

IS 2150 / TEL 2810

Introduction to Security



James Joshi
Associate Professor, SIS

Lecture 7
Oct 4, 2011

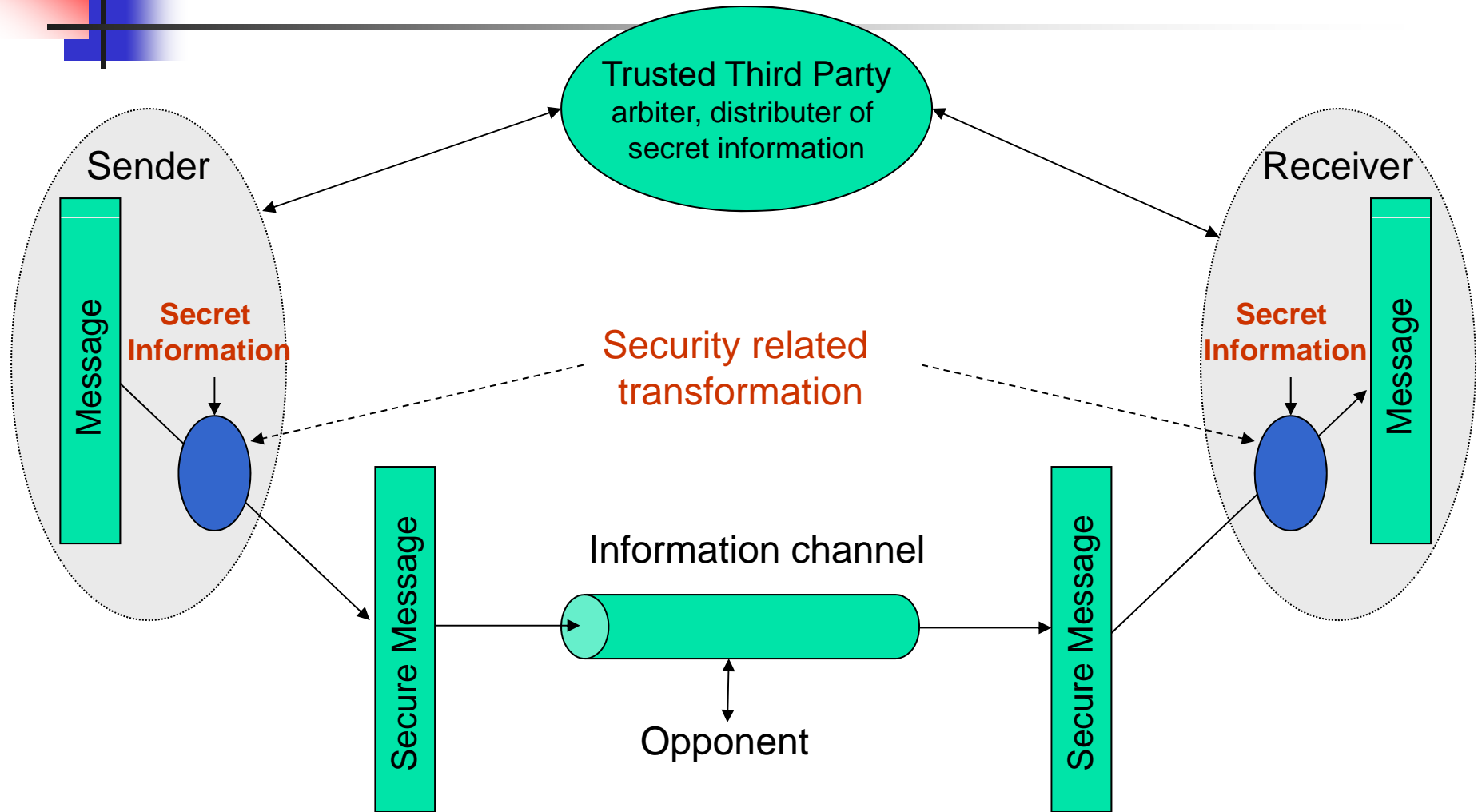
Basic Cryptography
Network Security



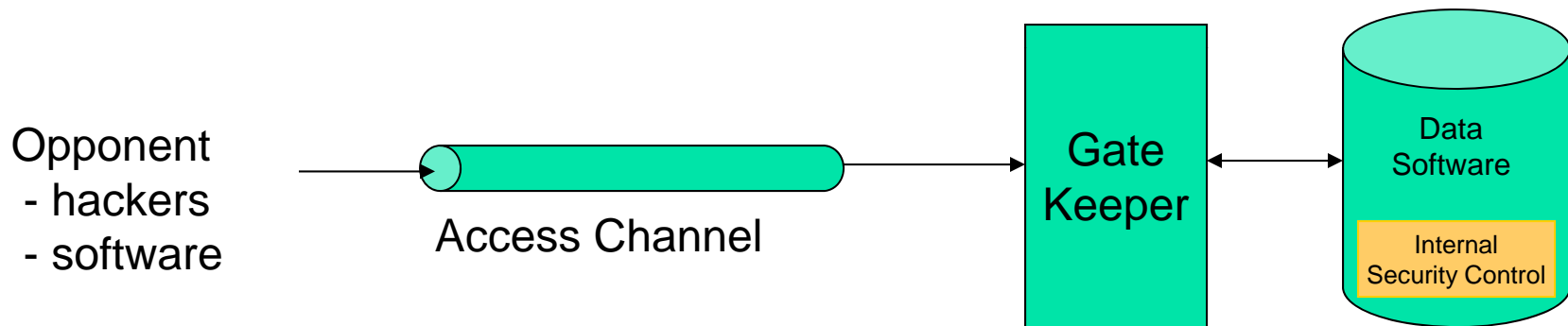
Objectives

- Understand/explain/employ the basic cryptographic techniques
 - Review the basic number theory used in cryptosystems
