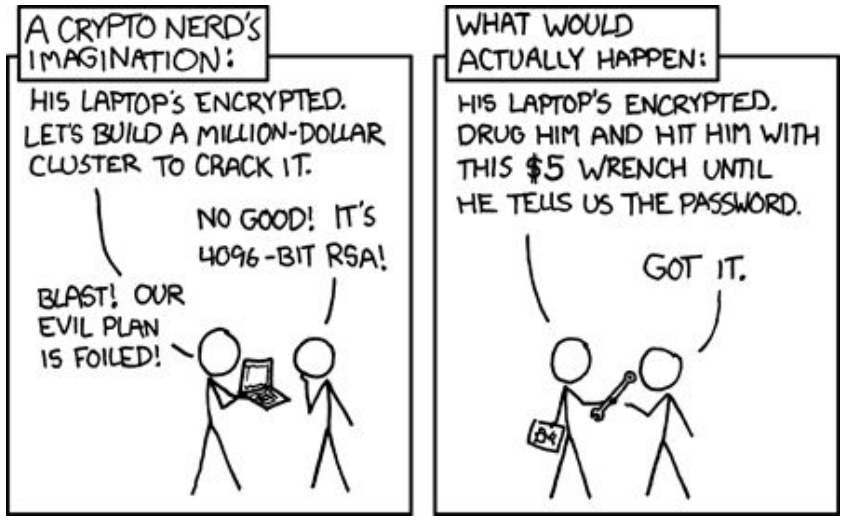
Διάλεξη #11 - Introduction to Cryptography

Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών

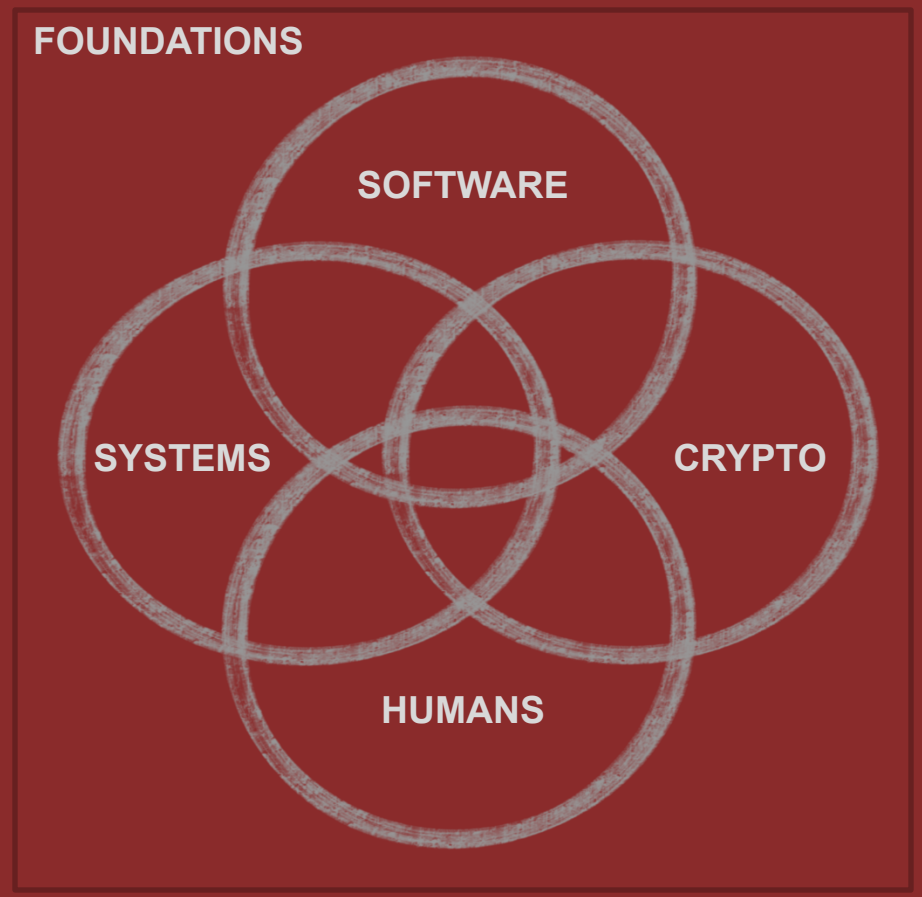
Εισαγωγή στην Ασφάλεια

Θανάσης Αυγερινός



Hfr frpher pelcgb cyrnfr!

Huge thank you to David Brumley from Carnegie Mellon University for the guidance and content input while developing this class (some slides from Dan Boneh @ Stanford!)



Ανακοινώσεις / Διευκρινίσεις

- Μόλις έκλεισε η εργασία #1 - η εργασία #2 θα βγει (λογικά*) μέχρι το τέλος της εβδομάδας

Ερωτήσεις:

- Στο Linux έχουμε τελικά ACLs;
- Γιατί χρησιμοποιούμε setuid binaries;

Την Προηγούμενη Φορά

- Reference Monitors
- "Gold" (Au) Standard
- Authorization Mechanisms / Access Control
  - Access Control Lists (ACLs) and Capabilities (CAP)
  - Discretionary Access Control (DAC)
  - Role-Based Access Control (RBAC)

Security in the News

Breach of the Week

Change Healthcare Finally Admits It Paid Ransomware Hackers—and Still Faces a Patient Data Leak

For Change Healthcare and the beleaguered medical practices, hospitals, and patients that depend on it, the confirmation of its extortion payment to the hackers adds a bitter coda to an already dystopian story. AlphV's digital paralysis of Change Healthcare, a subsidiary of UnitedHealth Group, snarled the insurance approval of prescriptions and medical procedures for hundreds of medical practices and hospitals across the country, making it by some measures the most widespread medical ransomware disruption ever. A survey of American Medical Association members conducted between March 26 and April 3 found that four out of five clinicians had lost revenue as a result of the crisis. Many said they were using their own personal finances to cover a practice's expenses. Change Healthcare, meanwhile, says it has lost $872 million to the incident and projects that number to rise well over a billion in the longer term.

Change Healthcare's confirmation of its ransom payment now appears to show that much of that catastrophic fallout for the US health care system unfolded *after* it had already paid the hackers an exorbitant sum—a payment in exchange for a decryption key for the systems the hackers had encrypted and a promise not to leak the company's stolen data. As is often the case in ransomware attacks, AlphV's disruption of its systems appears to have been so widespread that Change Healthcare's recovery process has extended long after it obtained the decryption key designed to unlock its systems.

