COMPUTER SECURITY

Cryptography

Topics for today

- Cryptography:
  - Problems encryption is designed to solve
  - Encryption tools categories, strengths, weaknesses
    - applications of each
  - Certificates and certificate authorities

CRYPTOGRAPHY

https://www.tripwire.com/state-of-security/security-data-protection/cryptography/ordinary-people-need-cryptography/

Communication Security

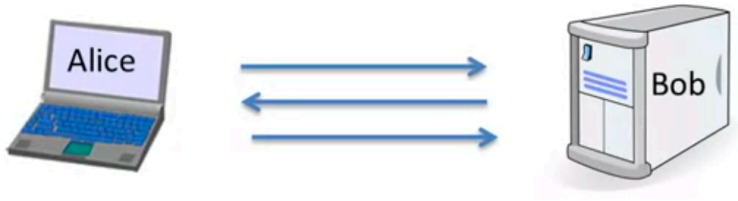
- Protects messages on route from sender to recipient
- An attacker may attempt to:
  - Intercept the message
  - Modify the message
  - Fabricate an authentic-looking alternate message
  - Block the message
- Encryption used to ensure secrecy of message
- Authentication used to ensure integrity

Encryption

- Taking a ***Plaintext*** message
- Applying a ***Cipher*** operation to it
- Results in an unreadable garbled ***Ciphertext*** message

Communication Security

- Two parties trying to communicate
- An eavesdropper tries to intercept message



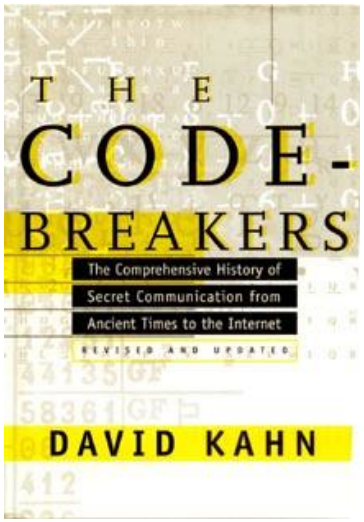
HISTORY

History

- Codes and ciphers usage began as early as 1900 BCE
- Classic cryptography used pen and paper
  - Or mechanical aids
- Mathematical models started developing in the 19th century
  - World war I and II

History of Cryptography

David Kahn, “The code breakers” (1996)



Caesar Cipher: c = m + 3

A B C D E F G H I J

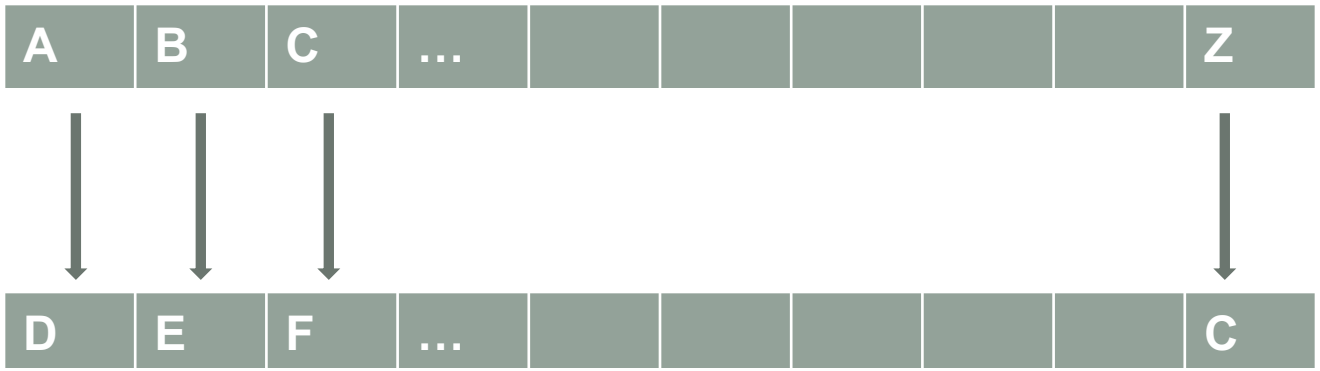
K L M N O P Q R

S T U V W X Y Z

Julius Caesar
100 BC- 44 BC

Caesar cipher

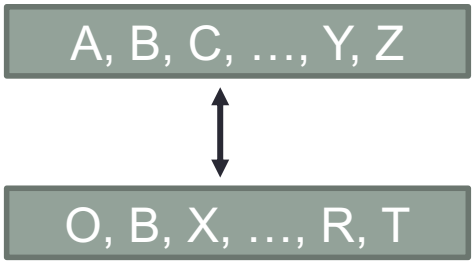
- Shift by some number of characters
  - Size of shift is a key (example = 3)



- Example: ‘cat’ -> ?
  - ‘cat’ -> ‘fdw’

Substitution Ciphers

- More generally, each letter is uniquely replaced by another
- How many different substitution ciphers are there?

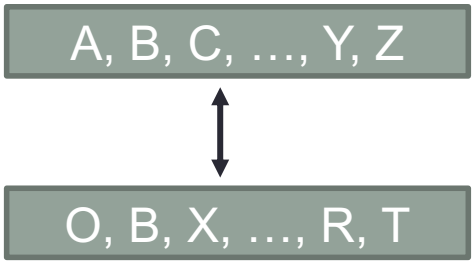


An arbitrary permutation of A,…, Z

Public domain image from http://en.wikipedia.org/wiki/File:ROT13.png

Substitution Ciphers

- More generally, each letter is uniquely replaced by another
- How many different substitution ciphers are there?



An arbitrary permutation of A,…, Z

- There are $26! \approx 4 \times 10^{26}$ such ciphers

Public domain image from http://en.wikipedia.org/wiki/File:ROT13.png

Substitution ciphers

- Key = substitution matrix
  - Example:

A	B	C	…						Z
R	M	G							A

  - Encrypt ‘ABC’:
    - ‘ABC’ -> ‘RMG’
- How many keys are possible?
  - Size of key space = 26!
    - That’s a huge number to search through, even for a computer

Letter encoding

A	B	C	D	E	F	G	H	I	J	K	L	M
↕	↕	↕	↕	↕	↕	↕	↕	↕	↕	↕	↕	↕
0	1	2	3	4	5	6	7	8	9	10	11	12

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
↕	↕	↕	↕	↕	↕	↕	↕	↕	↕	↕	↕	↕
13	14	15	16	17	18	19	20	21	22	23	24	25

Table 2: Encoding English capital letters using integers from $\mathbb{Z}_{26}$.

http://mvngu.wordpress.com/2008/08/20/the-shift-cipher-using-parigp/

Substitution ciphers

- Why are they vulnerable?
  - Letters in a natural language, like English, are not uniformly distributed
  - Knowledge of letter frequencies, can be used in cryptologic attacks against substitution ciphers
    - including pairs and triples

Substitution ciphers

- Frequency Analysis:
  - The practice of studying the frequency with which letters appear in a ciphertext
  - Can be used to attack substitution ciphers

Frequency Analysis

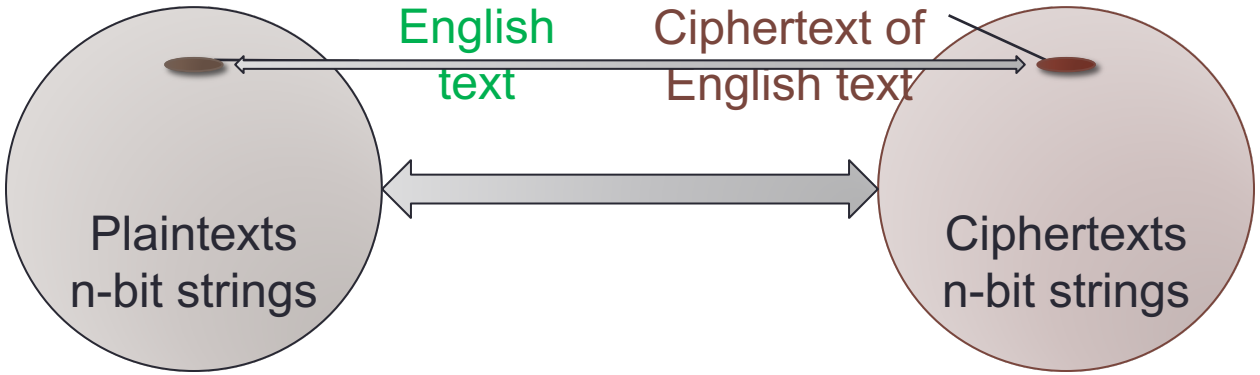
- Single letter frequency analysis:

a:	8.05%	b:	1.67%	c:	2.23%	d:	5.10%
e:	12.22%	f:	2.14%	g:	2.30%	h:	6.62%
i:	6.28%	j:	0.19%	k:	0.95%	l:	4.08%
m:	2.33%	n:	6.95%	o:	7.63%	p:	1.66%
q:	0.06%	r:	5.29%	s:	6.02%	t:	9.67%
u:	2.92%	v:	0.82%	w:	2.60%	x:	0.11%
y:	2.04%	z:	0.06%				

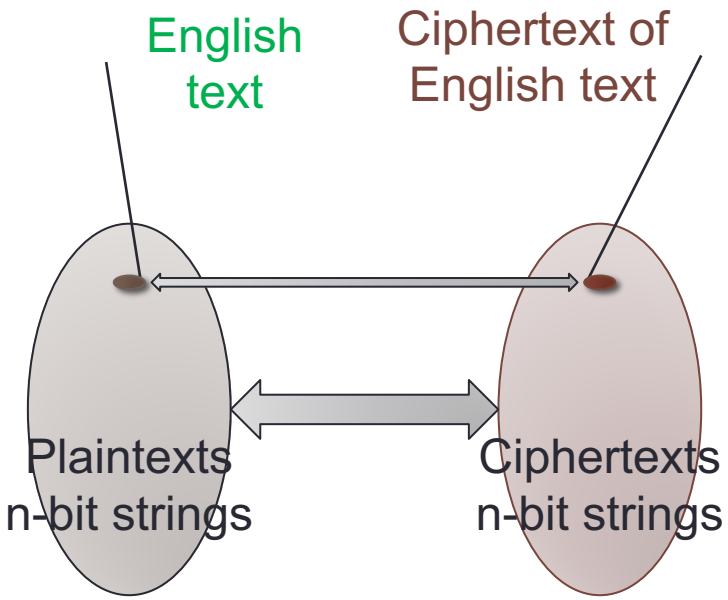
8.1: Letter frequencies in the book *The Adventures of Tom Sawyer*, by Twain.