 - Classical system
 - Public-key system
 - Some crypto analysis
 - Message digest

Secure Information Transmission (network security model)



Security of Information Systems (Network access model)



Gatekeeper – firewall or equivalent, password-based login

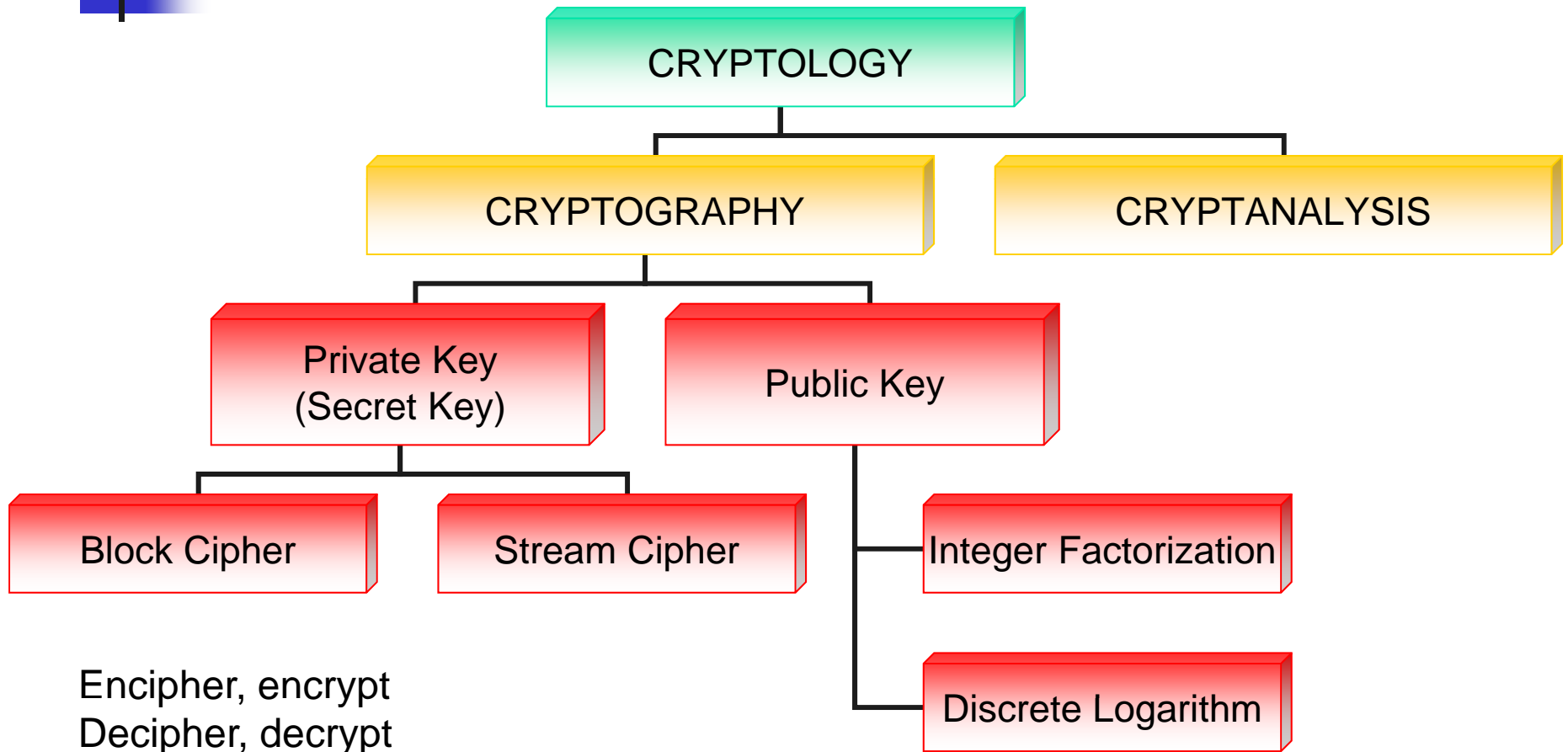
Internal Security Control – Access control, Logs, audits, virus scans etc.