As ransomware payments go, $22 million wouldn't be the most that a victim has forked over. But it's close, says Brett Callow, a ransomware-focused security researcher who spoke to WIRED about the suspected payment in March. Only a few rare payments,

https://www.wired.com/story/change-healthcare-admits-it-paid-ransomware-hackers/

Σήμερα

- About cryptography
- Terminology
- Traditional ciphers
- One-time pad

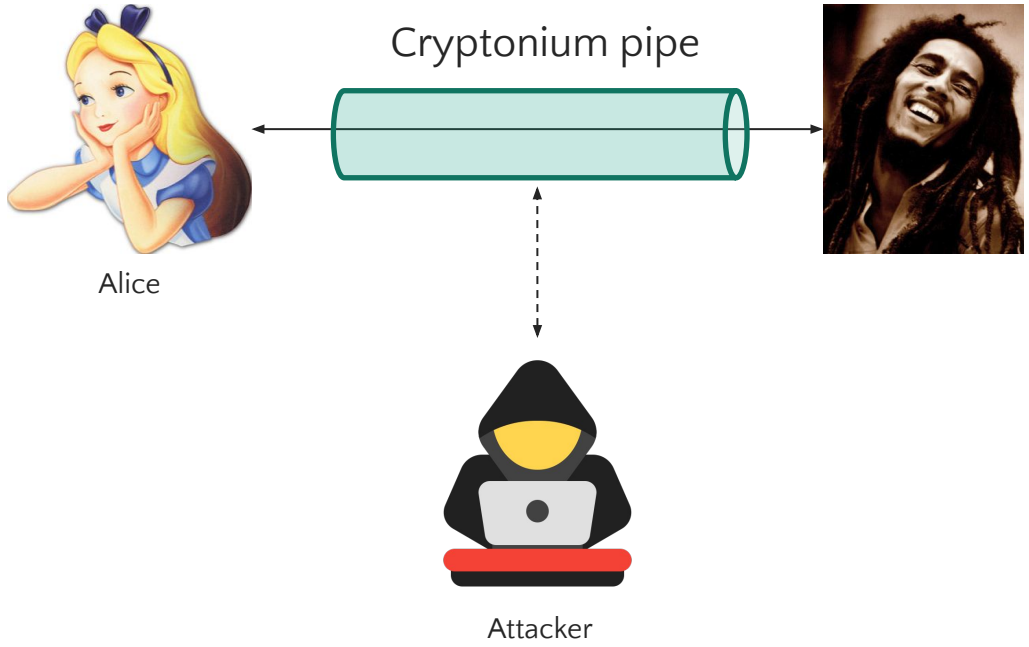
About Cryptography (aka crypto)

Cryptography

Cryptography, or **cryptology** (from Ancient Greek: κρυπτός, romanized: *kryptós* "hidden, secret"; and γράφειν *graphein*, "to write", or -λογία *-logia*, "study", respectively[1]), is the practice and study of techniques for secure communication in the presence of adversarial behavior.[2] More generally, cryptography is about constructing and analyzing protocols that prevent third parties or the public from reading private messages.[3]

https://en.wikipedia.org/wiki/Cryptography

Cryptography is About Secure Communication



Do we remember the 4 security properties?

Confidentiality (Secrecy), Integrity, Authenticity and Availability

Computer Security ≠ Crypto

"*Those who think that cryptography can solve their problems don't understand cryptography and don't understand their problems.*" – Bruce Schneier

- How can we generate good keys?
- How do we know the crypto implementations are correct?
- How do we build networks that are secure ***and*** available?
- How do we ensure only Alice can access her keys?
- …

Cryptography is everywhere…

- Secure communication:
  - HTTPS, 802.11i WPA2 (and WEP), SSL, GSM, Bluetooth
- Encrypting data at rest:
  - BitLocker (Windows), FileVault (MacOS)
- Content protection:
  - CSS (DVD), AACS (Blue-Ray)
- User authentication
  - Kerberos, HTTP Digest
- Crypto currencies:
  - Bitcoin, Ethereum, etc.
- Spectrum security
  - Frequency Hopping Spread Spectrum

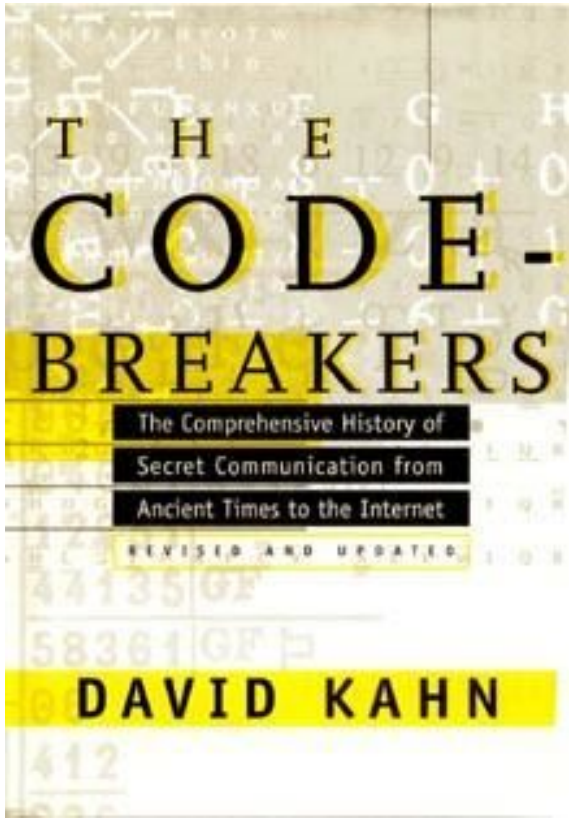
… and much, much more

…but how did it get there?

David Kahn,

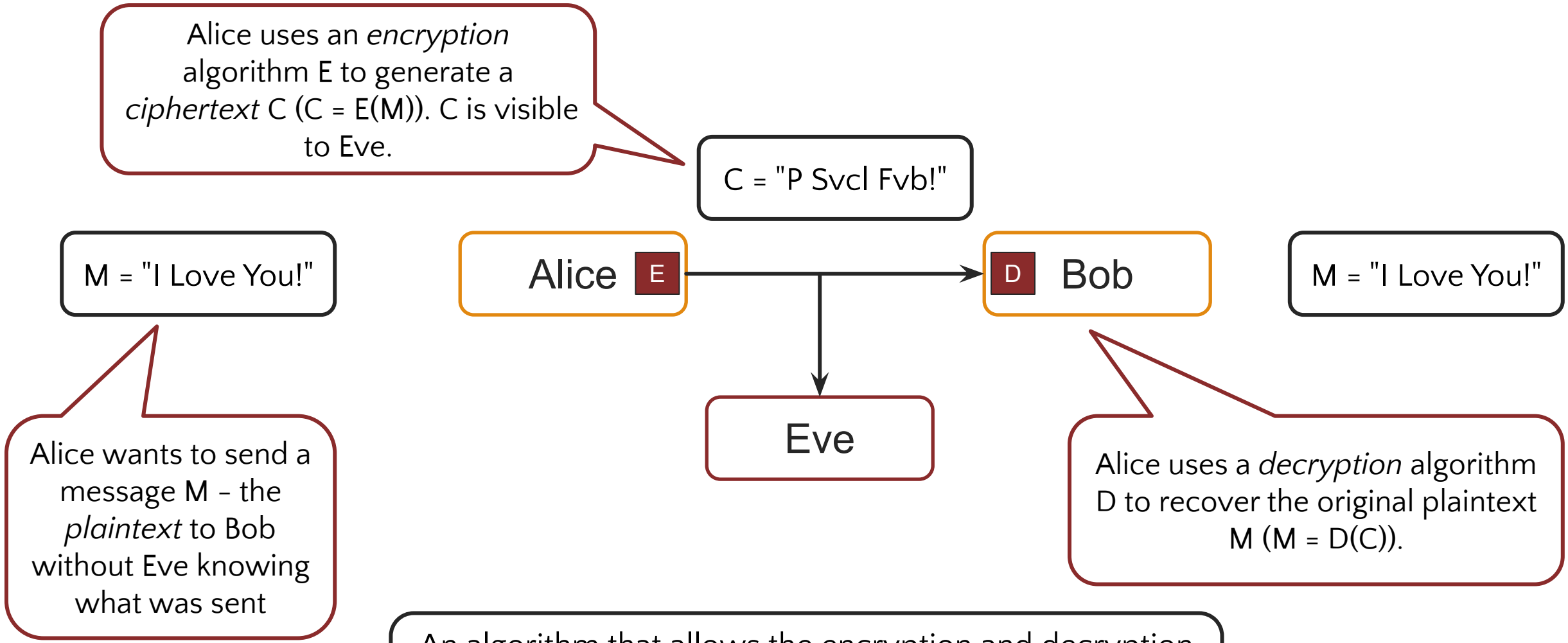
"The Code-Breakers"