Encrypting English Text

- English text typically represented with 8-bit ASCII encoding
- A message with t characters corresponds to an n-bit array, with $n = 8t$
- Redundancy due to repeated words and patterns
  - E.g., “th”, “ing”
- English plaintexts are a very small subset of all n-bit arrays



Encrypting English Text



Entropy of Natural Language

- How much information can an alphabet with 8 characters carry?
  - 3 bits of information
  - $2^3 = 8$
- For the English language, each character can convey log (26) = 4.7 bits of information

Entropy of Natural Language

- However, meaningful English text is only ~1.25 bits per char
- Therefore, plaintext redundancy = 4.7 – 1.25 = 3.45
- For example, if a word has 8 characters, what is the effective dictionary size?
  - How many words on average will we have?
  - $2^{8*1.25} = 2^{10} = 1024$

Entropy of English Language

- How do you statistically calculate the entropy of the next letter when the previous $N - 1$ letters are known?
  - In a word
- As N increases, the entropy approaches the entropy of English

	F_0	F_1	F_2	F_3
26 letter	4.70	4.14	3.56	3.3

http://people.seas.harvard.edu/~jones/cscie129/papers/stanford_info_paper/entropy_of_english_9.htm

Entropy of English Language

- Do we include the space?
  - The 27-letter sequences include the space as a letter
  - Onecanalmostalwaysfillinthespacesfromasequenceofwordswithnospaces
  - Therefore, spaces are basically redundant
    - will cause lower calculated entropies when taken into account

Entropy of English Language

- Do we include the space?
  - Entropy of each letter in a string in English:

	F_0	F_1	F_2	F_3
26 letter	4.70	4.14	3.56	3.3
27 letter	4.76	4.03	3.32	3.1

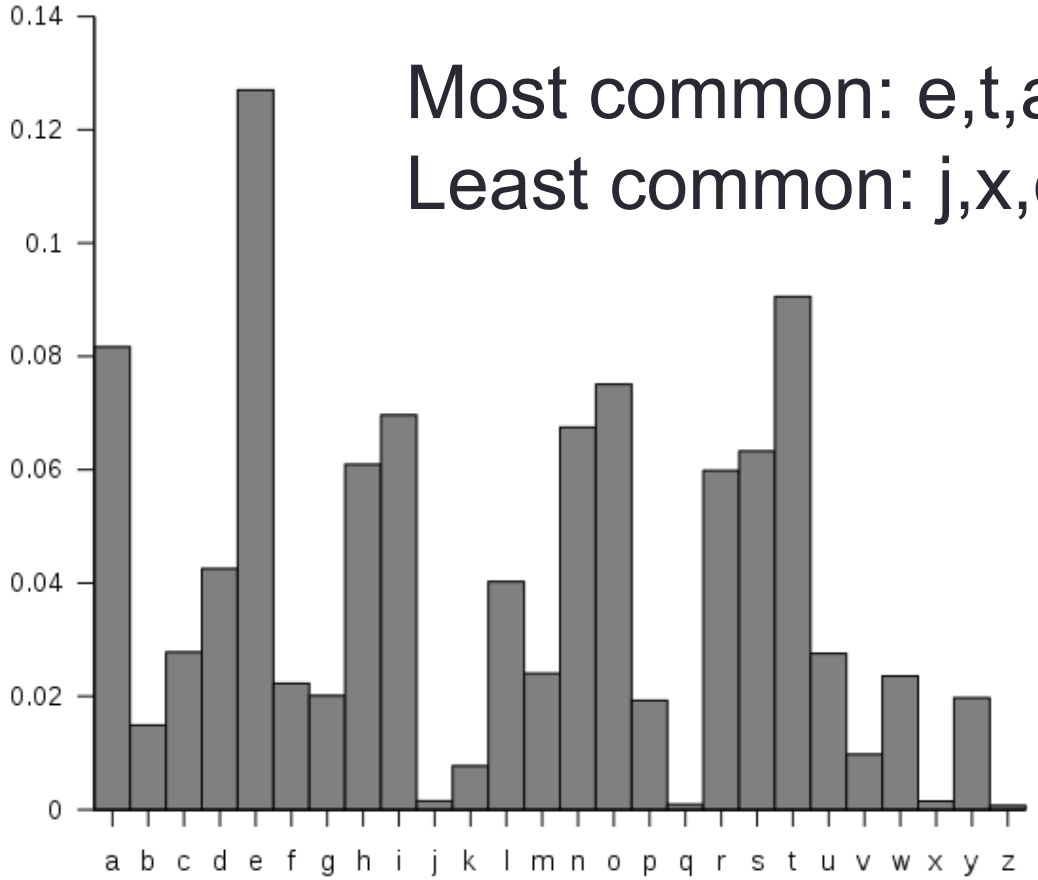
    - Spaces are basically redundant
      - will cause lower calculated entropies when taken into account
    - Only in the F_0 case, where no statistics are taken into account, is the entropy higher when the space is added
      - This simply adds another possible symbol, which means more uncertainty

http://people.seas.harvard.edu/~jones/cscie129/papers/stanford_info_paper/entropy_of_english_9.htm

Attacking Substitution Ciphers

Trick 1: *Word Frequency*

Trick 2: *Letter Frequency*



Most common: e,t,a,o,i,n

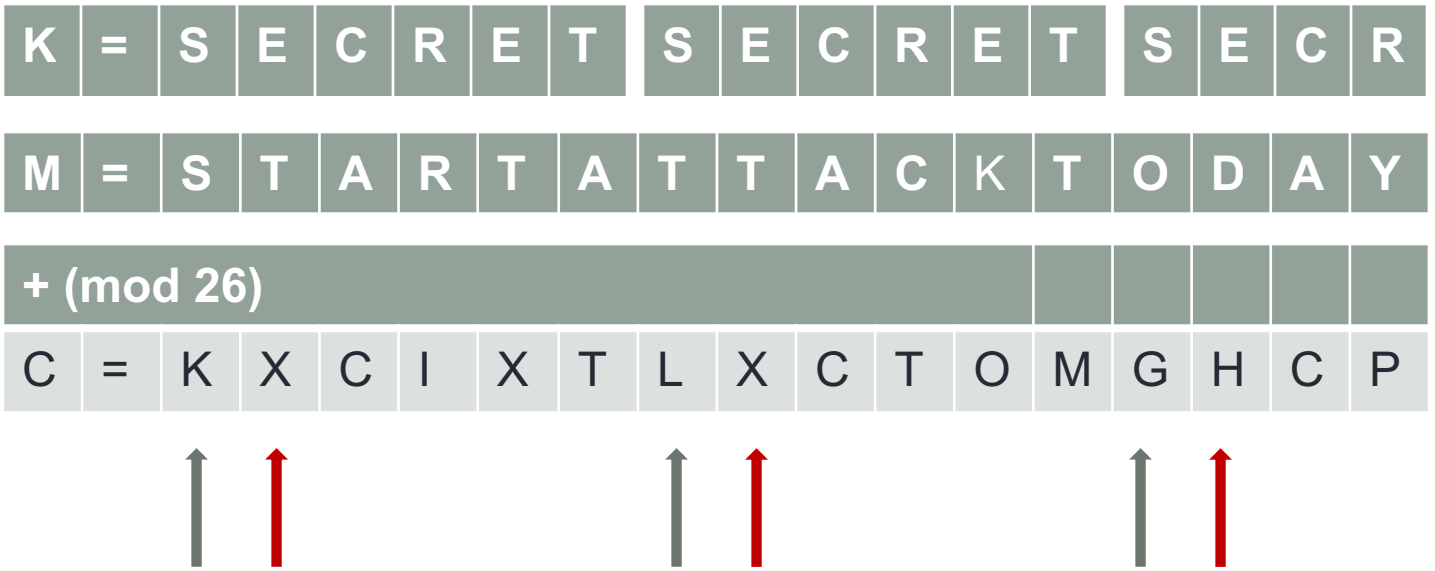
Least common: j,x,q,z

image source: wikipedia

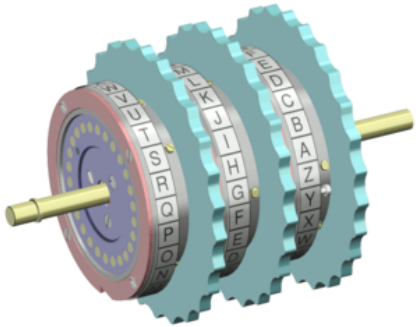
Jvl mlwclk yr jvl owmwez twp yusl w zyduo pjdcluj mqil zydkplmr. Hdj jvlz tykilc vwkc jy mlwku jvl wkj yr vwsiquo, tvqsv vlmflc mlwc jvlg jy oklwjulpp. Zyd vwnl jvl fyjlujqwm jy cy jvl pwgl. Zydk plsklj fwpptykc qp: JYWPJ

Vigenere Cipher (Rome, 16th' century)

- A series of different Caesar ciphers
  - Described by Giovan Battista Bellaso
    - Named for Blaise de Vigenère
  - Example:

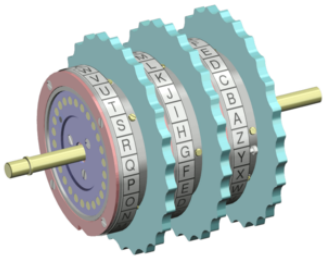


Mechanical Aids



- Rotor machine is an electro-mechanical stream cipher device used for encrypting and decrypting secret messages
  - Used in the 1920s–1970s
  - E.g., in the ***Enigma system***

Rotor machine



- Machine has rotating disks with an array of electrical contacts on either side
- Implements a fixed substitution of letters, replacing them in some complex fashion
- After encrypting each letter, the rotors advance positions, changing the substitution
- Produces a complex substitution cipher, which changes with every keypress