Issues in Network security

- Distribution of secret information to enable secure exchange of information
- Effect of communication protocols needs to be considered
- Encryption *if used cleverly and correctly*, can provide several of the security services
- Physical and logical placement of security mechanisms
- Countermeasures need to be considered

Cryptology





Elementary Number Theory

- Natural numbers $N = \{1, 2, 3, \dots\}$
- Whole numbers $W = \{0, 1, 2, 3, \dots\}$
- Integers $Z = \{\dots, -2, -1, 0, 1, 2, 3, \dots\}$
- Divisors
 - A number b is said to divide a if $a = mb$ for some m where $a, b, m \in Z$
 - We write this as $b \mid a$



Divisors

- Some common properties
 - If $a \mid 1$, $a = +1$ or -1
 - If $a \mid b$ and $b \mid a$ then $a = +b$ or $-b$
 - Any $b \in \mathbb{Z}$ divides 0 if $b \neq 0$
 - If $b \mid g$ and $b \mid h$ then $b \mid (mg + nh)$ where $b, m, n, g, h \in \mathbb{Z}$
- Examples:
 - The positive divisors of 42 are ?
 - $3 \mid 6$ and $3 \mid 21 \Rightarrow 3 \mid 21m + 6n$ for $m, n \in \mathbb{Z}$



Prime Numbers

- An integer p is said to be a prime number if its only positive divisors are 1 and itself
 - 2, 3, 7, 11, ..
- Any integer can be expressed as a *unique* product of prime numbers raised to positive integral powers
- Examples
 - $7569 = 3 \times 3 \times 29 \times 29 = 3^2 \times 29^2$
 - $5886 = 2 \times 27 \times 109 = 2 \times 3^3 \times 109$
 - $4900 = 7^2 \times 5^2 \times 2^2$
 - $100 = ?$
 - $250 = ?$
- This process is called *Prime Factorization*



Greatest common divisor (GCD)

- Definition: Greatest Common Divisor
 - This is the largest divisor of *both* a and b
- Given two integers a and b , the positive integer c is called their GCD or greatest common divisor if and only if
 - $c \mid a$ and $c \mid b$
 - Any divisor of both a and b also divides c
- Notation: $\gcd(a, b) = c$
- Example: $\gcd(49, 63) = ?$



Relatively Prime Numbers

- Two numbers are said to be relatively prime if their gcd is 1
 - Example: 63 and 22 are relatively prime
- How do you determine if two numbers are relatively prime?
 - Find their GCD or
 - Find their prime factors
 - If they do not have a common prime factor other than 1, they are relatively prime
 - Example: $63 = 9 \times 7 = 3^2 \times 7$ and $22 = 11 \times 2$



The modulo operation

- What is $27 \bmod 5$?
- Definition
 - Let a, r, m be integers and let $m > 0$
 - We write $a \equiv r \pmod{m}$ if m divides $r - a$ (or $a - r$) and $0 \leq r < m$
 - m is called ?
 - r is called ?
 - Note: $a = m \cdot q + r$; what is q ?



Modular Arithmetic

- We say that $a \equiv b \pmod{m}$ if $m \mid a - b$
 - Read as: a is congruent to b modulo m
 - m is called the modulus
 - Example: $27 \equiv 2 \pmod{5}$
 - Example: $27 \equiv 7 \pmod{5}$ and $7 \equiv 2 \pmod{5}$
- $a \equiv b \pmod{m} \Rightarrow b \equiv a \pmod{m}$
 - Example: $2 \equiv 27 \pmod{5}$
- We usually consider the *smallest positive remainder* which is called the **residue**



Modulo Operation

- The modulo operation “reduces” the infinite set of integers to a finite set
- Example: modulo 5 operation
 - We have five sets
 - $\{\dots, -10, -5, 0, 5, 10, \dots\} \Rightarrow a \equiv 0 \pmod{5}$
 - $\{\dots, -9, -4, 1, 6, 11, \dots\} \Rightarrow a \equiv 1 \pmod{5}$
 - $\{\dots, -8, -3, 2, 7, 12, \dots\} \Rightarrow a \equiv 2 \pmod{5}$, etc.
 - The set of residues of integers modulo 5 has five elements $\{0, 1, 2, 3, 4\}$ and is denoted Z_5 .



