWMRLIHY
SPHXLIQIMYLXSJXLIMWRIGXQEROIVFVIZEVAEKPIEWHXEAMWY
EPPXLMWYRMWXSGSWRMHIVEXMSWMGSTPHLEVHPFKPEZINTCMXI
VJSVLMRSCMWMSWVIRCIGXMWYMX

Example of Frequency Analysis (Analysis)

Count and Guess, Trial and Error.

Table: Analysis Steps

Ciphertext	Plaintext
I	e
XLI	the
E	a
Rtate	state
atthattMZe	atthattime
heVe	here
remarA	remark

Example of Frequency Analysis (Plaintext)

Hereupon Legrand arose, with a grave and stately air, and brought me the beetle from a glass case in which it was enclosed. It was a beautiful scarabaeus, and, at that time, unknown to naturalists – of course a great prize in a scientific point of view. There were two round black spots near one extremity of the back, and a long one near the other. The scales were exceedingly hard and glossy, with all the appearance of burnished gold. The weight of the insect was very remarkable, and, taking all things into consideration, I could hardly blame Jupiter for his opinion respecting it.

–Edgar Allan Poe's "The Gold-Bug"

Vigenère (poly-alphabetic shift) Cipher

- **Idea**: To "smooth out" the distribution in the ciphertext by mapping different instances of the same letter in the plaintext to different ones in the ciphertext
- **Encryption**: $c_i = m_i + k_{[i \bmod t]}$, t is the length (period) of k
- **Cryptanalysis**: Need find t; if t is known, need know whether the decryption "makes sense", but brute force (26^t) is infeasible for $t > 15$

Example (Key is 'cafe')

Plaintext	`tellhimaboutme`
Key	`cafecafecafeca`
Ciphertext	`????????????`

Kasiski's Method (to find t)

- To identify repeated patterns of length 2 or 3
- The distance between such appearances is a multiple of t
- t is the greatest common divisor of all the distances

Example (Key is 'beads')

```
themanandthewomanretrievedtheletterfromthepostoffice
beadsbeadsbeadsbeadsbeadsbeansdeadsbeadsbeadsbeadbea
VMFQTPFOHMJJXSFCSSIMTNFZXFYISEIYUIKHWPQMJJQSLVTGJKGF
```

Index of Coincidence (IC) Method (to find t)

For $\tau = 1, 2, \ldots,$ q_i is the probability of ith letter in $c_1, c_{1+\tau}, c_{1+2\tau}, \ldots,$ IC is

$$I_\tau \stackrel{\text{def}}{=} \sum_{i=0}^{25} q_i^2$$

If $\tau = t$, then $I_\tau \approx ?$; otherwise $q_i \approx \frac{1}{26}$ and

$$I_\tau \approx \sum_{i=0}^{25} \left(\frac{1}{26}\right)^2 \approx 0.038$$

Then reuse IC method to find k_i.

Arbitrary Adversary Principle

Security must be guaranteed for any adversary within the class of adversaries having the specified power

Cryptanalytic Attacks (homework assignment)

- Under COA, the requirement for ciphertext related to the size of the key space. Vigènere $>$ mono-alphabetic sub. $>$ shift
- Under KPA, trivially broken.

Lessons learned

- Sufficient key space principle
- Designing secure cipher is a hard task
- Complexity does not imply security (then what does?)
- Arbitrary adversary principle

Content

Three Main Principles of Modern Cryptography

1. The formulation of a rigorous **definition** of security / threat model
2. When the security of a cipher relies on an unproven **assumption**, this assumption must be precisely stated and be as minimal as possible
3. Cipher should be accompanied by a rigorous **proof** of security with the above definition and the above assumption

Why Principle 1 – Formulation of Exact Definitions

Q: how would you formalize the security for private-key encryption?

1. *No adversary can find the secret key when given a ciphertext.*
   $\mathsf{Enc}_k(m) = m$
2. *No adversary can find the plaintext that corresponds to the ciphertext.*
   $\mathsf{Enc}_k(m) = m_0 \| \mathsf{AES}_k(m)$
3. *No adversary can determine any character of the plaintext that corresponds to the ciphertext.*
   $m = 1000$, someone can learn $800 < m < 1200$
4. *No adversary can derive any meaningful information about the plaintext from the ciphertext.*
   Could you define so-called 'meaningful'?

Definitions of security should suffice for all potential applications.

Why Principle 1 – How to define

How To Define Security – Lesson From Alan Turing

- What's computation?[3]
  1. A direct appeal to **intuition**
  2. A **proof of the equivalence** of two definitions (The new one has a greater intuitive appeal)
  3. Giving **examples** solved using a definition
- Additional method for security: **Test of time**

[3] Q: Any "mathematical proof that there exist well-defined problems that computers cannot solve"? A: Halting Problem in computability theory

Principle 2 – Reliance on Precise Assumptions

Most cryptographic constructions **cannot be proven secure unconditionally**

- **Why?**
  1. Validation of the assumption
  2. Comparison of schemes
  3. Facilitation of proofs of security

  The construction is secure if the assumption is true.
- **How?**
  1. old, so well tested
  2. simple and lower-level, so easy to study, refute & correct

Principle 3 – Rigorous Proofs of Security

- **Why?** Proofs are more desirable in computer security than in other fields.
- **The reductionist approach**:

Theorem 1

Given that Assumption X is true, Construction Y is secure according to the given definition.

Proof.

Reduce the problem given by X to the problem of breaking Y. □

- **Ad-hoc approaches**: for those who need a "quick and dirty" solution, or who are just simply unaware.

Summary

- Cryptography secures information, transactions and computations
- Kerckhoffs's principle & Open cryptographic design
- Caesar's, shift, Mono-Alphabetic sub., Vigenère
- Brute force, letter frequency, Kasiski's, IC
- Sufficient key space principle
- Arbitrary adversary principle
- Rigorously proven security

What is cryptography? [xkcd:504]

Alice, Bob [xkcd:1323]

Changing the names would be easier, but if you're not comfortable lying, try only making friends with people named Alice, Bob, Carol, etc.