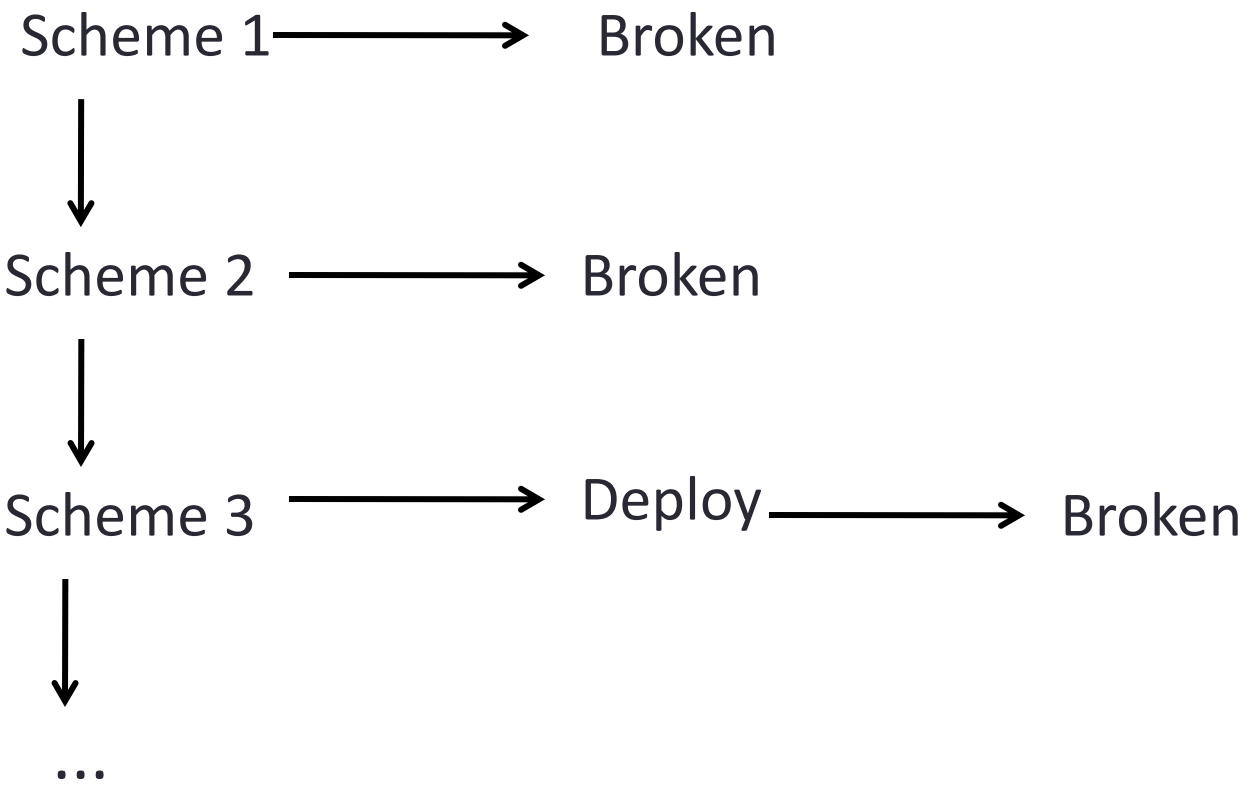
https://en.wikipedia.org/wiki/Rotor_machine

History

- During world war II, mathematical and statistical methods started being developed
- Successfully used to break into the German Enigma machine:
  - Messages were deciphered by the Allies
  - producing intelligence code-named Ultra

MODERN CRYPTOGRAPHY

Classical Approach: Iterated Design

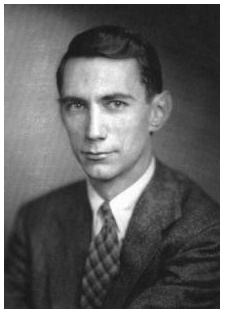


No way to say anything is secure
(and you may not know when broken)

Iterated design was only one we knew until 1945

Claude Shannon: 1916 - 2001

Claude Shannon



- Formally defined:
  - *security goals*
  - *adversarial models*
  - *security of system with regard to goals*
- Beyond iterated design: proof!

Encryption Terminology

- *Sender*
- *Recipient/Receiver*
- *Transmission medium*
- *Interceptor/intruder/Attacker/Adversary*
- *Encrypt, encode, or encipher*
  - Process that hides the meaning of the message
- *Decrypt, decode, or decipher*
  - Reveal the original message

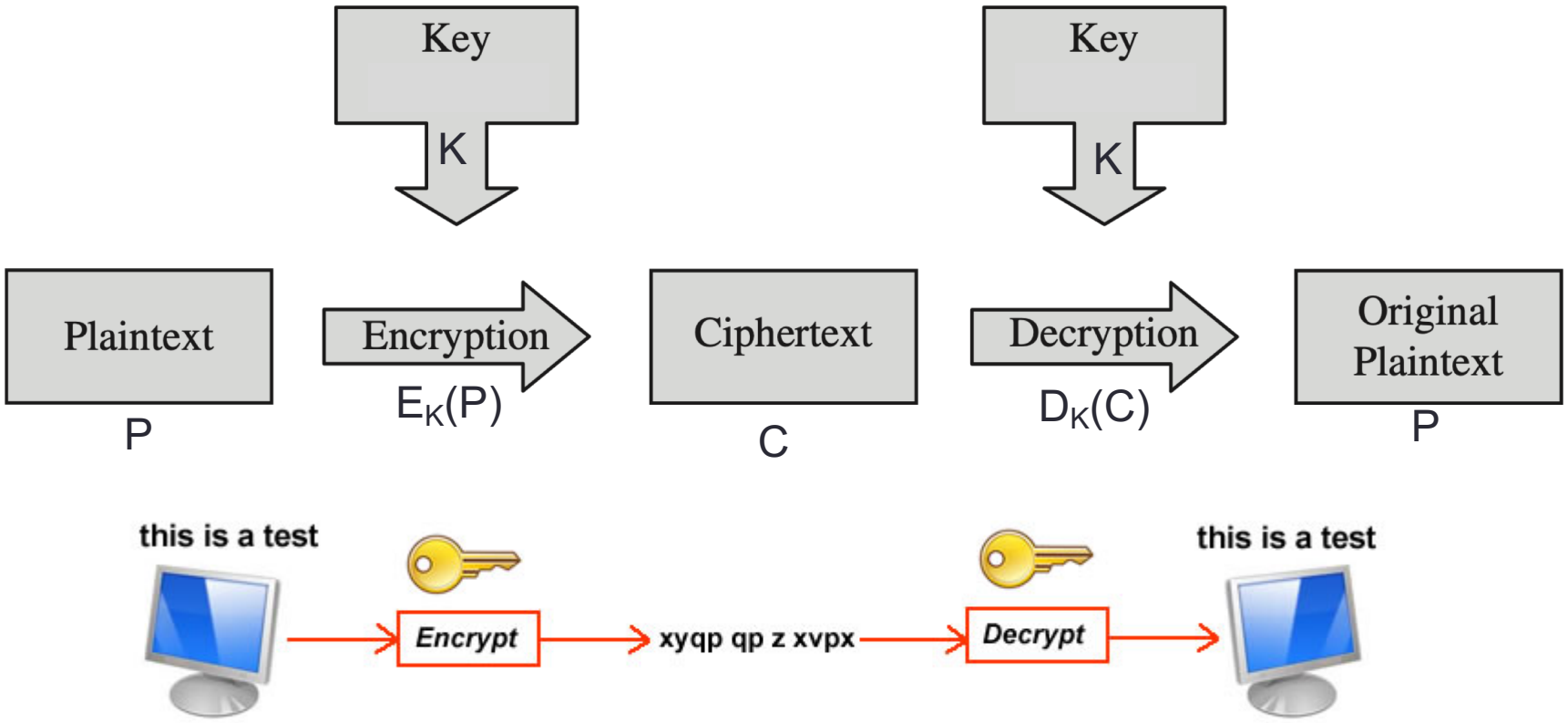
Encryption Terminology

- *Cryptosystem*
  - A system for encryption and decryption
- *Plaintext*
  - Original message
- *Ciphertext*
  - Encrypted message

Encryption Basics

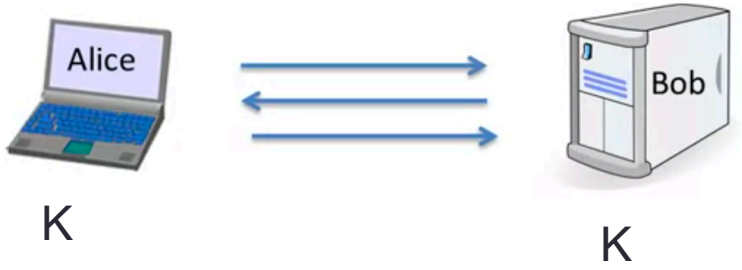
- Sender and Recipient often share a secret key
  - Known to them, but not anyone else
- An encryption process, used by sender
  - Takes plaintext and the key
  - Produces the encrypted ciphertext
- A decryption process, used by recipient
  - Takes ciphertext and the key
  - Recovers the original plaintext message

Encryption Basics

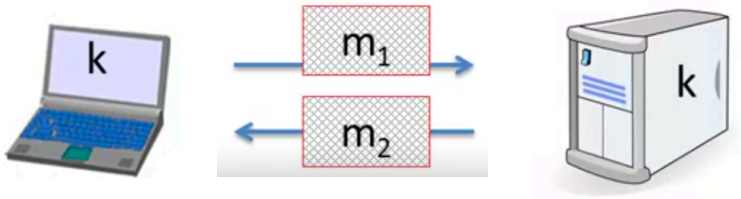


What is Cryptography?