Modulo Operation

- Properties

- $[(a \bmod n) + (b \bmod n)] \bmod n = (a + b) \bmod n$
- $[(a \bmod n) - (b \bmod n)] \bmod n = (a - b) \bmod n$
- $[(a \bmod n) \times (b \bmod n)] \bmod n = (a \times b) \bmod n$
- $(-1) \bmod n = n - 1$
 - (Using $b = q.n + r$, with $b = -1$, $q = -1$ and $r = n-1$)



Brief History

- All encryption algorithms from BC till 1976 were secret key algorithms
 - Also called private key algorithms or symmetric key algorithms
 - Julius Caesar used a substitution cipher
 - Widespread use in World War II (enigma)
- Public key algorithms were introduced in 1976 by Whitfield Diffie and Martin Hellman



Cryptosystem

- $(\mathcal{E}, \mathcal{D}, \mathcal{M}, \mathcal{K}, \mathcal{C})$
 - \mathcal{E} set of encryption functions
 $e: \mathcal{M} \times \mathcal{K} \rightarrow \mathcal{C}$
 - \mathcal{D} set of decryption functions
 $d: \mathcal{C} \times \mathcal{K} \rightarrow \mathcal{M}$
 - \mathcal{M} set of plaintexts
 - \mathcal{K} set of keys
 - \mathcal{C} set of ciphertexts



Example

- Cæsar cipher
 - $\mathcal{M} = \{ \text{sequences of letters} \}$
 - $\mathcal{K} = \{ i \mid i \text{ is an integer and } 0 \leq i \leq 25 \}$
 - $\mathcal{E} = \{ E_k \mid k \in \mathcal{K} \text{ and for all letters } m, \mathcal{E}(m) = (m + k) \bmod 26 \}$
 - $\mathcal{D} = \{ D_k \mid k \in \mathcal{K} \text{ and for all letters } c, \mathcal{D}(c) = (26 + c - k) \bmod 26 \}$
 - $\mathcal{C} = \mathcal{M}$



Cæsar cipher

- Let $k = 9$, $m = \text{"VELVET"} (21\ 4\ 11\ 21\ 4\ 19)$
 - $E_k(m) = (30\ 13\ 20\ 30\ 13\ 28) \bmod 26$
 $= \text{"4\ 13\ 20\ 4\ 13\ 2"} = \text{"ENUENC"}$
 - $D_k(m) = (26 + c - k) \bmod 26$
 $= (21\ 30\ 37\ 21\ 30\ 19) \bmod 26$
 $= \text{"21\ 4\ 11\ 21\ 4\ 19"} = \text{"VELVET"}$

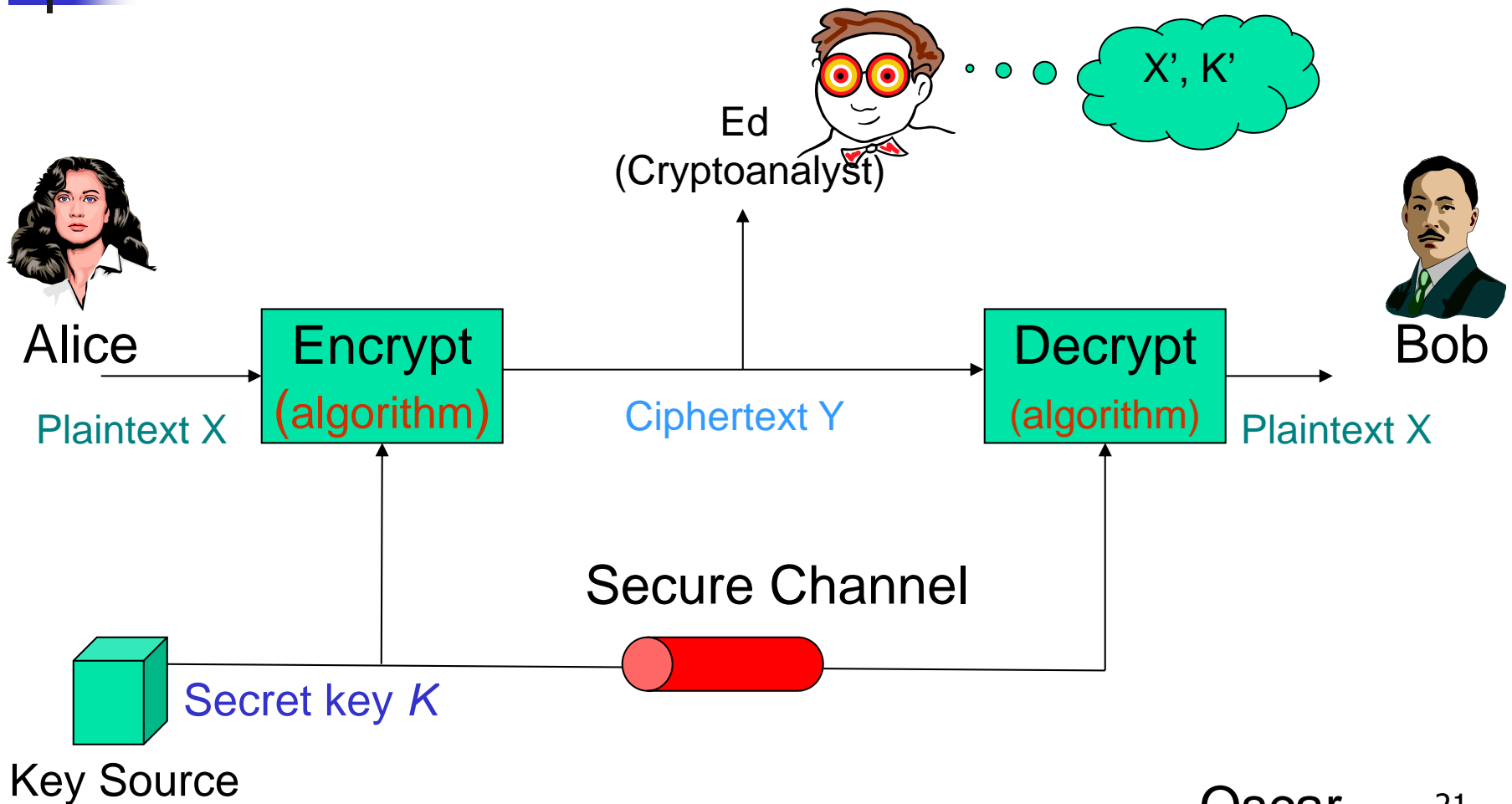
| | | | | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|----|----|----|
| A | B | C | D | E | F | G | H | I | J | K | L | M |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| N | O | P | Q | R | S | T | U | V | W | X | Y | Z |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |



