Computer Security (2017)

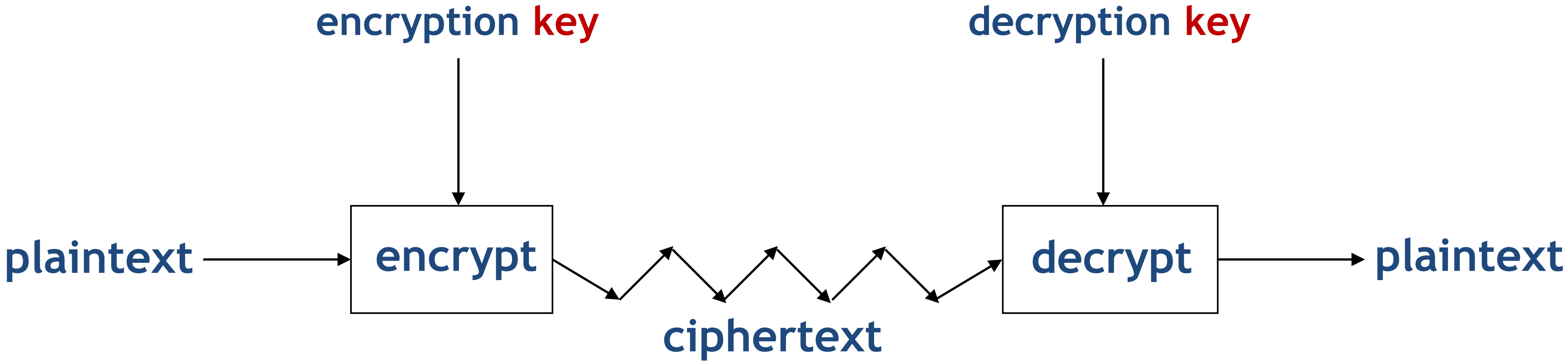
Classical Cryptography

Hyoungshick Kim
Department of Software
College of Software
Sungkyunkwan University

Cryptography

“secret” “writing”

Crypto as black box



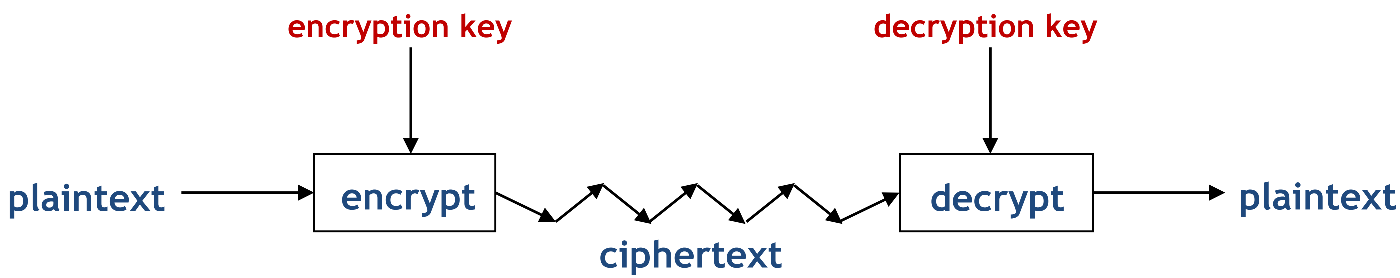
A generic view of key crypto

Tools for confidentiality and Integrity

Types of crypto systems

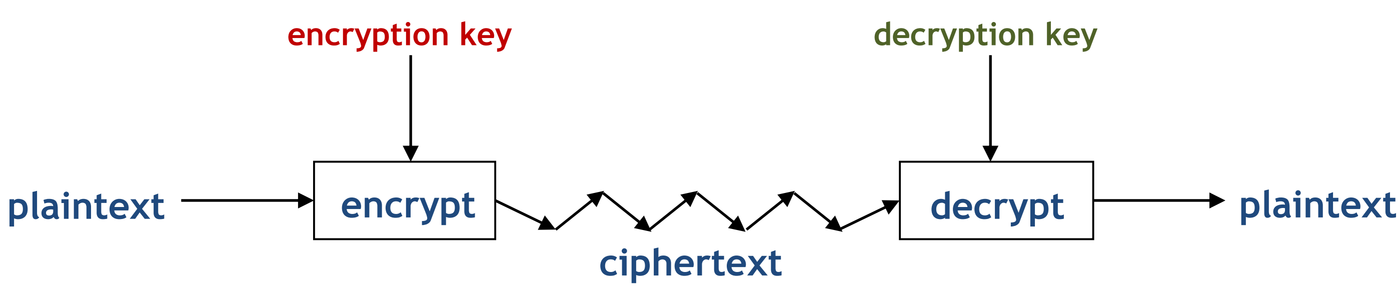
- Symmetric

(decryption key = encryption key)