(Revised ed. 1996)



It started with Secrets

The Secrecy Game



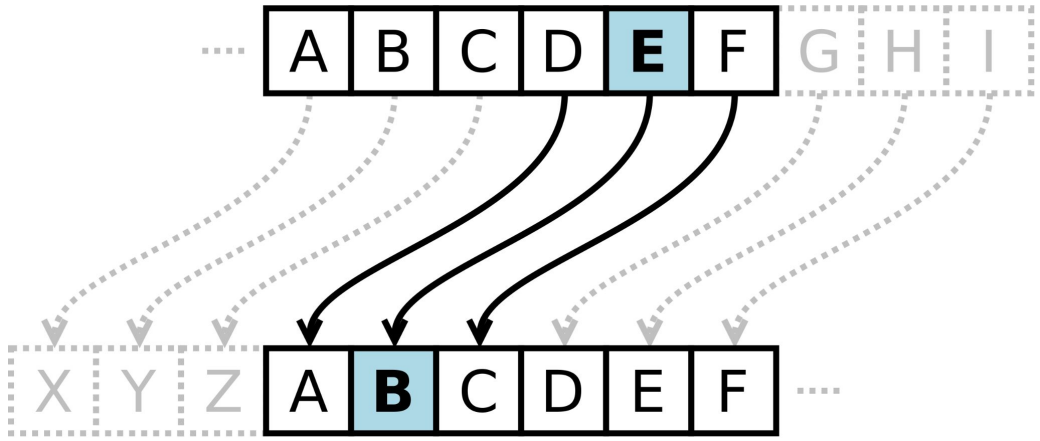
An algorithm that allows the encryption and decryption (E, D) of data is called a *cipher*. Typically requires a secret in order to function, also known as the *key* K.

Caesar Cipher:
replace each letter with letter + 3 (e.g., A →D)

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



How large is the key space?

How would you *attack* it?

Popular variant:
Rot-13

Julius Caesar
100 BC- 44 BC

Substitution Cipher

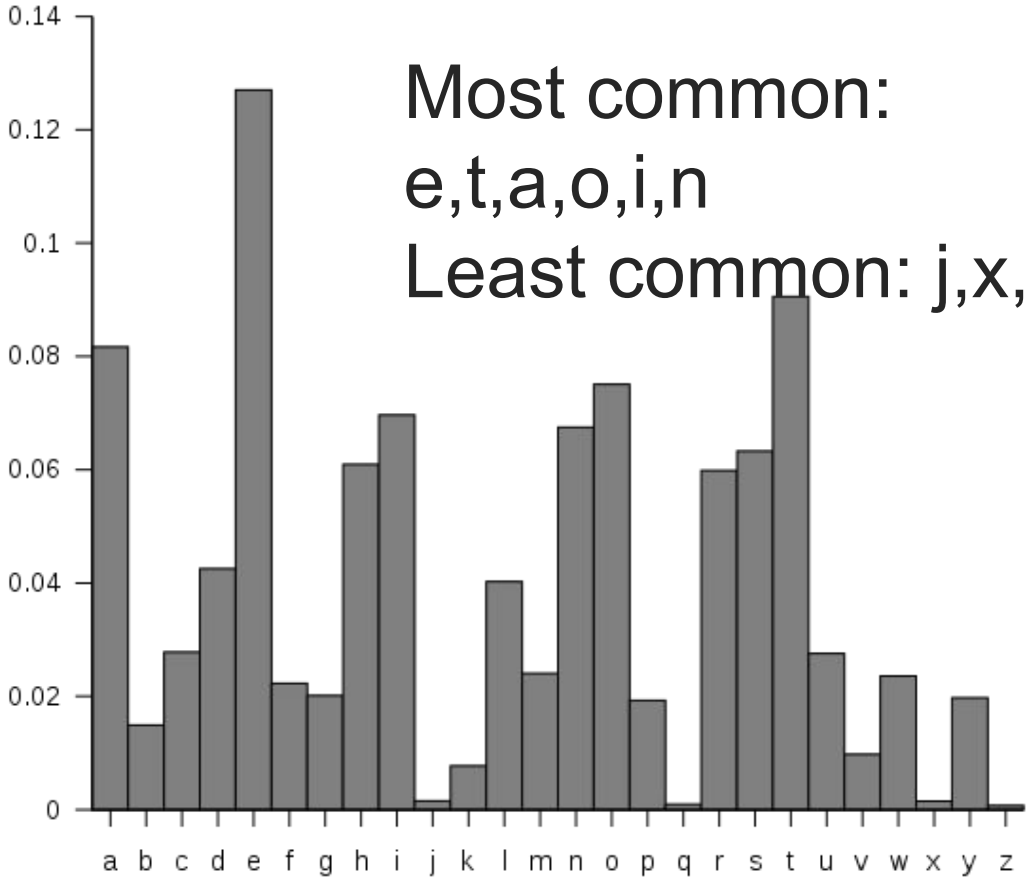
- Pick a random permutation from [A-Z] -> [A-Z]
- How large is the key space?
- How do you remember the permutation?
- Example: Use a secret key word/phrase
  - Map: ABCDEFGHIJKLMNOPQRSTUVWXYZ
    to: MONKEYSABCDFGHIJLPQRTUVWXY
- "State of the art" for thousands of years
- How large is this key space?

How would ***you*** decrypt messages encrypted with a substitution cipher?