Attacks

- *Ciphertext only:*
 - adversary has only Y ;
 - goal ?
- *Known plaintext:*
 - adversary has X, Y ;
 - goal ?
- *Chosen plaintext:*
 - adversary gets a specific plaintext enciphered;
 - goal ?

Classical Cryptography





Classical Cryptography

- Sender, receiver share common key
 - Keys may be the same, or trivial to derive from one another
 - Sometimes called *symmetric cryptography*
- Two basic types
 - Transposition ciphers
 - Substitution ciphers
- Product ciphers
 - Combinations of the two basic types



Classical Cryptography

- $y = E_k(x)$: Ciphertext \rightarrow Encryption
- $x = D_k(y)$: Plaintext \rightarrow Decryption
- k = encryption and decryption key
- The functions $E_k()$ and $D_k()$ must be inverses of one another
 - $E_k(D_k(y)) = ?$
 - $D_k(E_k(x)) = ?$
 - $E_k(D_k(x)) = ?$



Transposition Cipher

- Rearrange letters in plaintext to produce ciphertext
- Example (Rail-Fence Cipher)
 - Plaintext is "HELLO WORLD"
 - Rearrange as
 - HLOOL
 - ELWRD
 - Ciphertext is HLOOL ELWRD



Attacking the Cipher

- Anagramming
 - If 1-gram frequencies match English frequencies, but other n -gram frequencies do not, probably transposition
 - Rearrange letters to form n -grams with highest frequencies



Example

- Ciphertext: HLOOLELWRD
- Frequencies of 2-grams beginning with H
 - HE 0.0305
 - HO 0.0043
 - HL, HW, HR, HD < 0.0010
- Frequencies of 2-grams ending in H
 - WH 0.0026
 - EH, LH, OH, RH, DH ≤ 0.0002
- Implies E follows H



Example

- Arrange so that H and E are adjacent

HE

LL

OW

OR

LD

- Read off across, then down, to get original plaintext



Substitution Ciphers

- Change characters in plaintext to produce ciphertext
- Example (Cæsar cipher)
 - Plaintext is HELLO WORLD;
 - Key is 3, usually written as letter 'D'
 - Ciphertext is KHOOR ZRUOG



Attacking the Cipher

- Brute Force: Exhaustive search
 - If the key space is small enough, try all possible keys until you find the right one
 - Cæsar cipher has 26 possible keys
- Statistical analysis
 - Compare to 1-gram model of English



Statistical Attack

- Ciphertext is KHOOR ZRUOG
- Compute frequency of each letter in ciphertext:

| | | | | | | | |
|---|-----|---|-----|---|-----|---|-----|
| G | 0.1 | H | 0.1 | K | 0.1 | O | 0.3 |
| R | 0.2 | U | 0.1 | Z | 0.1 | | |
- Apply 1-gram model of English
 - Frequency of characters (1-grams) in English is on next slide