- Two main functions:
  - Secret key establishment:
    - Both parties need to agree on a secret key



  - Secure communication:`
    - Use key to encrypt message
      - Provide confidentiality and integrrity

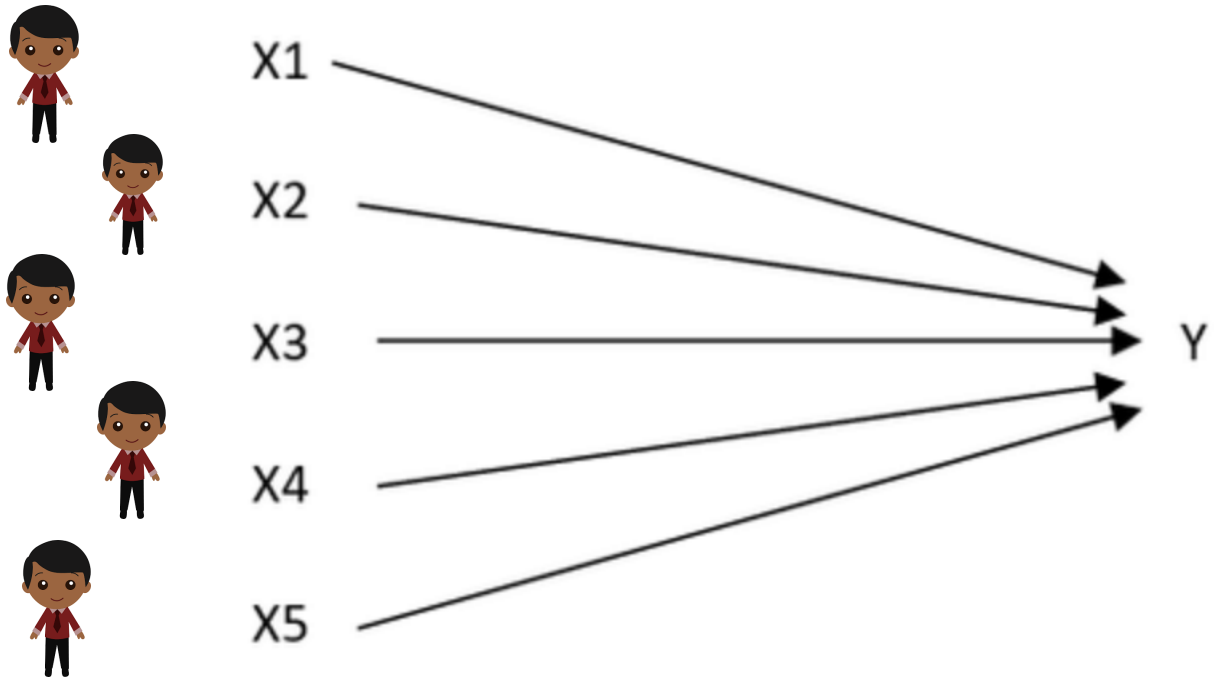


What is Cryptography?

- Digital Signatures
  - In the analog world, I can use the same signature to sign multiple documents
    - Not possible in the digital world
  - The signature has to be a function of the content being signed
    - Copying the signature from one document to the other will not work

What is Cryptography?

- Secure multi-party computation
  - Can be used to compute election, auction results



What is Cryptography?

- Secure multi-party computation (cont.)
  - Goal: Compute f(x1, x2, x3, x4, x5)
    - Can be done with a trusted party
      - i.e., ebay, election mechanism, etc.
  - Theorem:
    - Anything that can be done with a trusted authority, can also be done without
      - Through an algorithm
        - May take very long time, depending on the function calculated

What is Cryptography?

- Zero-Knowledge (proof of knowledge)
  - One party proves to the other that he knows something
    - Without sharing this data

What is Cryptography

- For each concept in cryptography:
  - Threat model has to be specified precisely
  - Solution is proposed
  - Breaking the solution will be hard
    - Too long to compute

Security through Obscurity

http://www.treachery.net/articles_papers/tutorials/why_security_through_obscurity_isnt/index.htm

Security through Obscurity

- Reliance on the secrecy of the design or implementation
  - as the main system security method
    - or component of a system
- System may have security vulnerabilities
  - System designers believe that if the flaws are not known, it prevents a successful attack

http://ithare.com/advocating-obscurity-pockets-as-a-complement-to-security-part-ii-deployment-scenarios-more-crypto-primitives-and-obscurity-pocket-as-security/

Security through Obscurity

- Reliance on the secrecy of the design or implementation
  - as the main system security method
    - or component of a system
- Rejected by security experts!
  - obscurity should never be the only security mechanism!
  - has been historically used without success by several organizations

http://ithare.com/advocating-obscurity-pockets-as-a-complement-to-security-part-ii-deployment-scenarios-more-crypto-primitives-and-obscurity-pocket-as-security/

How is it related to Cryptography?

- Even if everything is known but the key, system should still be secure
- A.K.A. **Kerchoffs's principle**

Kerchoffs's principle

- A cryptosystem should be secure even if everything about the system is public knowledge
  - except the ***key***
- A.K.A. “Shanon’s Maxim”
  - The enemy knows the system

Cryptography Terminology

- ***Cryptology***: Practice of coding and hiding messages

- ***Cryptanalysis***: Practice of “breaking” messages
  - Trying to decipher hidden messages
    - Frequency analysis is an example of cryptanalysis method

Steganography

- Hiding information from observers
- Does not necessarily use encoding
  - But data is hidden from site
- Examples:
  - Message written in invisible ink
  - Text hidden in picture
    - Pixels not visible to the naked eye, reader needs to know about them to look for them.
- What do we need to make steganography more secure?

Steganography

- Some implementations of steganography lack a shared secret
  - => forms of security through obscurity
- Key-dependent steganographic schemes adhere to Kerckhoffs's principle.

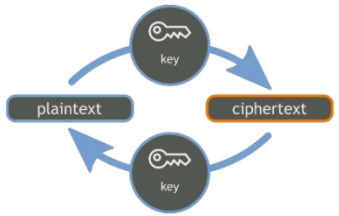
ENCRYPTION

Encryption Basics

- A cryptosystem involves a set of rules for how to encrypt a plaintext and decrypt the ciphertext
  - Rules == algorithms
- The resulting ciphertext depends on:
  - The algorithm (typically public and known to all)
  - The original message
  - The key value

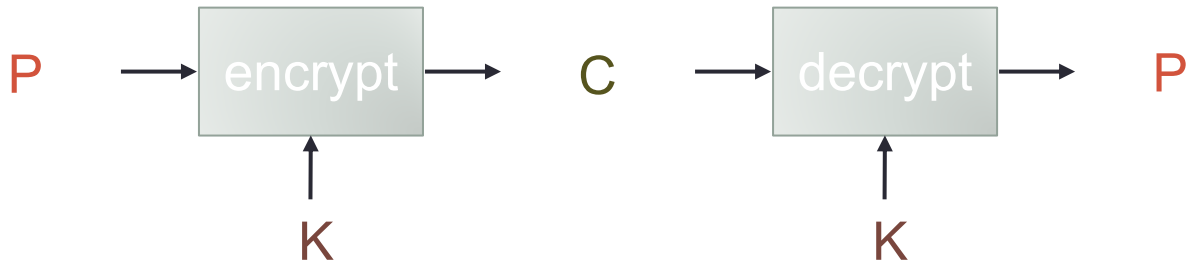
Symmetric Encryption

- *Symmetric encryption* uses the same key, K, both to encrypt a message and later to decrypt it
  - Also called single-key or secret key encryption
  - Uses a pair of efficient algorithms (E,D)
    - D and E are mirror-image processes
    - E is often randomized
    - D is always deterministic

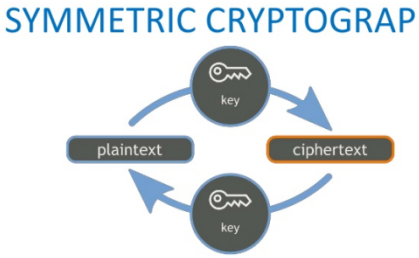


Symmetric Cryptosystem

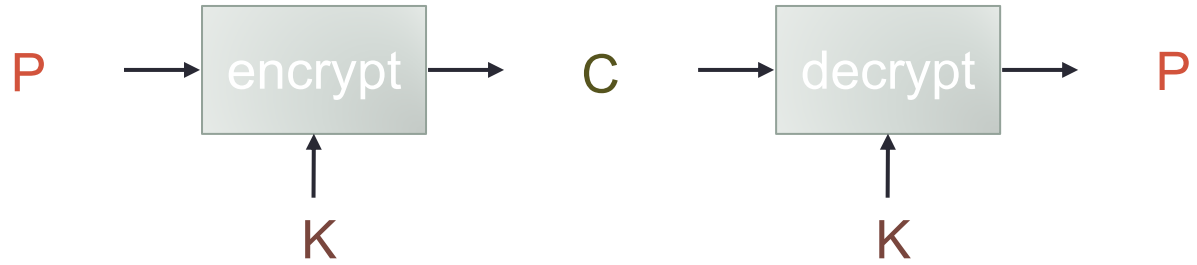
- Scenario
  - Alice wants to send a message (plaintext P) to Bob.
  - The communication channel is insecure and can be eavesdropped
  - If Alice and Bob have previously agreed on a symmetric encryption scheme and a secret key K
    - => the message can be sent encrypted (ciphertext C)



Symmetric Cryptosystem



- Issues
  - What is a good symmetric encryption scheme?
  - What is the complexity of encrypting/decrypting?
  - What is the size of the ciphertext, relative to the plaintext?



https://www.slideshare.net/TusharSwami1/cryptography-72969031

Cryptanalysis

- A cryptanalyst's goal is to break an encryption
  - Attempts to deduce the original meaning of a ciphertext message
  - Attempts to determine which decrypting algorithm, and key matches the encrypting algorithm *t*
    - to be able to break other messages encoded in the same way