- Public or Asymmetric

(decryption key != encryption key)



A framework for crypto

- Cryptography (making), cryptanalysis (breaking), cryptology (both)
- Traditional cryptanalysis – what goes wrong with the design of the algorithms
- Then – what goes wrong with their implementation (power analysis, timing attacks)
- Then – what goes wrong with their use

Things to remember

- Cryptography is:
  - a tremendous tool
  - the basis for many security mechanisms
- Cryptography is not:
  - the solution to all security problems
  - reliable unless implemented and used properly
  - something you should try to invent yourself
    - » too many examples of broken ad-hoc designs

"If you think cryptography is the answer to your problem, you don't know what your problem is."

Peter G. Neumann

The three steps in cryptography

1. Precisely specify a threat (or attack) model

2. Propose a construction

3. Prove that breaking construction under threat mode will solve an underlying hard problem (Security Proof)

Attack models

- Ciphertext only
- Known plaintext
- Chosen plaintext (CPA)
- Chosen ciphertext (CCA1, CCA2)
- Exhaustive key search (in honey encryption)

Ciphertext only attack

- Attacker has access to a set of ciphertexts
- Guess-and-check
- Frequency analysis

Known plaintext attack

- Attacker has the ciphertext and some samples of plaintext
- Apply the known plaintext to the ciphertext to help decryption
- Preferred over ciphertext-only

Chosen plaintext attack

- Chooses a plaintext and receives corresponding ciphertext
- Two variations
  - Batch Chosen-Plaintext
    - Attacker chooses a “batch” of plaintexts before any encrypted ciphertext is received
  - Adaptive chosen-plaintext
    - Attacker makes n-amounts of interactive queries and alters their plaintext based on the previous queries

Chosen ciphertext attack

- Tries to discover the key
- Uses a ciphertext chosen by attacker
- Relies on being able to obtain decrypted plaintext
- Two variations
  - Lunchtime attack (CCA1)
  - Adaptive chosen-ciphertext (CCA2)

Lunchtime attack (CCA1)

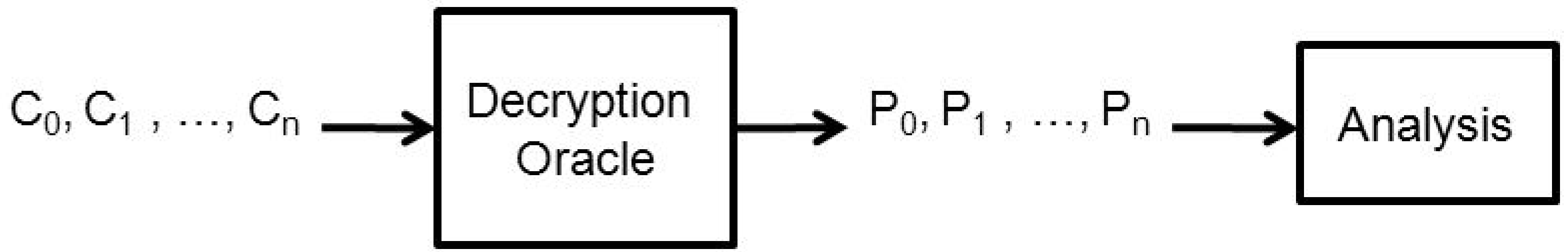
- Attacker can make queries but only up until a certain point
- Attacker cannot adapt queries
  - Results given after the ability to makes queries expires

Adaptive Chosen-Ciphertext (CCA2)