Attacking Substitution Ciphers

Trick 1: *Word Frequency*

Trick 2: *Letter Frequency*

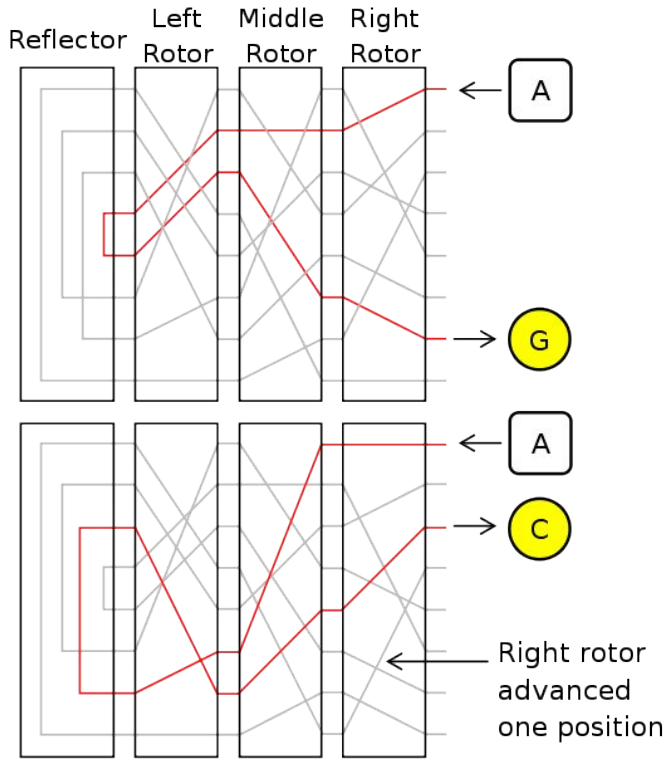


Most common:
e,t,a,o,i,n
Least common: j,x,q,z

image source: wikipedia

Rotor Machines

Most famous: the Enigma (3-5 rotors)



Used a substitution cipher along with other complexities on top.

Employed by Nazi Germany during World War II.

https://picoctf.com

Jvl mlwclk yr jvl owmwez twp yusl w zyduo pjdcluj mqil zydkplmr. Hdj jvlz tykilc vwkc jy mlwku jvl wkj yr vwsiquo, tvqsv vlmflc mlwc jvlg jy oklwjulpp. Zyd vwnl jvl fyjlujqwm jy cy jvl pwgl. Zydk plsklj fwpptykc qp: JYWPJ

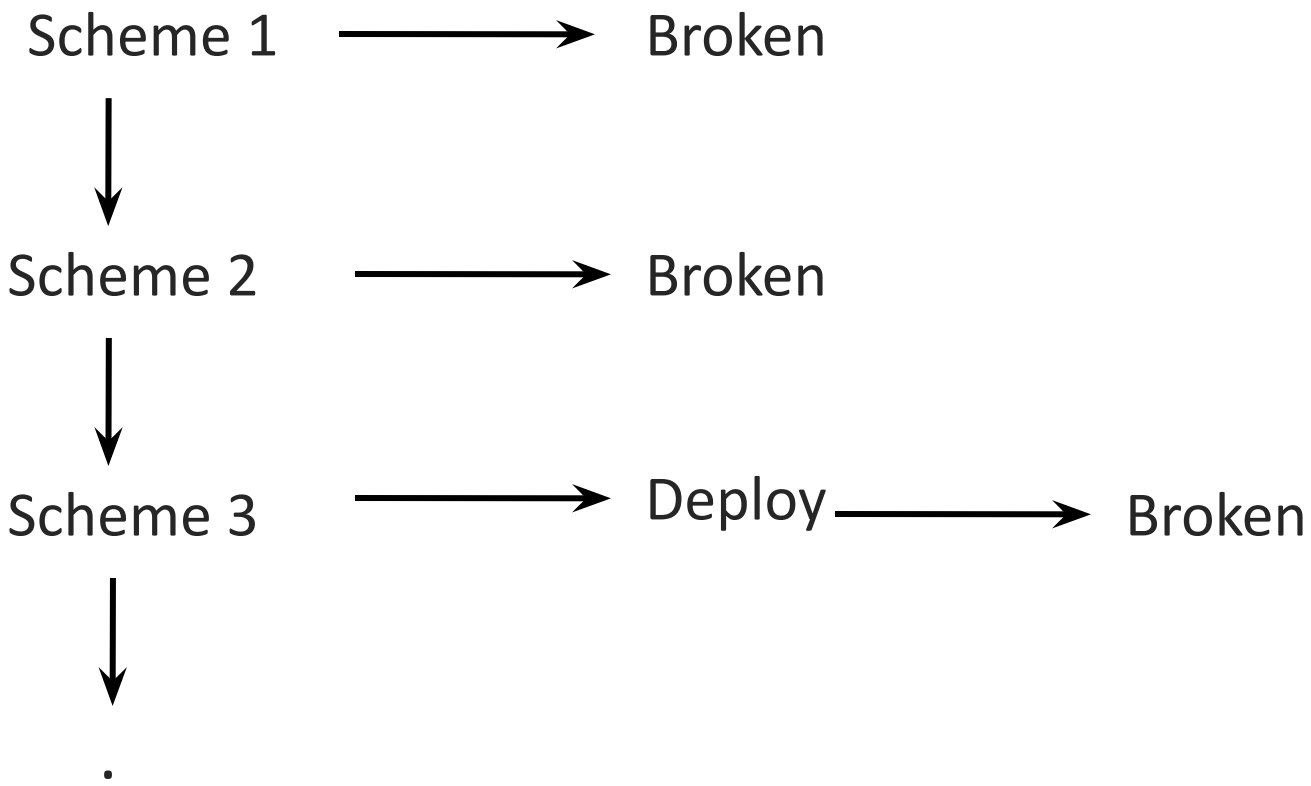
https://cryptopals.com

Kerckhoffs' Principle

A cryptosystem should be secure, even if *everything* about the system, *except* the key, is public knowledge.

Does this remind us of a standard security principle?

Classical Approach: Iterated Design



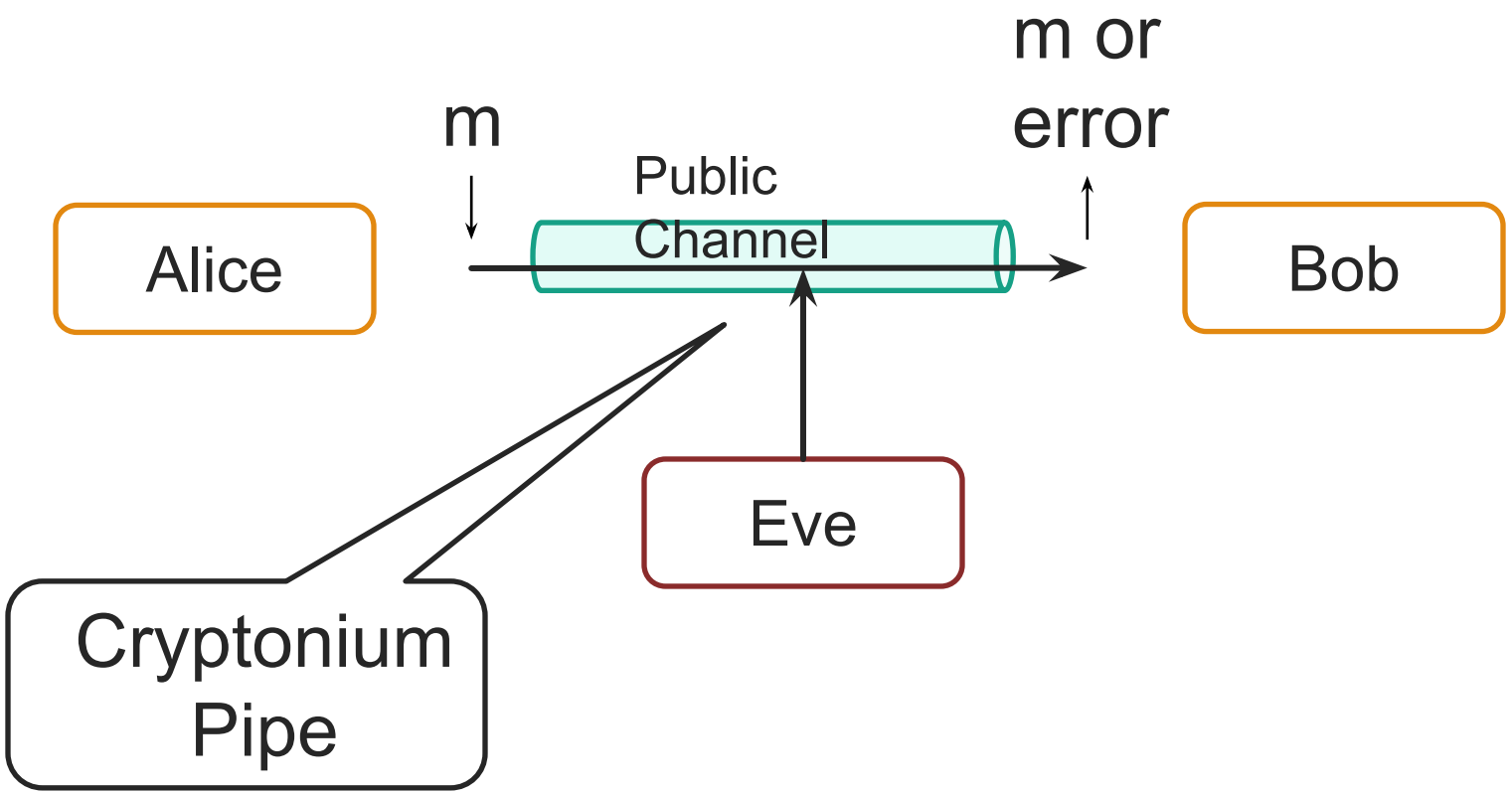
No way to say anything is secure
(and you may not know when broken)