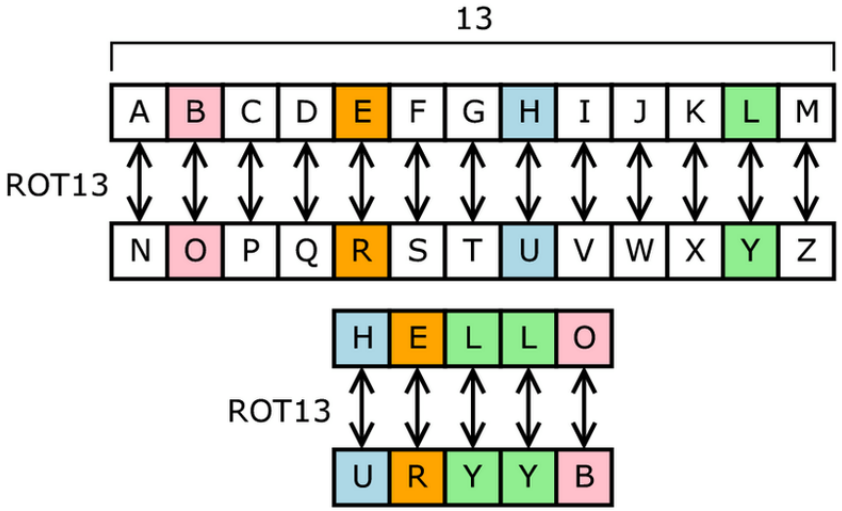
Cryptanalysis

- An encryption algorithm is called ***breakable*** if:
  - it is feasible to decrypt the original message without knowing the key
    - given enough time and data
- However, an algorithm that is theoretically breakable may be impractical to break
  - May take too long (e.g. billions of years)
- The difficulty of breaking an encryption is called its ***work factor***

CIPHERS

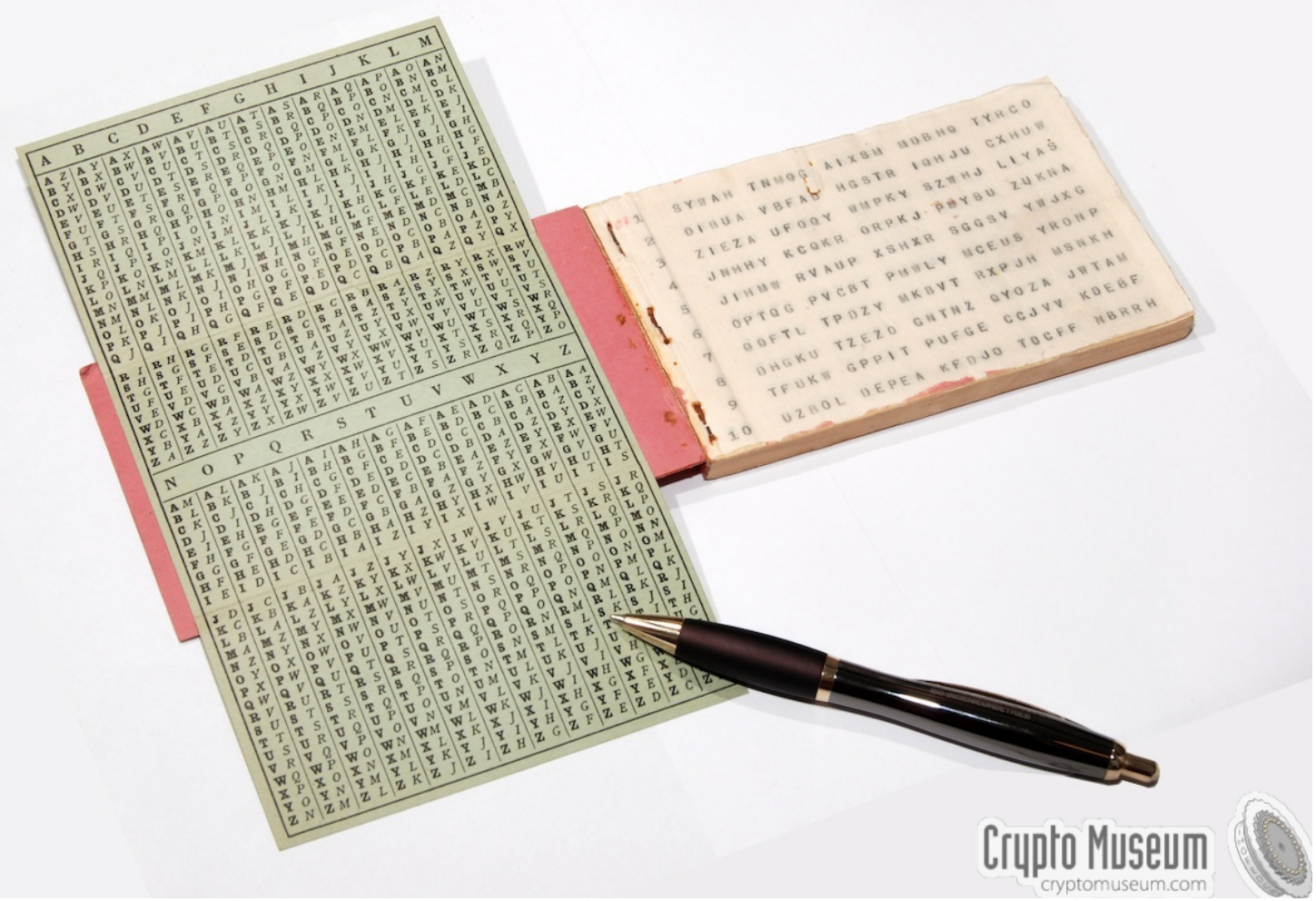
Substitution Ciphers

- Traditional ciphers, used for 1000s of years
  - Not used in modern systems anymore
- One popular substitution “cipher” for some Internet posts is ROT13.



Public domain image from http://en.wikipedia.org/wiki/File:ROT13.png

One-Time Pad



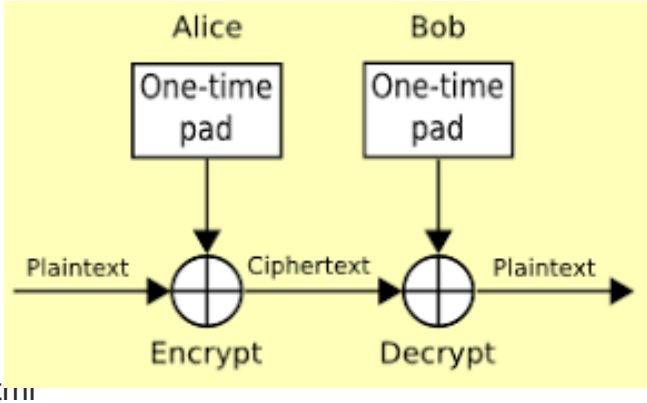
- https://www.cryptomuseum.com/crypto/otp/index.htm

One-Time Pad

- https://www.cryptomuseum.com/crypto/otp/index.htm

One-Time Pads

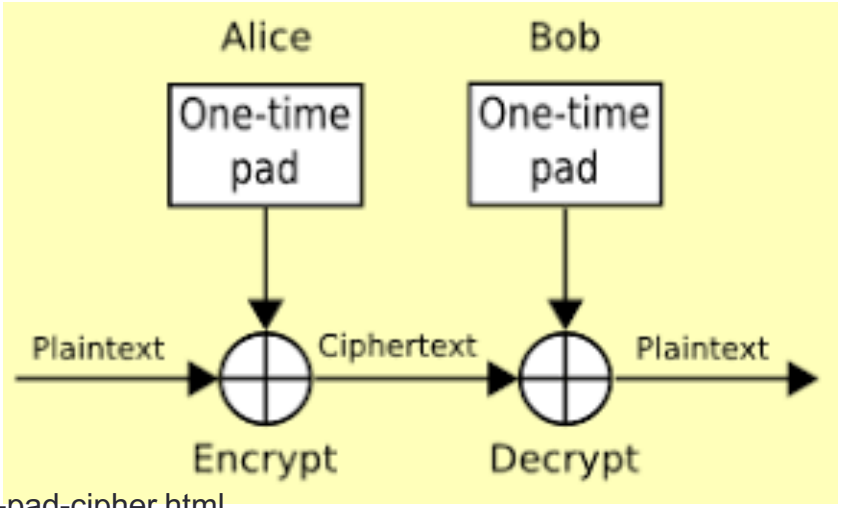
- One variation of substitution cipher is theoretically unbreakable.
  - The one-time pad was invented in 1917 by Joseph Mauborgne and Gilbert Vernam
  - We use a block of shift keys, $(k_1, k_2, \ldots, k_n)$, to encrypt a plaintext, M, of length n
    - with each shift key being chosen uniformly at random



https://programmingcode4life.blogspot.com/2015/10/one-time-pad-cipher.html

One-Time Pads

- We use a block of shift keys, $(k_1, k_2, \ldots, k_n)$, to encrypt a plaintext, M, of length n
  - with each shift key being chosen uniformly at random
- Since each shift is random, every ciphertext is equally likely for any plaintext.



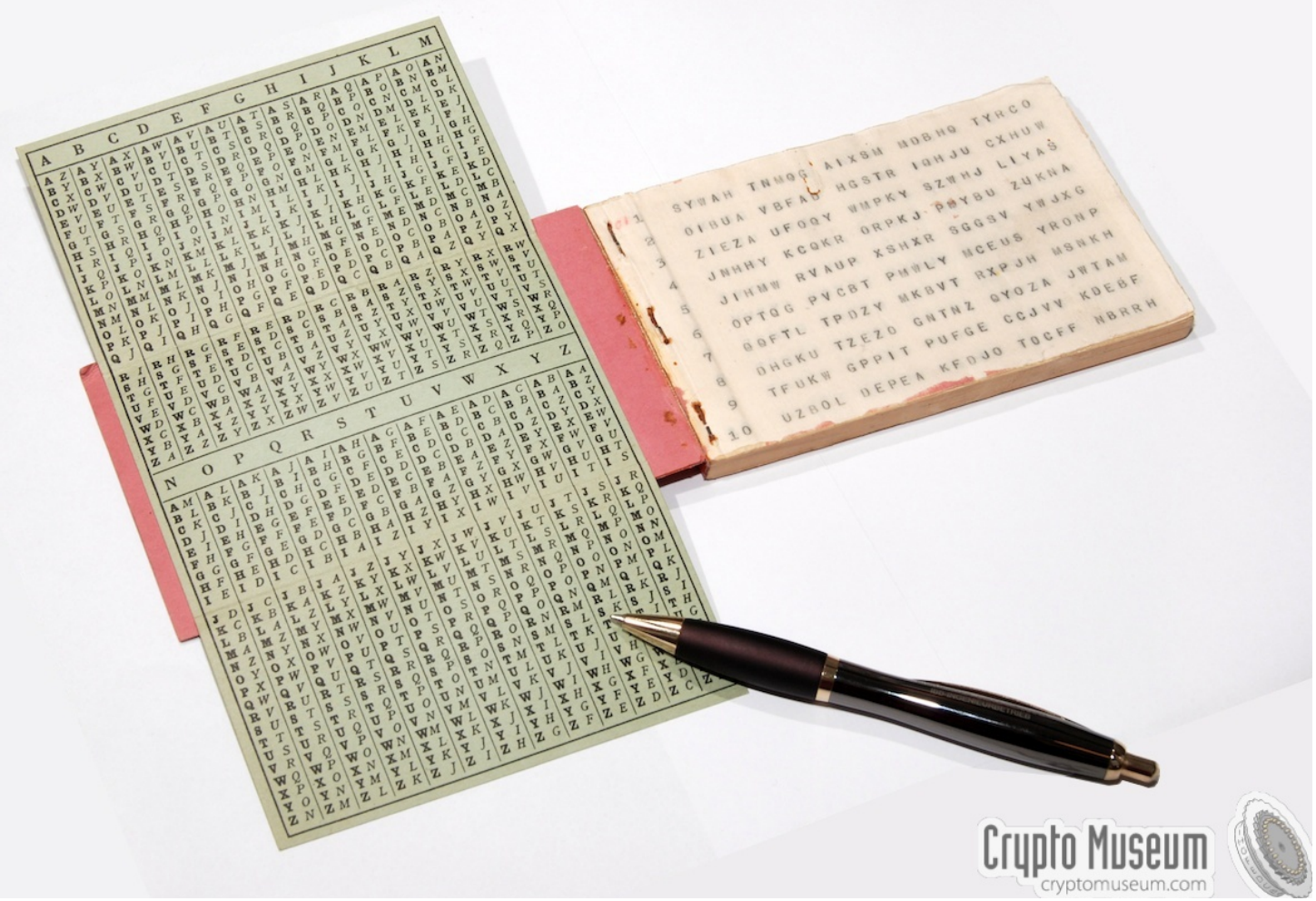
https://programmingcode4life.blogspot.com/2015/10/one-time-pad-cipher.html