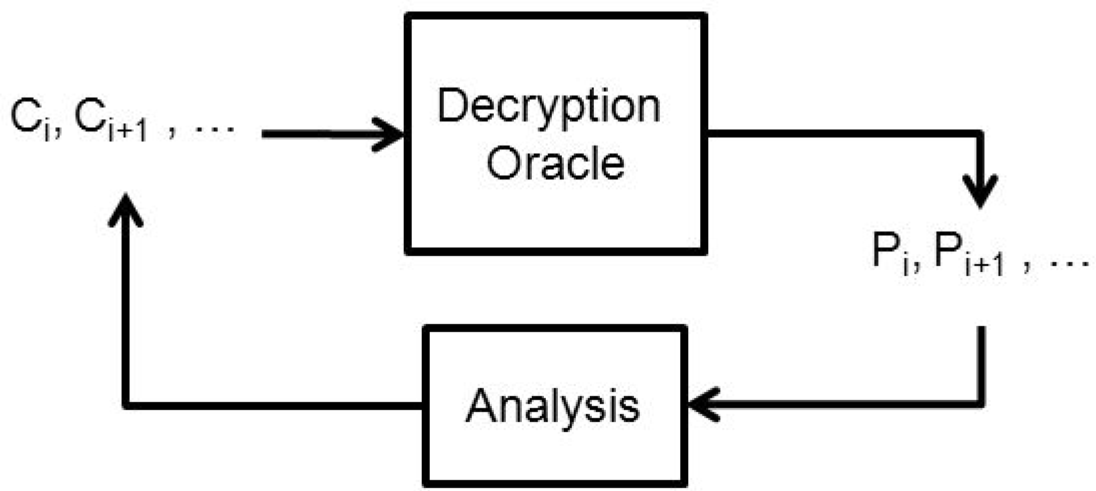
- Similar to normal chosen-ciphertext
- Ciphertext is chosen based on the result of the previous queries

CCA1 vs CCA2

- CCA1 (Lunchtime Attack)



- CCA2 (Adaptive Chosen Ciphertext Attack)



History

- Egyptians 4000 years ago
- World war 1&2

Enigma machine

- As a tool to protect national secrets and strategies

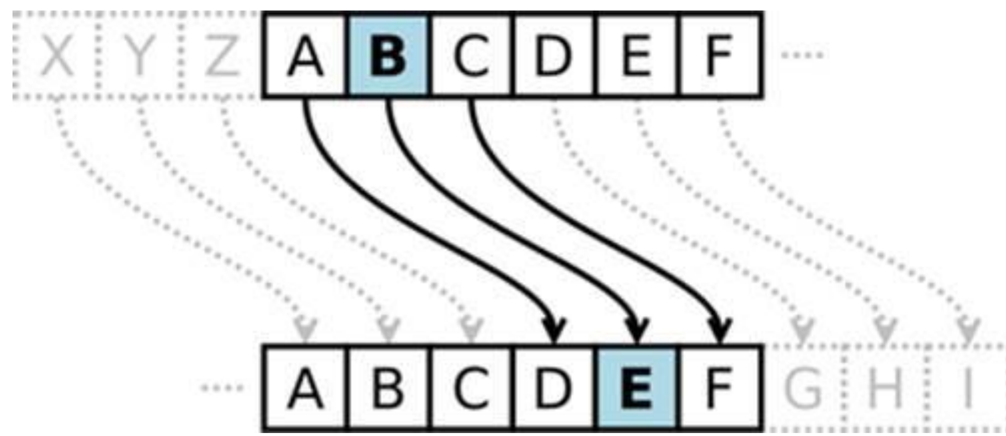
The Imitation game (movie)

https://www.youtube.com/watch?v=eYfCvBDVSQY

Ceasar cipher (no key)

Shift by 3

a → d
b → e
c → f
…
z → c



$$c = E_k(\text{“abbc”})$$

$$= \text{“deef”}$$

Basic assumptions

- The system is completely known to the attacker
- Only the key is secret
- That is, crypto algorithms are not secret
- This is known as **Kerckhoff's Principle**
- Why do we make this assumption?
  - Experience has shown that secret algorithms are weak when exposed
  - Secret algorithms never remain secret
  - Better to find weaknesses beforehand

Substitution cipher with shift

Shift by k

a → c
b → d
c → e
…
z → b

Key k

$c = E_k(\text{“abbc”}),\ k = 2$

$= \text{“cdde”}$

How to break this cipher? Try them all

- A simple substitution (shift by n) is used
  - But the key is unknown
- Given ciphertext: cdde
- How to find the key?
- Only 26 possible keys — try them all!
- **Exhaustive key search**
- Solution: key is $\mathrm{k} = 2$