Character Frequencies (Denning)

| | | | | | | | |
|---|-------|---|-------|---|-------|---|-------|
| a | 0.080 | h | 0.060 | n | 0.070 | t | 0.090 |
| b | 0.015 | i | 0.065 | o | 0.080 | u | 0.030 |
| c | 0.030 | j | 0.005 | p | 0.020 | v | 0.010 |
| d | 0.040 | k | 0.005 | q | 0.002 | w | 0.015 |
| e | 0.130 | l | 0.035 | r | 0.065 | x | 0.005 |
| f | 0.020 | m | 0.030 | s | 0.060 | y | 0.020 |
| g | 0.015 | | | | | z | 0.002 |



Statistical Analysis

- $f(c)$ frequency of character c in ciphertext
- $\varphi(i)$:
 - correlation of frequency of letters in ciphertext with corresponding letters in English, assuming key is i
 - $\varphi(i) = \sum_{0 \leq c \leq 25} f(c)p(c-i)$
 - so here,
$$\varphi(i) = 0.1p(6-i) + 0.1p(7-i) + 0.1p(10-i) + 0.3p(14-i) + 0.2p(17-i) + 0.1p(20-i) + 0.1p(25-i)$$
 - $p(x)$ is frequency of character x in English
 - Look for maximum correlation!

Correlation: $\varphi(i)$ for $0 \leq i \leq 25$

| i | $\varphi(i)$ | i | $\varphi(i)$ | i | $\varphi(i)$ | i | $\varphi(i)$ |
|-----|--------------|-----|--------------|-----|--------------|-----|--------------|
| 0 | 0.0482 | 7 | 0.0442 | 13 | 0.0520 | 19 | 0.0315 |
| 1 | 0.0364 | 8 | 0.0202 | 14 | 0.0535 | 20 | 0.0302 |
| 2 | 0.0410 | 9 | 0.0267 | 15 | 0.0226 | 21 | 0.0517 |
| 3 | 0.0575 | 10 | 0.0635 | 16 | 0.0322 | 22 | 0.0380 |
| 4 | 0.0252 | 11 | 0.0262 | 17 | 0.0392 | 23 | 0.0370 |
| 5 | 0.0190 | 12 | 0.0325 | 18 | 0.0299 | 24 | 0.0316 |
| 6 | 0.0660 | | | | | 25 | 0.0430 |



The Result

- Ciphertext is KHOOR ZRUOG
- Most probable keys, based on φ :
 - $i = 6, \varphi(i) = 0.0660$
 - plaintext EBIIL TLOLA (How?)
 - $i = 10, \varphi(i) = 0.0635$
 - plaintext AXEEH PHKEW (How?)
 - $i = 3, \varphi(i) = 0.0575$
 - plaintext HELLO WORLD (How?)
 - $i = 14, \varphi(i) = 0.0535$
 - plaintext WTAAD LDGAS
- Only English phrase is for $i = 3$
 - That's the key (3 or 'D')



Cæsar's Problem

- Key is too short
 - Can be found by exhaustive search
 - Statistical frequencies not concealed well
 - They look too much like regular English letters
- So make it longer
 - Multiple letters in key
 - Idea is to smooth the statistical frequencies to make cryptanalysis harder



Vigenère Cipher

- Like Cæsar cipher, but use a phrase
- Example
 - Message THE BOY HAS THE BALL
 - Key VIG
 - Encipher using Cæsar cipher for each letter:

| | |
|--------|------------------|
| key | VIGVIGVIGVIGVIGV |
| plain | THEBOYHASTHEBALL |
| cipher | OPKWECIYOPKWIRG |



Relevant Parts of Tableau

| | | | |
|----------|----------|----------|----------|
| | <i>G</i> | <i>I</i> | <i>V</i> |
| <i>A</i> | <i>G</i> | <i>I</i> | <i>V</i> |
| <i>B</i> | <i>H</i> | <i>J</i> | <i>W</i> |
| <i>E</i> | <i>K</i> | <i>M</i> | <i>Z</i> |
| <i>H</i> | <i>N</i> | <i>P</i> | <i>C</i> |
| <i>L</i> | <i>R</i> | <i>T</i> | <i>G</i> |
| <i>O</i> | <i>U</i> | <i>W</i> | <i>J</i> |
| <i>S</i> | <i>Y</i> | <i>A</i> | <i>N</i> |
| <i>T</i> | <i>Z</i> | <i>B</i> | <i>O</i> |
| <i>Y</i> | <i>E</i> | <i>H</i> | <i>T</i> |

- Tableau with relevant rows, columns only
- Example encipherments:
 - key *V*, letter *T*: follow *V* column down to *T* row (giving "O")
 - Key *I*, letter *H*: follow *I* column down to *H* row (giving "P")