Iterated design was all we knew until 1945

Claude Shannon: 1916 - 2001

- Formally define:
  - *security goals*
  - *adversarial models*
  - *security of system w.r.t. goals*
- Beyond iterated design: Proof!

Our Goal: Secure Channel



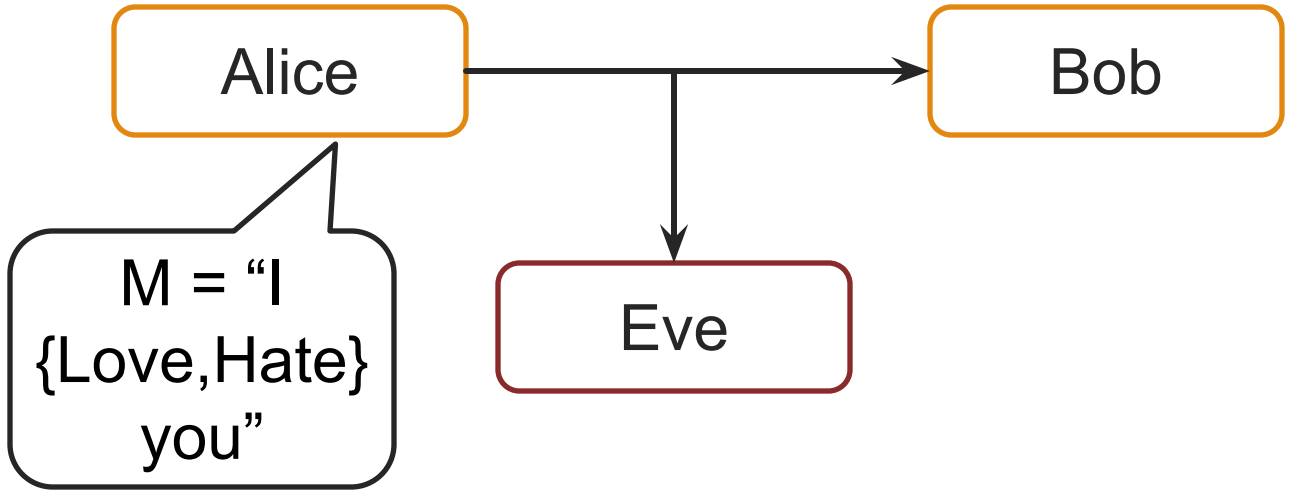
Sub Goal 1: *Secrecy*

Eve should not be able to learn m

How Secure Is Secure Enough?

Suppose there are two possible messages that differ on one bit, e.g., whether Alice Loves or Hates Bob

Secrecy means Eve still should not be able to determine which message was sent



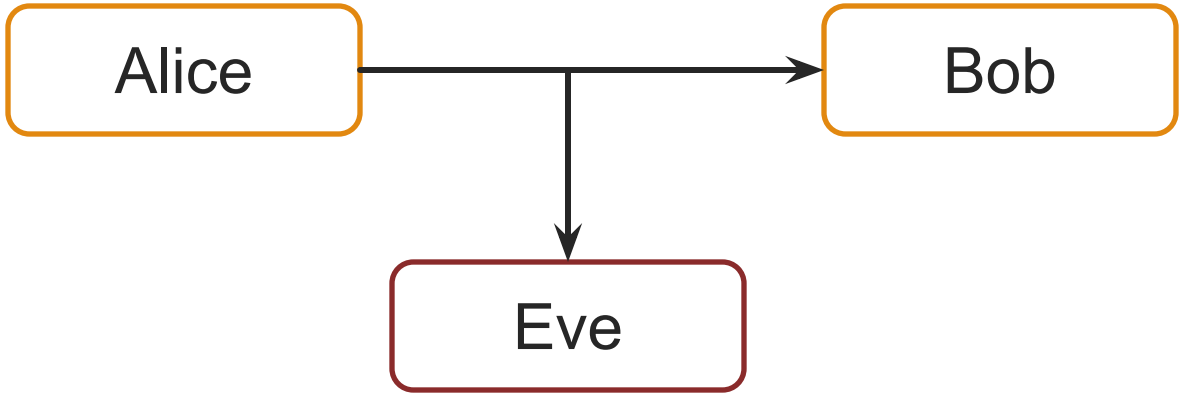
Security guarantees should hold for all messages, not just a particular kind of message

Secure Against Whom (Adversary)?