Substitution cipher with permutation

Use any permutation of letters

a → e
b → c
c → q
…
z → a

Key permutation

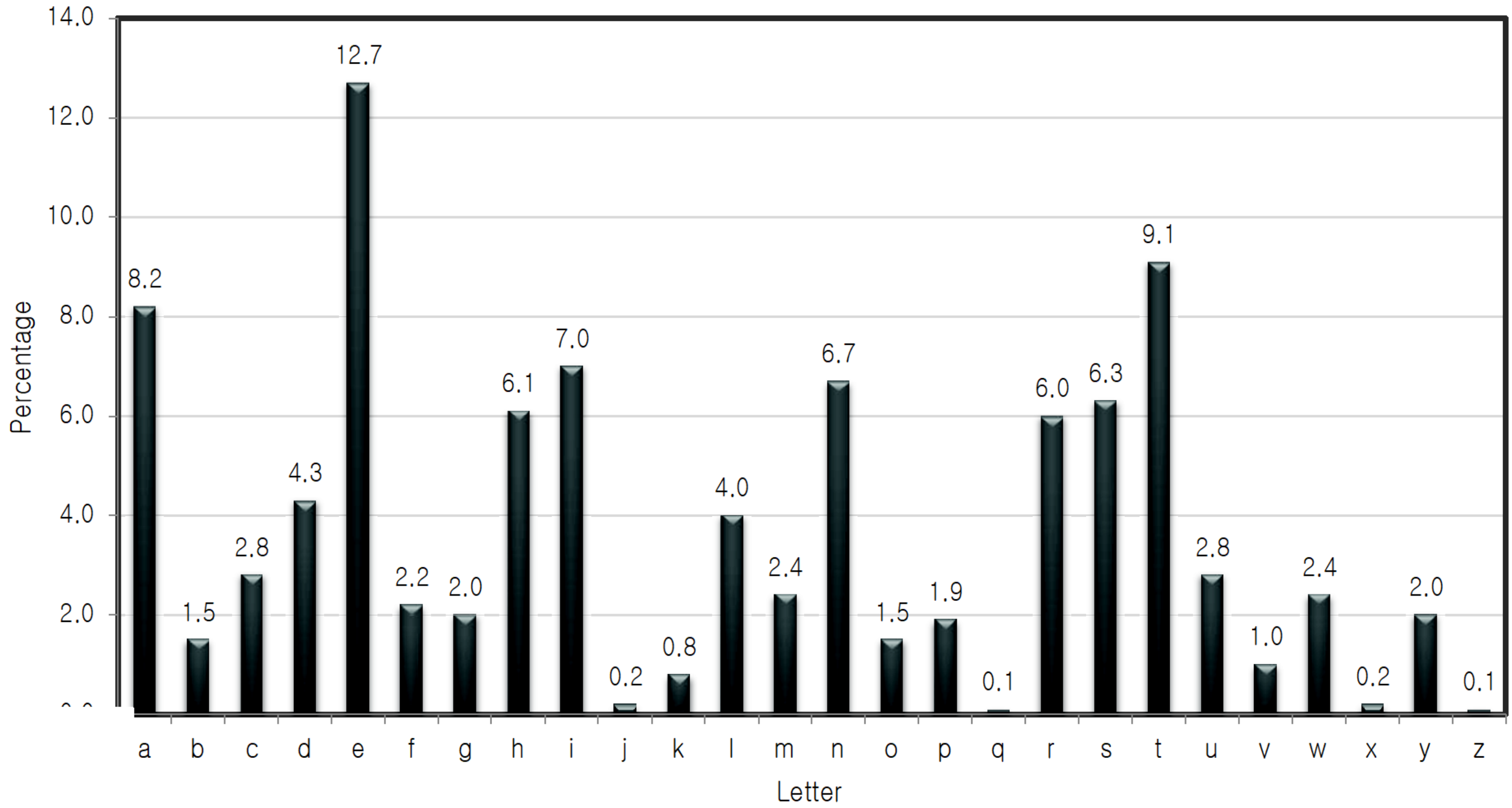
$$c = E_k(\text{“abbc”})$$

$$= \text{“eccq”}$$

Then $26! > 2^{88}$ possible keys!

How to break this cipher?