Useful Terms

- *period*: length of key
 - In earlier example, period is 3
- *tableau*: table used to encipher and decipher
 - Vigenere cipher has key letters on top, plaintext letters on the left
- *polyalphabetic*: the key has several different letters
 - Caesar cipher is monoalphabetic

Attacking the Cipher

- Key to attacking vigenère cipher
 - determine the key length
 - If the keyword is n , then the cipher consists of n monoalphabetic substitution ciphers

| | |
|--------|---------------------|
| key | VIGVIGVIGVIGVIGV |
| plain | THEBOYHASTHEBALL |
| cipher | O [redacted] PKWIRG |

| | |
|--------|--------------------------------|
| key | DECEPTIVEDECEPTIVEDECEPTIVE |
| plain | WEAREDISCOVEREDSAVEYOURSELF |
| cipher | ZICV [redacted] TWAVZHCQYGLMGJ |



One-Time Pad

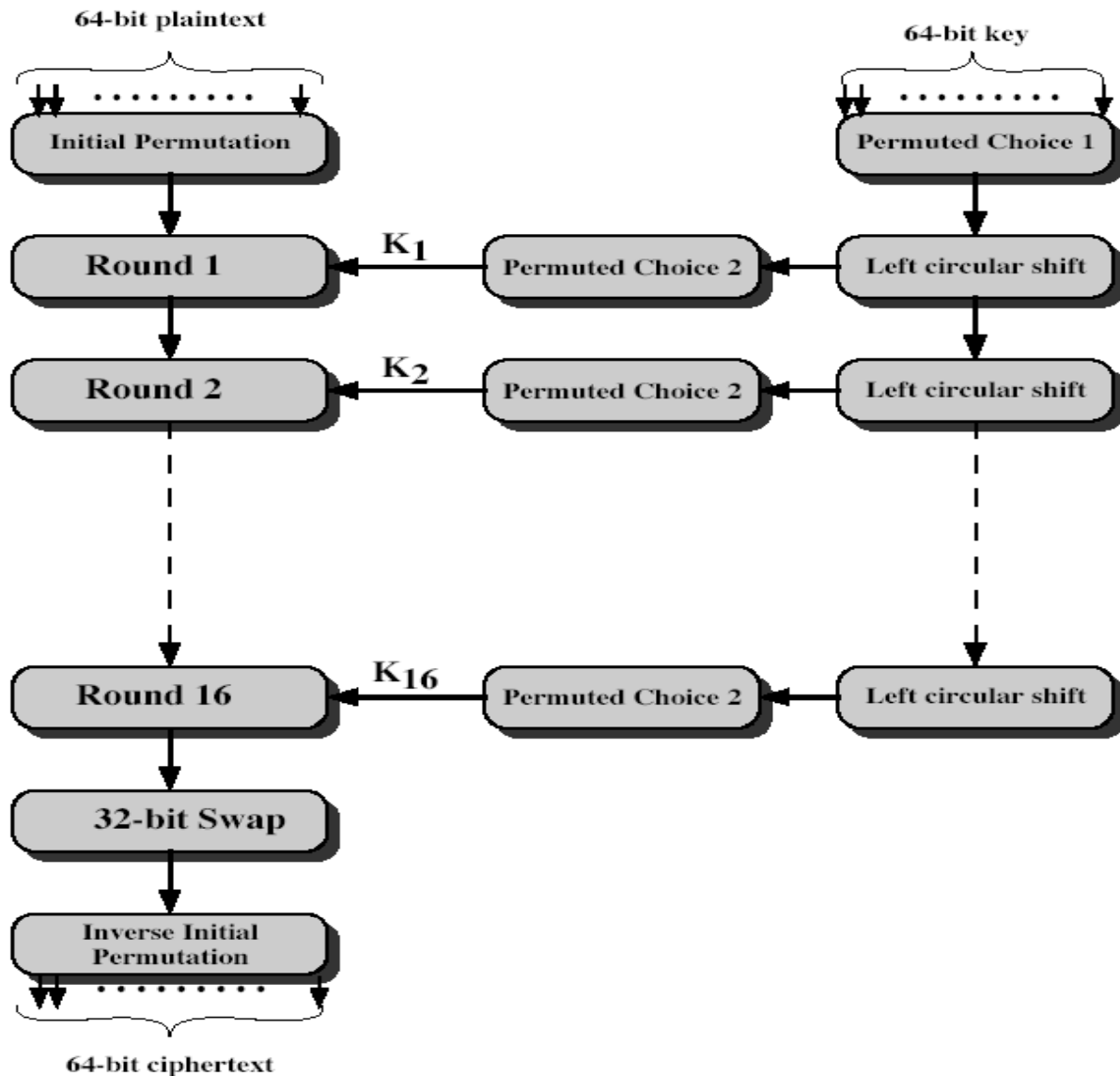
- A Vigenère cipher with a random key at least as long as the message
 - Provably unbreakable; Why?
 - Consider ciphertext DXQR. Equally likely to correspond to
 - plaintext DOIT (key AJIY) and
 - plaintext DONT (key AJDY) and any other 4 letters
 - Warning: keys *must* be random, or you can attack the cipher by trying to regenerate the key