Conversion Table

Conversion Table

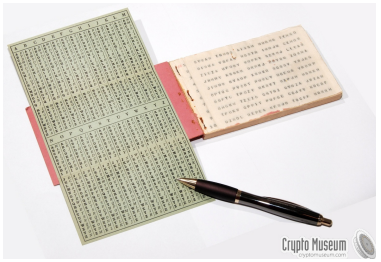
A	= 1	K	= 11	U	= 21
B	= 2	L	= 12	V	= 22
C	= 3	M	= 13	W	= 23
D	= 4	N	= 14	X	= 24
E	= 5	O	= 15	Y	= 25
F	= 6	P	= 16	Z	= 26
G	= 7	Q	= 17		
H	= 8	R	= 18		
I	= 9	S	= 19		
J	= 10	T	= 20		

One-Time Pad



Weaknesses of the One-Time Pad

- While perfect secure in theory, one-time pads have some weaknesses
- The key has to be as long as the plaintext
- Keys can never be reused
  - Repeated use of one-time pads allowed the U.S. to break some of the communications of Soviet spies
    - during the Cold War.



Publication: https://www.cia.gov/library/center-for-the-study-of-intelligence/csi-publications/books-and-monographs/venona-soviet-espionage-and-the-american-response-1939-1957/part2.htm

STREAM AND BLOCK CIPHER

Stream Encryption

- Each bit or byte of the data stream is encrypted separately
  - May be applicable for data stream processing

Stream Encryption

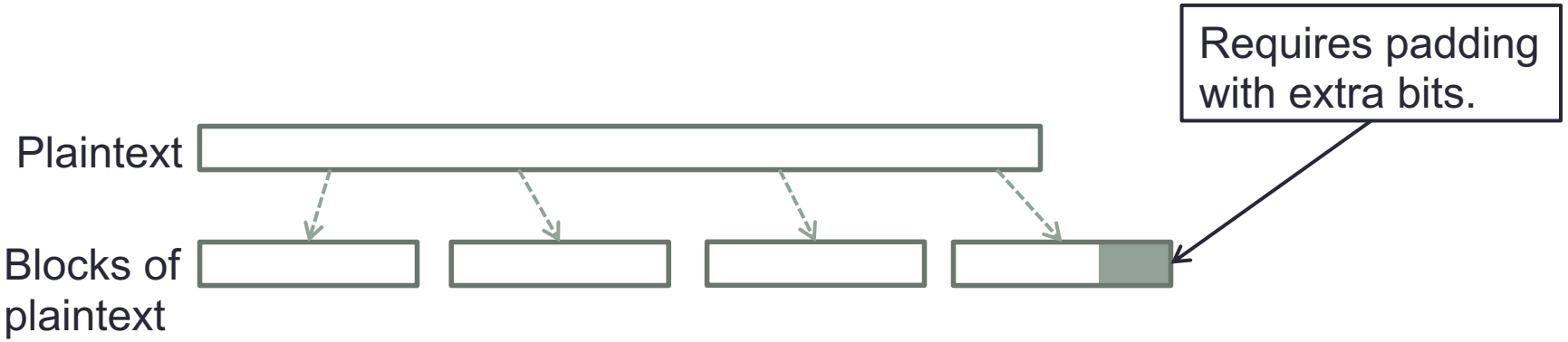
- Advantage:
  - it can be applied immediately to whatever data items are ready to transmit
- Disadvantage:
  - most encryption algorithms involve complex transformations
    - To do these transformations on one or a few bits at a time is expensive

Block Ciphers

- Perhaps the most-used technique for encryption
  - Groups plaintext symbols into fixed-size blocks
  - A block cipher algorithm performs its work on a quantity of plaintext data all at once

Block Ciphers

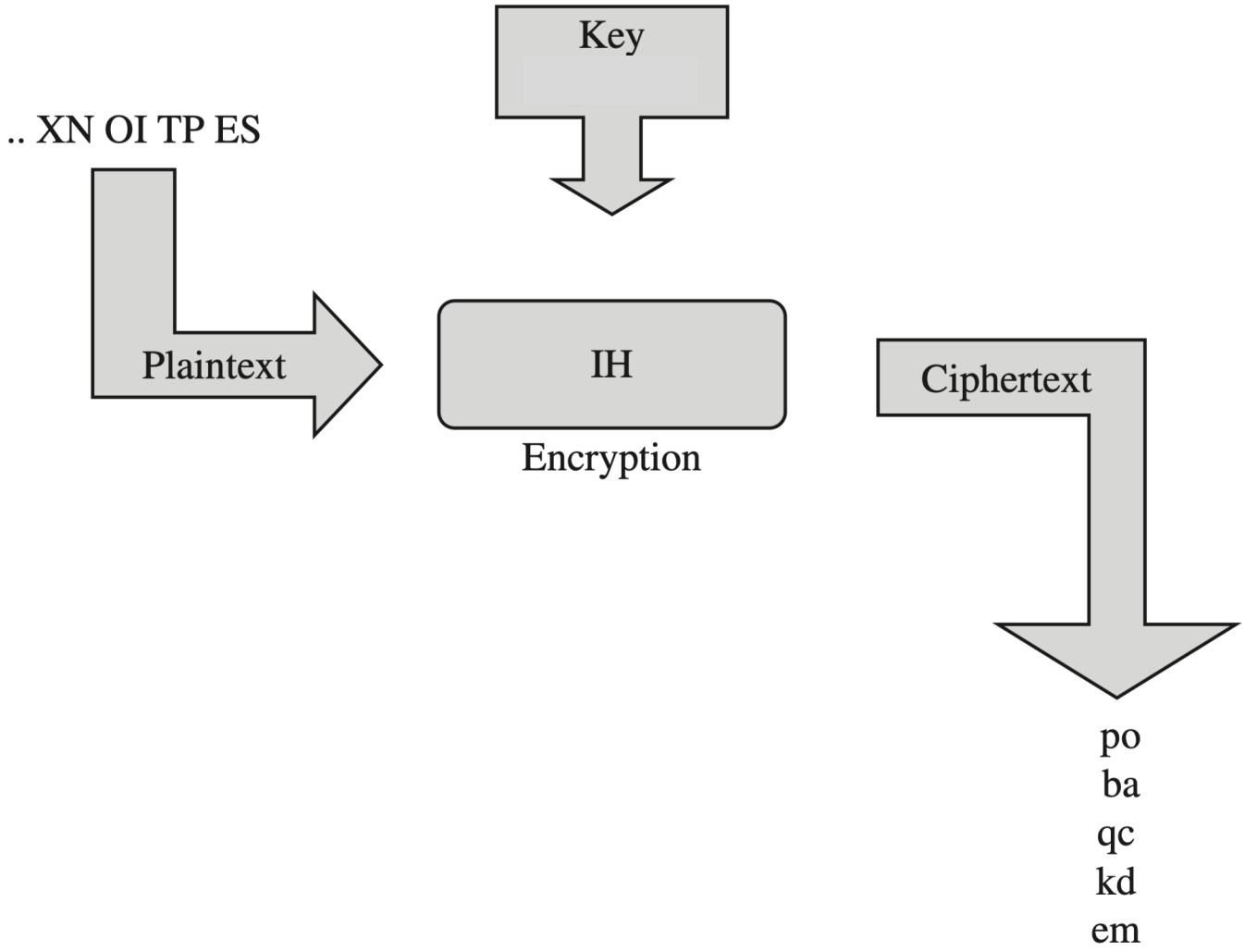
- Plaintext is partitioned into a sequence of blocks,
  - $P = P[0], \ldots, P[m-1]$
- Each block is of fixed length b (e.g., 128 bits)
- Blocks encrypted (more or less) separately
  - BlockCipher(key, plaintextBlock) → ciphertextBlock



Padding

- Plaintext length must be a multiple of the block size
  - Otherwise we must pad last partial block to a full block
  - On decryption, recipient must be able to tell when data ends, padding begins
- Example for block-size = 128 (16 bytes)
  - Plaintext: “Roberto” (7 bytes)
  - Padded plaintext: “Roberto999999999” (16 bytes)
    - Problem: cannot tell if plaintext was “Roberto” or “Roberto9”
  - Better: padded plaintext “Roberto099999999”