Eve's Possible Powers

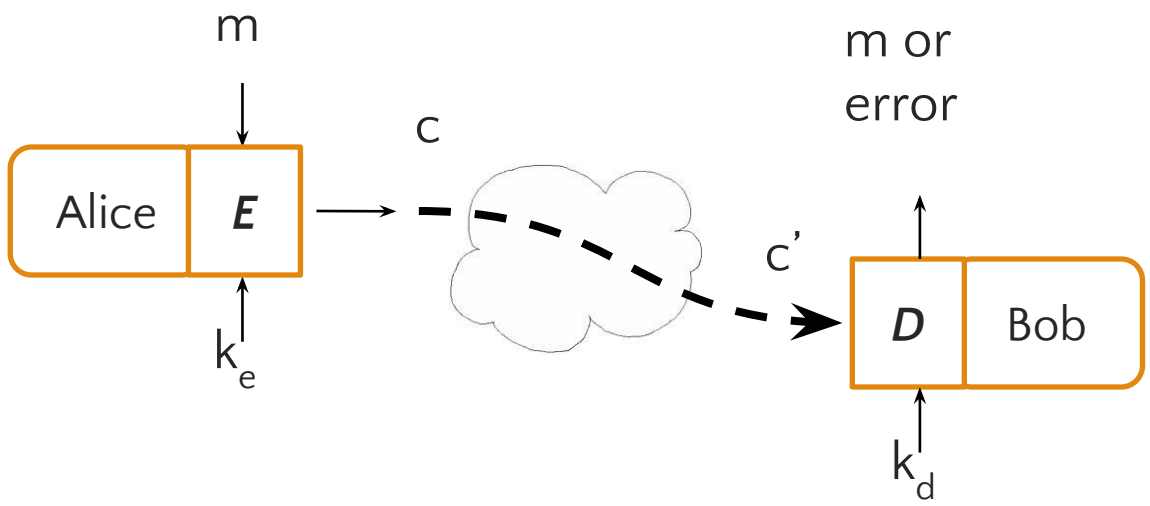
- *Ciphertext only*: only access to ciphertext
- *Known Plaintext Attack (KPA)*: Access to a <message,ciphertext> list
- *Chosen Plaintext Attack (CPA)*: Ability to have messages encrypted
- *Chosen Ciphertext Attack (CCA):* Ability to have ciphertexts decrypted

Note: Eve succeeds only if she gains information on a "fresh" ciphertext

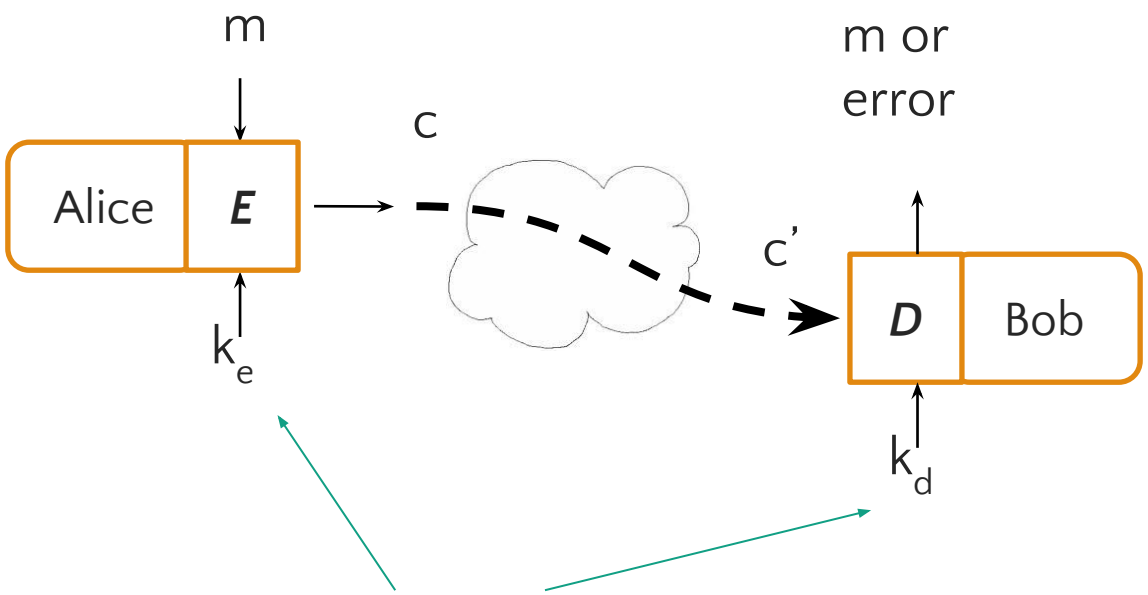


Generic Encryption Scheme

Var	Description
m	Message (aka plaintext). From the message space M
c	Ciphertext. From the ciphertext space C
E	Encryption Algorithm
D	Decryption Algorithm
$\mathbf{k_e}$	Encryption key from the key space K
$\mathbf{k_d}$	Decryption key from the key space K

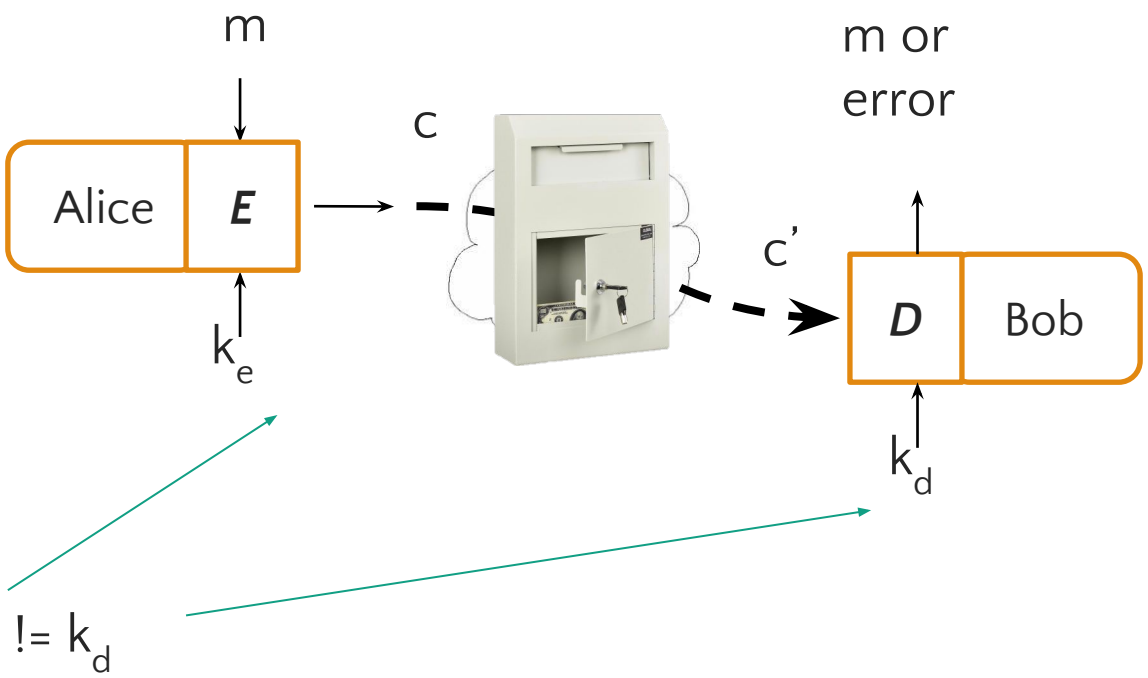


Symmetric Encryption



- $k = k_e = k_d$
- Everyone who knows k knows the full secret

Asymmetric Encryption



- k_e != k_d
- Encryption Example:
  - Bob generates private (k_d)/public(k_e) keypair
    - Sends Alice public key
  - To encrypt a message to Bob, Alice computes c = E(m,k_e)
  - To decrypt, Bob computes m = D(c', k_d)

An Interesting Story...

1974

- A student enrolls in the Computer Security course @ Stanford
- Proposes idea for public key crypto
- Professor shoots it down

Picture: http://www.merkle.com

1975

- Submits a paper to the Communications of the ACM

- “I am sorry to have to inform you that the paper is not in the main stream of present cryptography thinking and I would not recommend that it be published in the Communications of the ACM. Experience shows that it is extremely dangerous to transmit key information in the clear."