- Use letter frequencies; most common letters in English are e, t, a, I, o, n, s, h, r, d, l, u



An example of cryptanalysis

- Ciphertext:

PBFPVYFBQXZTYFPBFEQJHDXXQVAPTPQJKTOYQWIPBVWLXTOXBTFXQWAXBVCXQWAXFQJVWLE
QNTOZQGGQLFXQWAKVWLXQWAEBIPBFXFQVXGTVJVWLBTPQWAEBFPBFHCVLXBQUFEVWLXG
DPEQVPQGVPPBFTIXPFHXZHVFAGFOTHFEFBQUFTDHZBQPOTHXTYFTODXQHFTDPTOGHFQPBQ
WAQJJTODXQHFOQPWTBDHHIXQVAPBFZQHCFWPFHPBFIPBQWKFABVYYDZBOTHPBQPQJTQOT
OGHFQAPBFEQJHDXXQVAVXEBQPEFZBVFOJIWFFACFCCFHQWAUVWFLQHGFXVAFXQHFUFHILTT
AVWAFFAWTEVOITDHFHFQAITIXPFHXAFQHEFZQWGFLVWPTOFFA

Analyze this message using statistics below

Ciphertext frequency counts:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
21	26	6	10	12	51	10	25	10	9	3	10	0	1	15	28	42	0	0	27	4	24	22	28	6	8

This might be ‘e’.

Vigenère cipher

- 16th century – the Vigenère

plaintext	tobeornottobethatistheques ···
key	runrunrunrunrunrunrunrunru ···
ciphertext	KIOVIEEIGKIOVNURNVJNUVKHVM ···

- Use repeated patterns at multiples of key length (Kasiski, 1883) – here, 'KIOV'
- Size of key space?
  – If keys are 14-character strings; then key space has size $26^{14} \approx 2^{66}$
- Modern cryptanalysis (1915): Using index of coincidence to guess the key length

Variant Vigenère cipher

- Easier to work with ASCII plaintext and hex ciphertext
  - Easier to implement
  - Easier to use (plaintext not limited to lowercase characters)

- Easier to work with byte-wise XOR rather than modular addition

Variant Vigenère cipher

- The key is a string of bytes
- The plaintext is a string of ASCII characters
- To encrypt, XOR each character in the plaintext with the next character of the key
- Decryption just reverses the process

Example

- Say plaintext is “Hello!” and key is 0xA1 2F
- “Hello!” = 0x48 65 6C 6C 6F 21
- XOR with 0xA1 2F A1 2F A1 2F
- 0x48 $\oplus$ 0xA1
  - 0100 1000 $\oplus$ 1010 0001 = 1110 1001 = 0xE9

- Ciphertext: 0xE9 4A CD 43 CE 0E

Attacking the Vigenère cipher

- Two steps:
  - Determine the key length
  - Determine each character of the key

Determining the key length

- Let p_i (for $0 \leq i \leq 255$) be the frequency of byte i in plaintext (assuming English text)
  - I.e., p_i =0 for i<0 or i>127
  - I.e., p_{97} = frequency of 'a'
  - The distribution is far from uniform
- If the key length is N, then every N^{th} character of the plaintext is encrypted using the same "shift"
  - If we take every N^{th} character and calculate frequencies, we should get the p_i's in permuted order
  - If we take every M^{th} character (M not a multiple of N) and calculate frequencies, we should get something close to uniform

Determining the key length

- How to distinguish these two?
- For some candidate distribution $q_0, \ldots, q_{255}$, compute $\Sigma\, q_i^2$
  - If close to uniform, $\Sigma\, q_i^2 \approx 256 \cdot (1/256)^2 = 1/256$
  - If a permutation of p_i, then $\Sigma\, q_i^2 \approx \Sigma\, p_i^2$
    - Could compute $\Sigma\, p_i^2$ (but somewhat difficult)
    - Key point: will be much larger than 1/256
- Try all possibilities for the key length, compute $\Sigma\, q_i^2$, and look for maximum value

Index of Coincidence - Plaintext

Letter	a	b	c	d	e	f	g	h	i	j	k	l	m
Frequency	.082	.015	.028	.043	.127	.022	.020	.061	.070	.002	.008	.040	.024
Letter	n	o	p	q	r	s	t	u	v	w	x	y	z
Frequency	.067	.075	.019	.001	.060	.063	.091	.028	.010	.023	.001	.020	.001

Beker and Piper, *Cipher Systems: The Protection of Communications*, Wiley.