Block Ciphers



Stream vs. Block Cipher

	Stream	Block
Encryption	Individual Chars (bits)	Groups of chars (blocks)
Speed	Faster	Slower
Hardware Circuitry	Simpler	More complex
Data Buffering	Limited or none	More space, relative to block size
Error Propagation	Limited Good for noisy channel	Faster Helps assure message integrity

www.ccs.neu.edu/home/noubir/Courses/CSG252/F04/slides4.pdf

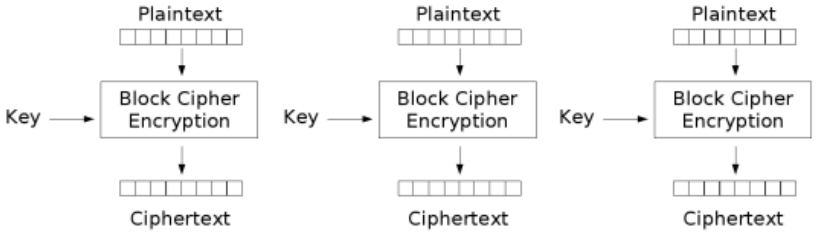
BLOCK CIPHER MODE

Block Cipher Modes

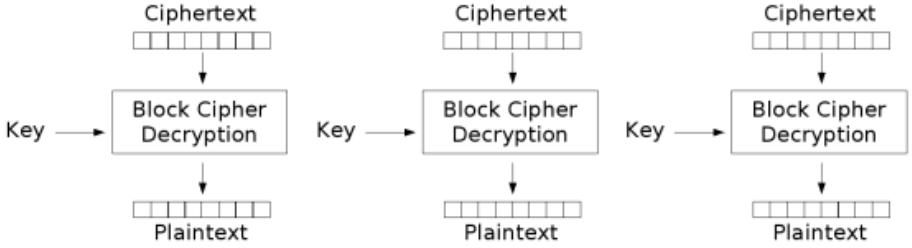
- Describe the way a block cipher encrypts and decrypts a sequence of message blocks
- Many modes exist
  - "Simple": ECB, CBC, CTR, …
  - "Authenticated": GCM, CCM, OCB, …
  - Special purpose: CMC, EME, "tweakable modes"
- We will cover ECB and CBC as examples

Electronic Code Book (ECB) Mode

- The simplest: Electronic Code Book (ECB) Mode
  - Block P[i] encrypted into ciphertext block C[i] = EK(P[i])
  - Block C[i] decrypted into plaintext block M[i] = DK(C[i])



Electronic Codebook (ECB) mode encryption



Electronic Codebook (ECB) mode decryption

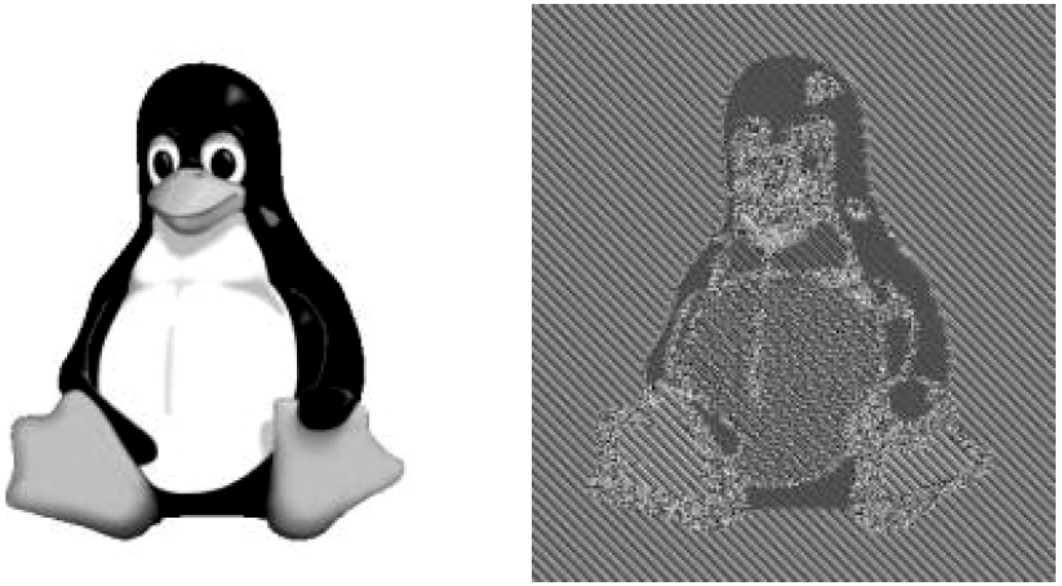
Strengths of ECB

- Very simple
- Allows for parallel encryptions of the blocks of a plaintext
- Can tolerate the loss or damage of a block

Weaknesses of ECB

- Not secure enough, patterns in the plaintext are repeated in the ciphertext
- Documents and images are not suitable for ECB encryption

Weaknesses of ECB



(a) (b)

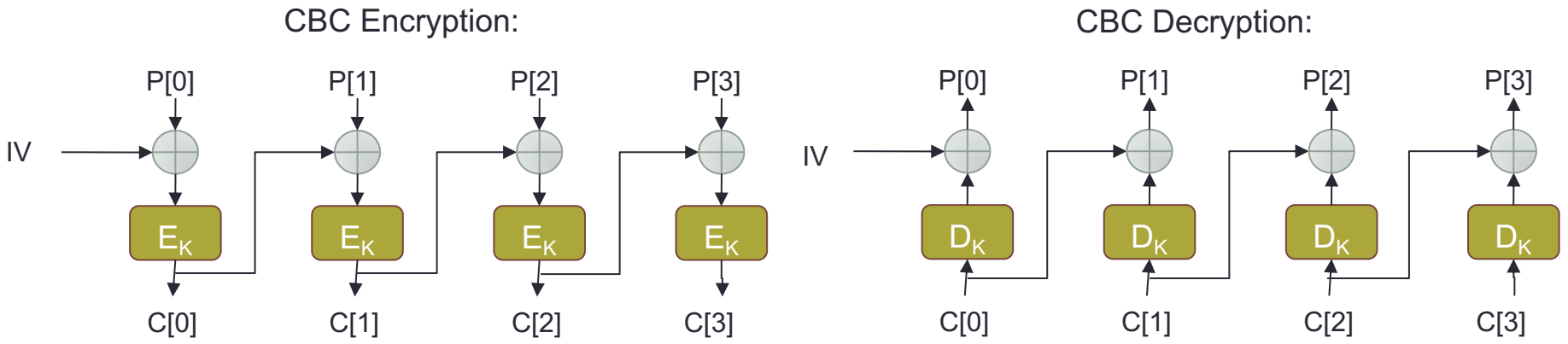
Figure 8.6: How ECB mode can leave identifiable patterns in a sequence of blocks: (a) An image of Tux the penguin, the Linux mascot. (b) An encryption of the Tux image using ECB mode. (The image in (a) is by Larry Ewing, lewing@isc.tamu.edu, using The Gimp; the image in (b) is by Dr. Juzam. Both are used with permission via attribution.)

Cipher Block Chaining (CBC) Mode

- How do we overcome this weakness?
  - Make different blocks depend on each other
  - Also use ***randomized encryption***: encrypting the same thing many times yield a different ciphertext every time

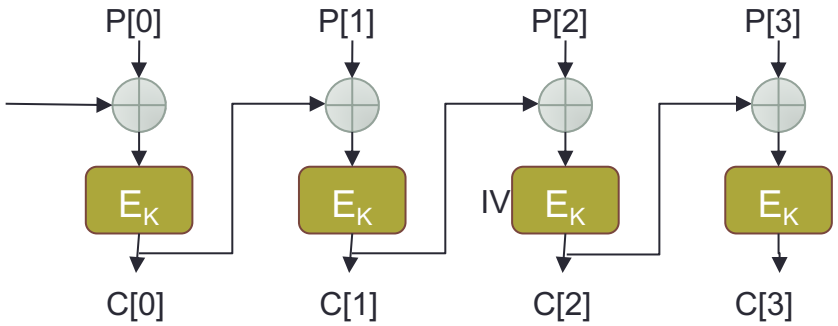
Cipher Block Chaining (CBC) Mode

- In Cipher Block Chaining (CBC) Mode
  - The previous ciphertext block is combined with the current plaintext block $C[i] = E_K(C[i-1] \oplus P[i])$
  - $C[1] = IV$, a random block separately transmitted encrypted (known as the initialization vector)
  - Decryption: $P[i] = C[i-1] \oplus D_K(C[i])$

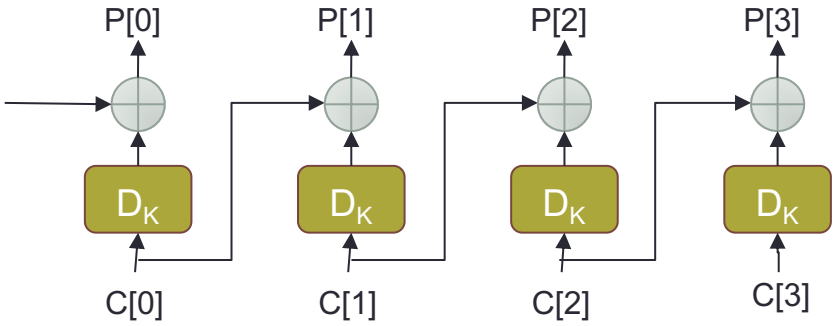


Cipher Block Chaining (CBC) Mode

CBC Encryption:



CBC Decryption:



Cipher Block Chaining (CBC) Mode

- How do we overcome this weakness?
  - Make different blocks depend on each other
  - Also use ***randomized encryption***: encrypting the same thing many times yield a different ciphertext every time

- CBC is a ***randomized encryption*** mode
  - if we choose a different iv – initial value

Strengths of CBC

- Fast and relatively simple
- Doesn’t show patterns in the plaintext
- Is the most common mode in practice
  - Usually in conjunction with an authentication method

Weaknesses of CBC

- Weaknesses:
  - ***Encryption*** is not parallelizable
    - must be done sequentially
  - Requires reliable transmission of all the blocks sequentially
    - Not suitable for applications with packet losses (e.g., music and video streaming)
  - Still not secure enough by itself: does not provide authentication

Authentication

- Simple modes such as ECB, CBC, CTR **do not provide authentication**
  - Attacker may be able to manipulate ciphertext, inducing meaningful changes on the decrypted value
  - Some of these attacks are very sophisticated
- These modes must be used in conjunction with some other authentication mechanism
  - Providing integrity for the encrypted data

Authentication

- Some modes are specifically designed to provide both secrecy and authentication
  - E.g., GCM (Galois/Counter Mode), CCM, OCB, …

BLOCK CIPHERS IN PRACTICE

Substitution Boxes

- Substitution still used as one component (among others) in modern ciphers
- Usually applied to numbers
  - Described by substitution boxes, or S-boxes.