Picture: http://www.merkle.com

Today

Ralph Merkle:
A Pioneer of Cryptography

Picture: http://www.merkle.com

Ciphers and the one-time pad

Symmetric Cipher

Def'n: A **symmetric cipher** over key space $\mathcal{K}$, message space $\mathcal{M}$, and ciphertext space $\mathcal{C}$ comprises three polynomial time algorithms:

1. *KeyGen*($\lambda \in \mathbb{N}$) $\rightarrow$ k $\in \mathcal{K}$
   A randomized algorithm that returns a fresh key of length λ. We say that λ is the *security parameter.*

2. *E*(k $\in \mathcal{K}$, m $\in \mathcal{M}$) $\rightarrow$ c $\in \mathcal{C}$
   A (usually randomized) algorithm that encrypts *m* under *k* to produce ciphertext *c.*

3. *D*(k $\in \mathcal{K}$, c $\in \mathcal{C}$) $\rightarrow$ m $\in \mathcal{M}$ U ERROR
   A *deterministic* algorithm that decrypts c with key k, returning either a message *m* or an ERROR indicating decryption failure.

We say that a cipher is ***correct*** if it satisfies the following condition:

$$\forall\ k \in \mathcal{K} : \forall\ m \in \mathcal{M} : D(k, E(k, m)) = m$$

The One-Time Pad (OTP)

Miller, 1882 and Vernam, 1917

$$E(k, m) = k \oplus m = c$$

$$D(k, c) = k \oplus c = m$$

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

	m:	0	1	1	0	1	1	0
$\oplus$	k:	1	1	0	1	0	0	0
	c:	1	0	1	1	1	1	0
$\oplus$	k:	1	1	0	1	0	0	0
	m:	0	1	1	0	1	1	0

The One-Time Pad (OTP)

Miller, 1882 and Vernam, 1917

$$E(k, m) = k \oplus m = c$$

$$D(k, c) = k \oplus c = m$$

Is it a cipher?

✔ Efficient

✔ Correct

$$\begin{aligned} D(k, E(k, m)) &= D(k, k \oplus m) \\ &= k \oplus (k \oplus m) \\ &= 0 \oplus m \\ &= m \end{aligned}$$

Participation Question

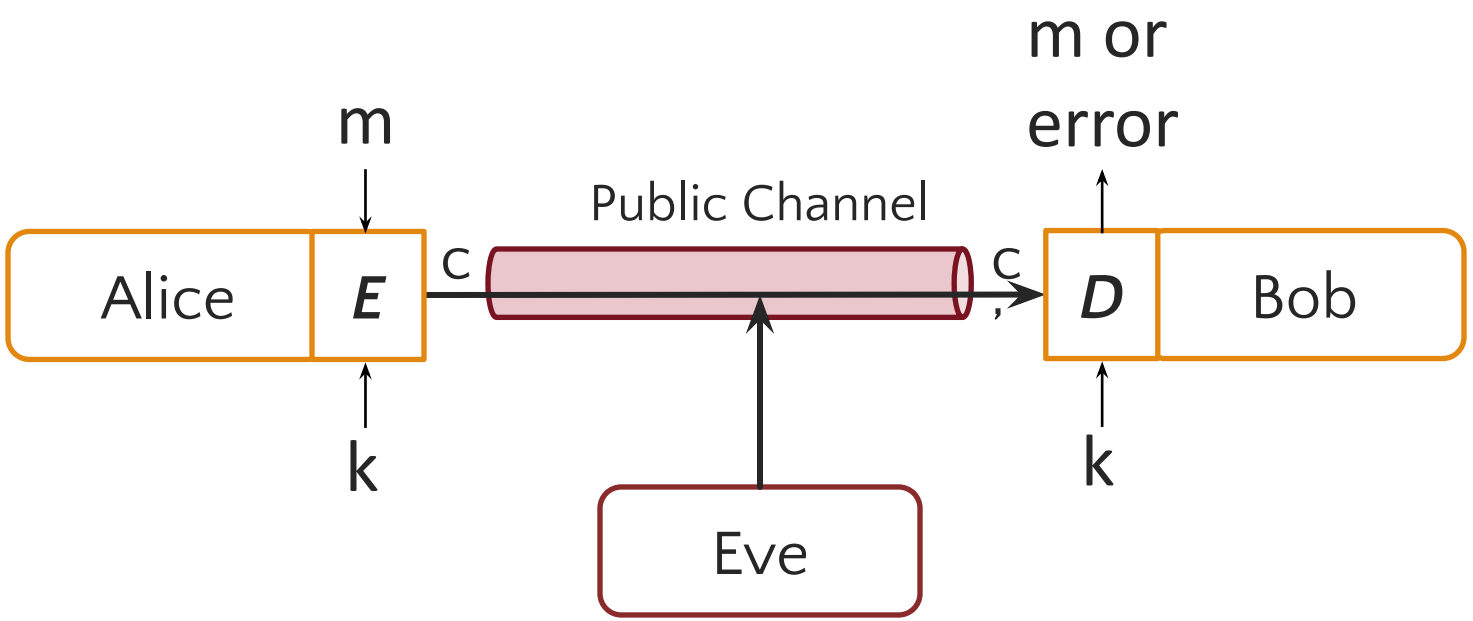
Given *m* and *c* encrypted with an OTP, can you compute the key?

$$E(k, m) = k \oplus m = c$$

1. No
2. Yes, the key is k = m ⊕ c
3. I can only compute half the bits
4. Yes, the key is k = m ⊕ m

So is OTP secure?

Our Goal: Secure Communication



Sub Goal 1: *Secrecy*

Eve should not be able to learn m.

Possible Security Definition

- Given a message space M
- Given a ciphertext c = Enc(k, m) for m in M
- We want Pr[Adversary guesses m] <= 1/|M|

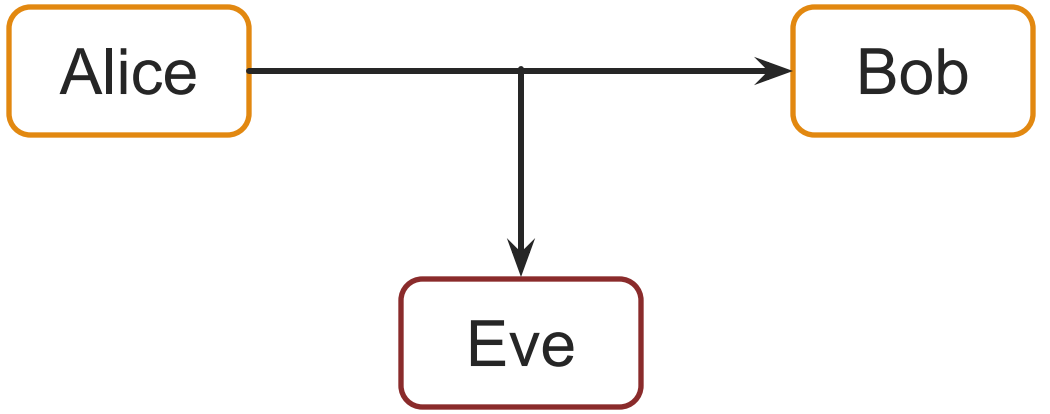
Is this a good definition of security?