$$\begin{array}{ccccccccc} \text{aa} & \text{or} & \text{bb} & \text{or} & \text{cc} & \text{or} & \ldots & \text{or} & \text{zz} \\ .082\times.082 & + & .015\times.015 & + & .028\times.028 & + & \ldots & + & .001\times.001 \end{array}$$

$$I \approx 0.0656010$$

Index of Coincidence - Uniform

$$I \approx \underbrace{\left(\frac{1}{26}\times\frac{1}{26}\right)+\left(\frac{1}{26}\times\frac{1}{26}\right)+\left(\frac{1}{26}\times\frac{1}{26}\right)+\ldots+\left(\frac{1}{26}\times\frac{1}{26}\right)}_{\text{26 terms}}=\frac{1}{26}\approx 0.038$$

Determining the i^{th} byte of the key

- Assume the key length N is known
- Look at every N^{th} character of the ciphertext, starting with the i^{th} character
  - Call this the i^{th} ciphertext "stream"
  - Note that all bytes in this stream were generated by XORing plaintext with the same byte of the key
- Try decrypting the stream using every possible byte value B
  - Get a candidate plaintext stream for each value

Determining the i^{th} byte of the key

- When the guess B is correct:
  - All bytes in the plaintext stream will be between 0 and 127
  - Frequencies of lowercase letters (as a fraction of all lowercase letters) should be close to known English-letter frequencies
    - Tabulate $q_a, \ldots, q_z$
    - Should find $\Sigma\, q_i\, p_i \approx \Sigma\, p_i^2 \approx 0.065$
    - In practice, take B that maximizes $\Sigma\, q_i\, p_i$, subject to caveat above (and possibly others)

Attack time?

- The key length is between 1 and L
- Determining the key length: $\approx 256\, L$
- Determining all bytes of the key:
  - Guessing B at ith character: 256
  - Calculating $\Sigma\, q_i\, p_i$ at ith character: 256
  - Total: $256^2\, L$

- Brute-force key search: $\approx 256^L$

The attack in practice

- Attacks get more reliable as the ciphertext length grows larger

- Attacks still work for short(er) ciphertexts, but more “tweaking” and manual involvement is needed

One Time Pad (OTP)

First example of a "secure" cipher

$\mathcal{M} = \mathcal{C} = \{0, 1\}^n$

Choose key k as random bit string as long the message!

$\mathcal{K} = \{0, 1\}^n$

$$\boxed{\text{Encryption: } c = k \oplus m}$$

Very fast enc/dec !!

… but long keys (as long as plaintext)

Quiz

You are given a message (m) and its OTP encryption (c).

Q. Can you compute the OTP key from m and c ?

Yes, the key is $k = m \oplus c$.

OTP has perfect secrecy

Q. What is the perfect secrecy?

Information Theoretic Security

(Shannon 1949)

Def: A cipher $(\boldsymbol{E}, \boldsymbol{D})$ over $(\mathcal{K}, \mathcal{M}, \mathcal{C})$ has **perfect secrecy** if

$\forall m_0, m_1 \in \mathcal{M} \quad (|m_0| = |m_1|)$ and $\forall c \in \mathcal{C}$

$$Pr[\, \boldsymbol{E}(\boldsymbol{k}, \boldsymbol{m_0}) = \boldsymbol{c} \,] = Pr[\, \boldsymbol{E}(\boldsymbol{k}, \boldsymbol{m_1}) = \boldsymbol{c} \,] \quad \text{where} \quad k \overset{R}{\leftarrow} \mathcal{K}$$

- Given *c*, we can't tell if *m* is m_0 or m_1
- Any adversary can't learn any information about *m* from *c*
- In other words, *m* and *c* are independent
- No ciphertext only attack!! (but other attacks might be possible)

Claude Shannon

Claude Shannon

Scholar

About 60,400 results (**0.03** sec)

Articles

Case law

My library

A mathematical theory of communication

CE **Shannon** - ACM SIGMOBILE Mobile Computing and ..., 2001 - dl.acm.org

T HE recent development of various methods of modulation such as PCM and PPM which exchange band-width for signal-to-noise ratio has intensified the interest in a general theory of communication. A basis for such a theory is contained in the important papers of Nyquist 1

Cited by 93703 Related articles All 677 versions Cite Saved

OTP has perfect secrecy
(Shannon 1949)

Proof:

$\forall m, c$ **:**

$$\Pr_k[E(k, m) = c] = |\{ k \in \mathcal{K} : E(k, m) = c \}| \,/\, |\mathcal{K}|$$

$$= 1 \,/\, |\mathcal{K}|$$

Unfortunately …

(Shannon's theorem)

<u>Thm</u>: If a shared-key encryption has perfect secrecy

$\Rightarrow \quad |\mathcal{K}| \geq |\mathcal{M}|$

- That is, key length should be greater than message length to achieve perfect secrecy.
- It is hard to use in practice!!!