	00	01	10	11
00	0011	0100	1111	0001
01	1010	0110	0101	1011
10	1110	1101	0100	0010
11	0111	0000	1001	1100

(a)

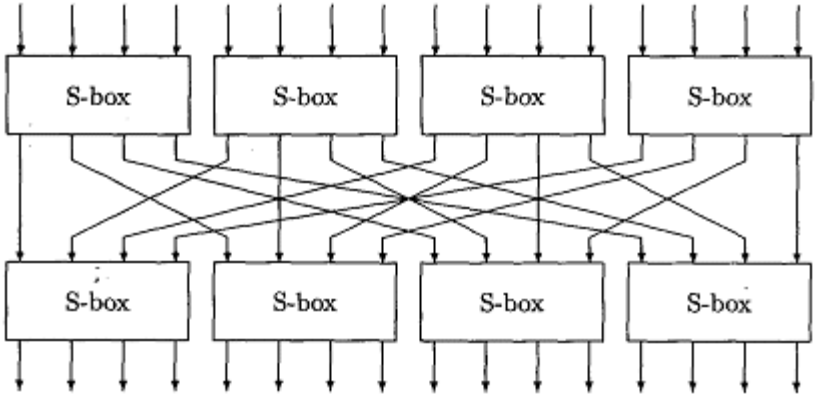
	0	1	2	3
0	3	8	15	1
1	10	6	5	11
2	14	13	4	2
3	7	0	9	12

(b)

Figure 8.3: A 4-bit S-box (a) An S-box in binary. (b) The same S-box in decimal.

Substitution Boxes

- Some modern ciphers use S-boxes that are connected to each other



https://www2.dmst.aueb.gr/dds/secimp/crypto/sp.htm

Data Encryption Standard (DES)

- Symmetric encryption algorithm
- Developed by IBM and adopted by NIST in 1977
- Encrypts 64-bit blocks using 56-bit keys
  - Relies heavily on S-boxes for security
- Adopted as an official FIPS (Federal Information Processing Standard)
  - Adopted for securing government data

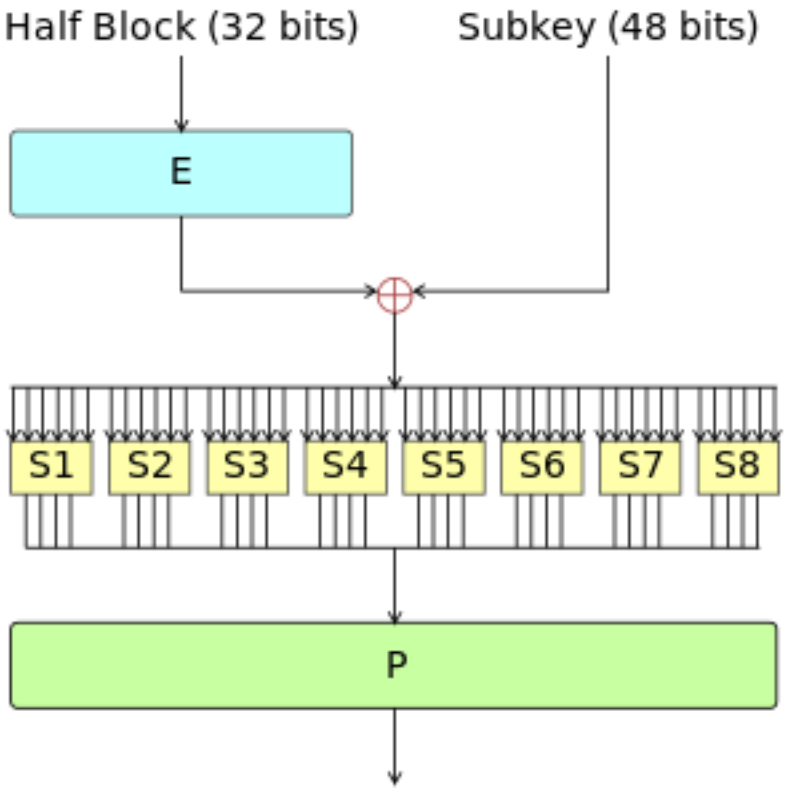
Data Encryption Standard (DES)

- DES is a combination of two fundamental building blocks of encryption:
  - substitution and transposition
- These techniques are repeated one on top of the other, for a total of 16 cycles

Data Encryption Standard (DES)

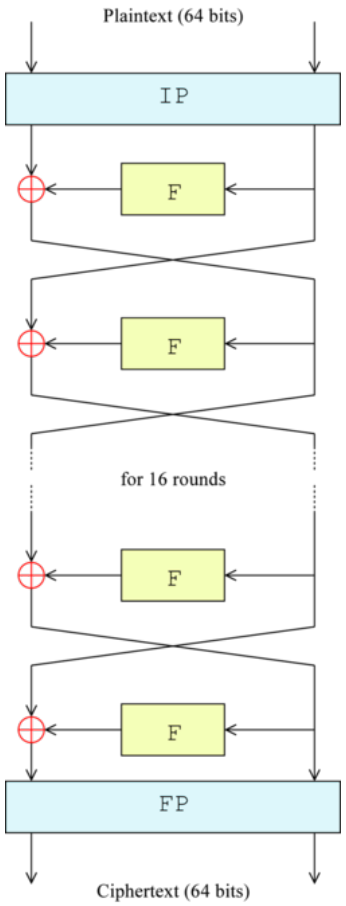
- Each round includes:
  - Substitution:
    - replacing blocks of bits
  - Permutation:
    - Shuffling the bits
  - Transformation:
    - Mingling bits from the key

DES Single Round (Feistel Function)



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

DES Algorithm



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

Data Encryption Standard (DES)

- Symmetric encryption algorithm
- Developed by IBM and adopted by NIST in 1977
- Encrypts 64-bit blocks using 56-bit keys
  - Relies heavily on S-boxes for securtity
- Small key space makes exhaustive search attack feasible since late 90s
  - **Not secure – should not be used!**
  - So what can be used instead?

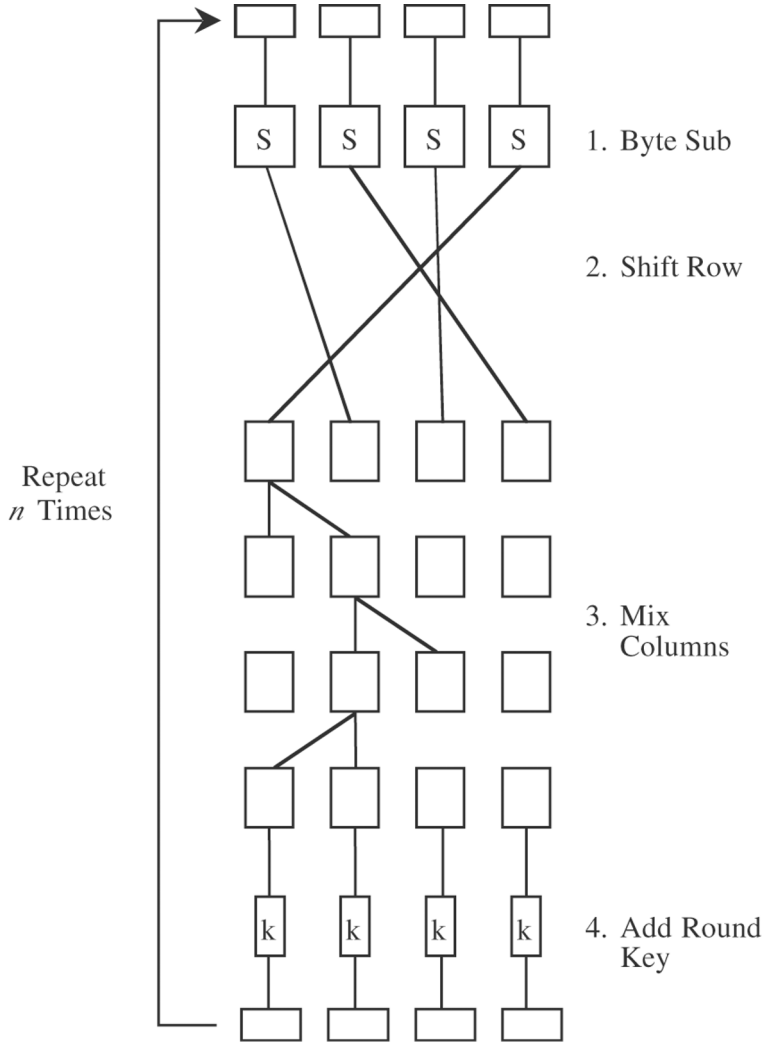
Advanced Encryption Standard (AES)

- Symmetric block cipher
- Developed in 1999 by independent Dutch cryptographers
- Selected by NIST in 2001 through open international competition and public discussion

Advanced Encryption Standard (AES)

- 128-bit blocks and several possible key lengths: 128, 192 and 256 bits
- Exhaustive search attack not currently possible
- AES-256 is the symmetric encryption algorithm of choice
  - Still in common use

AES: Advanced Encryption System



DES vs. AES

	DES	AES
Date designed	1976	1999
Block size	64 bits	128 bits
Key length	56 bits (effective length); up to 112 bits with multiple keys	128, 192, 256 (and possibly more) bits
Operations	16 rounds	10, 12, 14 (depending on key length); can be increased
Encryption primitives	Substitution, permutation	Substitution, shift, bit mixing
Cryptographic primitives	Confusion, diffusion	Confusion, diffusion
Design	Open	Open
Design rationale	Closed	Open
Selection process	Secret	Secret, but open public comments and criticisms invited
Source	IBM, enhanced by NSA	Independent Dutch cryptographers

Questions?