A. Yes
B. No
C. I don't know

Sadly, *no cipher* can be secure by that definition!

Suppose Eve knows that there are exactly 3 messages Alice may send *and* the probability of each:

- m_1: The attack is at 12pm. The probability of this message is 1/2
- m_2: The attack is at 3pm. The probability of this message is 1/4
- m_3: The attack is at 5pm. The probability of this message is 1/4



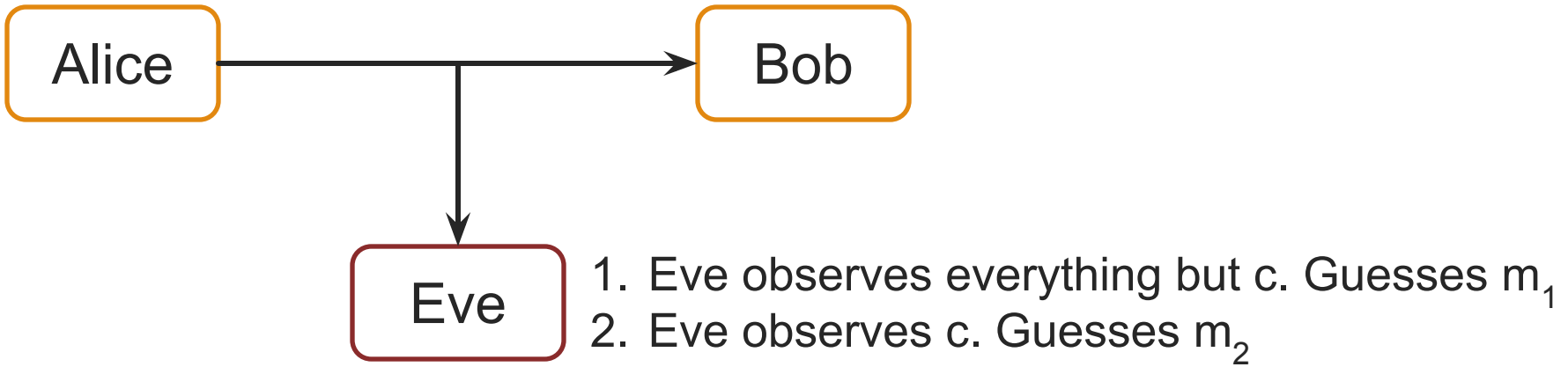
Always guess m_1

M	m_1	m_2	m_3
Pr[M=m]	½	¼	¼

Perfect Secrecy [Shannon1945]
(Information Theoretic Secrecy)

Defn: Perfect Secrecy (informal)

We're no better off determining the plaintext whether or not we see the ciphertext



Perfect Secrecy [Shannon1945]
(Information Theoretic Secrecy)

Defn Perfect Secrecy (formal):

$$k \xleftarrow{\$} \mathcal{K}$$

$$\forall m_0, m_1 \in \mathcal{M} \quad \text{where} \quad |m_0| = |m_1|$$

$$\forall c \in \mathcal{C}$$

$$\Pr\left[E\left(k, m_0\right) = c\right] = \Pr\left[E\left(k, m_1\right) = c\right]$$

Participation Question

How many OTP keys map *m* to *c*?

$$E(k, m) = k \oplus m = c$$
$$D(k, c) = k \oplus c = m$$

A. 1

B. 2

C. Depends on m

Good News: OTP Has Perfect Secrecy

Thm: The one-time pad is perfectly secret

Must show: $$\Pr[E(k, m_0) = c] = \Pr[E(k, m_1) = c]$$

where |M| = |K| = $\{0,1\}^m$

Intuition: Say that M = {00,01,10,11}, and *m* = 11. The Adversary receives c = 10. It asks itself whether the plaintext was m_0 or m_1 (e.g., 01 or 10). It reasons:

- if m_0, then k = $m_0 \oplus c$ = 01 ⊕ 10 = 11.
- if m_1, then k = $m_1 \oplus c$ = 10 ⊕ 10 = 00.

But all keys are equally likely, so Adv doesn't know which one it is.

Good News: OTP Has Perfect Secrecy

Thm: The one-time pad is perfectly secret

Must show: $\Pr[E(k, m_0) = c] = \Pr[E(k, m_1) = c]$

where |M| = |K| = $\{0,1\}^m$

Proof:

$$\Pr[E(k, m_0) = c] = \Pr[k \oplus m_0 = c] \quad (1)$$

$$= \frac{|k \in \{0,1\}^m : k \oplus m_0 = c|}{\{0,1\}^m} \quad (2)$$

$$= \frac{1}{2^m} \quad (3)$$

$$\Pr[E(k, m_1) = c] = \Pr[k \oplus m_1 = c] \quad (4)$$

$$= \frac{|k \in \{0,1\}^m : k \oplus m_1 = c|}{\{0,1\}^m} \quad (5)$$

$$= \frac{1}{2^m} \quad (6)$$

Therefore, $\Pr[E(k, m_0) = c] = \Pr[E(k, m_1) = c]$

All Keys Must Be Equally Likely

Two possible messages

Let M={000,001}

Let K = {000, 001, 010, 011, 100, 101, 110, 111}

8 keys

Note that if K is randomly selected as 000, then M = C.

Is this a problem?

Two-time pad is completely broken!

Two Time Pad:

$c_1 = m_1 \oplus k$

$c_2 = m_2 \oplus k$

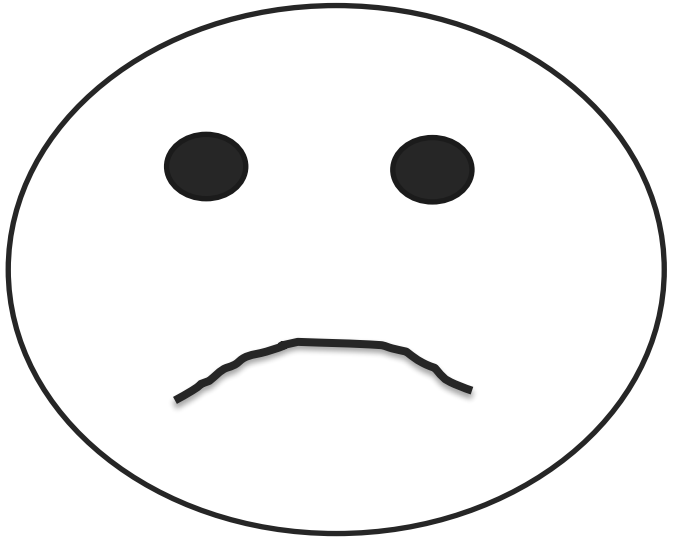
Enough redundancy in ASCII (and English) that $m_1 \oplus m_2$ is enough to know m_1 and m_2

Eavesdropper gets c_1 and c_2
What is the problem?

$c_1 \oplus c_2 = m_1 \oplus m_2$

The “Bad News” Theorem

Theorem: Perfect secrecy requires |K| >= |M|



In practice, we usually shoot for *computational security*

Ευχαριστώ και καλή μέρα εύχομαι!

Keep hacking!