Using the same key twice?

- $c_1 = k \oplus m_1$

  $c_2 = k \oplus m_2$

- Attacker can compute

  $c_1 \oplus c_2 = (k \oplus m_1) \oplus (k \oplus m_2) = m_1 \oplus m_2$

- This leaks information about m_1, m_2!
  - No longer perfectly secret! (e.g., $m_1 \oplus m_2$ reveals where m1, m2 differ)
  - Frequency analysis

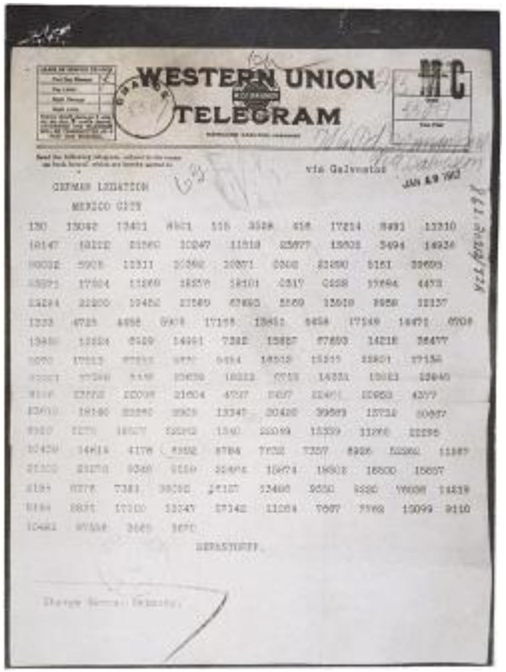
OTP in practice: The Moscow-Washington hotline

- A device called *Electronic Teleprinter Cryptographic Regenerative Repeater Mixer II* (ETCRRM II) encrypted the teletype messages.
- ETCRRM II used OTP.
- Each country delivered keying tapes used to encode its messages via its **embassy** abroad.

Codebook Cipher

- Literally, a book filled with “codewords”
- Zimmerman Telegram encrypted via codebook

Februar	13605
fest	13732
finanzielle	13850
folgender	13918
Frieden	17142
Friedenschluss	17149
:	:



- Modern block ciphers are codebooks!

One more thing!

- Fundamental concepts
  - **Confusion** —— obscure relationship between plaintext and ciphertext
  - **Diffusion** —— spread plaintext statistics through the ciphertext
- One-time pad is confusion-only

Double transposition

Plaintext: attackxatxdawn

	col 1	col 2	col 3
row 1	a	t	t
row 2	a	c	k
row 3	x	a	t
row 4	x	d	a
row 5	w	n	x

Permute rows and columns

	col 1	col 3	col 2
row 3	x	t	a
row 5	w	x	n
row 1	a	t	t
row 4	x	a	d
row 2	a	k	c

- Ciphertext: xtawxnattxadakc
- What is the key?
- Key is matrix size and permutations: (3,5,1,4,2) and (1,3,2)
- Double transposition is diffusion-only

Modern Cipher Systems

- Many systems from the last century use stream ciphers for speed / low gate count
- Bank systems use a 1970s block cipher, the data encryption standard or DES; recently moving to triple-DES for longer keys
- New systems mostly use the Advanced Encryption Standard (AES), regardless of whether a block cipher or stream cipher is needed

Questions?

Sungkyunkwan
Campus
Sciences
established
University
university
Suwon
School
Seoul
Natural
Graduate
campus
English
Korean
Social
Global
edit
constructed
Professors
Confucian
Humanities
lecturers
Building
Institute
Education
Business
schools
Science
Administration
Academic
ASLE-Korea
Advanced
Engineering
located
Joseon
Anniversary
University's
establishment
now
two
four
also
city
one
heart
trees