Overview of the DES

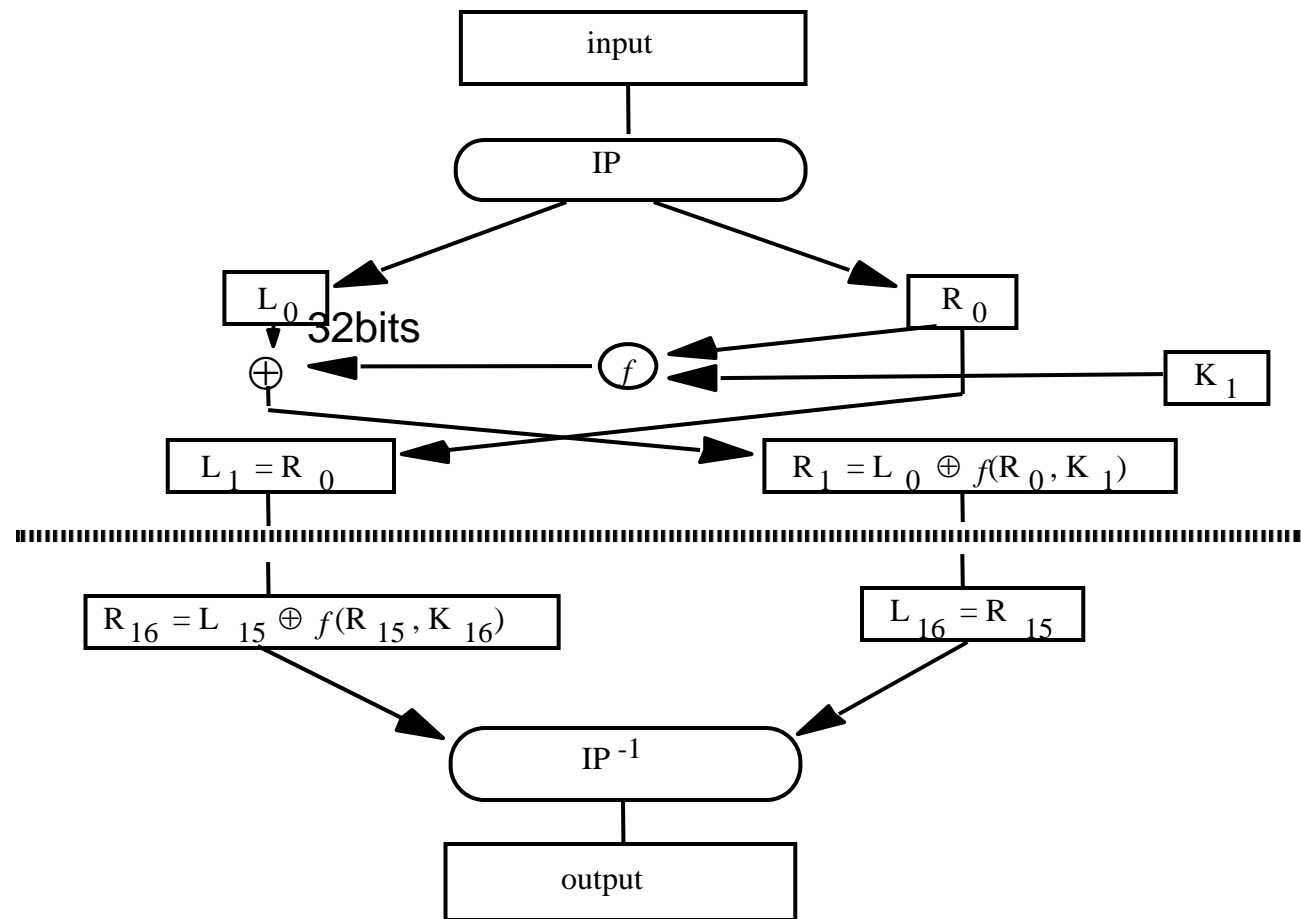
- A block cipher:
 - encrypts blocks of 64 bits using a 64 bit key
 - outputs 64 bits of ciphertext
 - A **product cipher**
 - performs both substitution and transposition (permutation) on the bits
 - basic unit is the bit
- Cipher consists of 16 rounds (iterations) each with a round key generated from the user-supplied key

DES

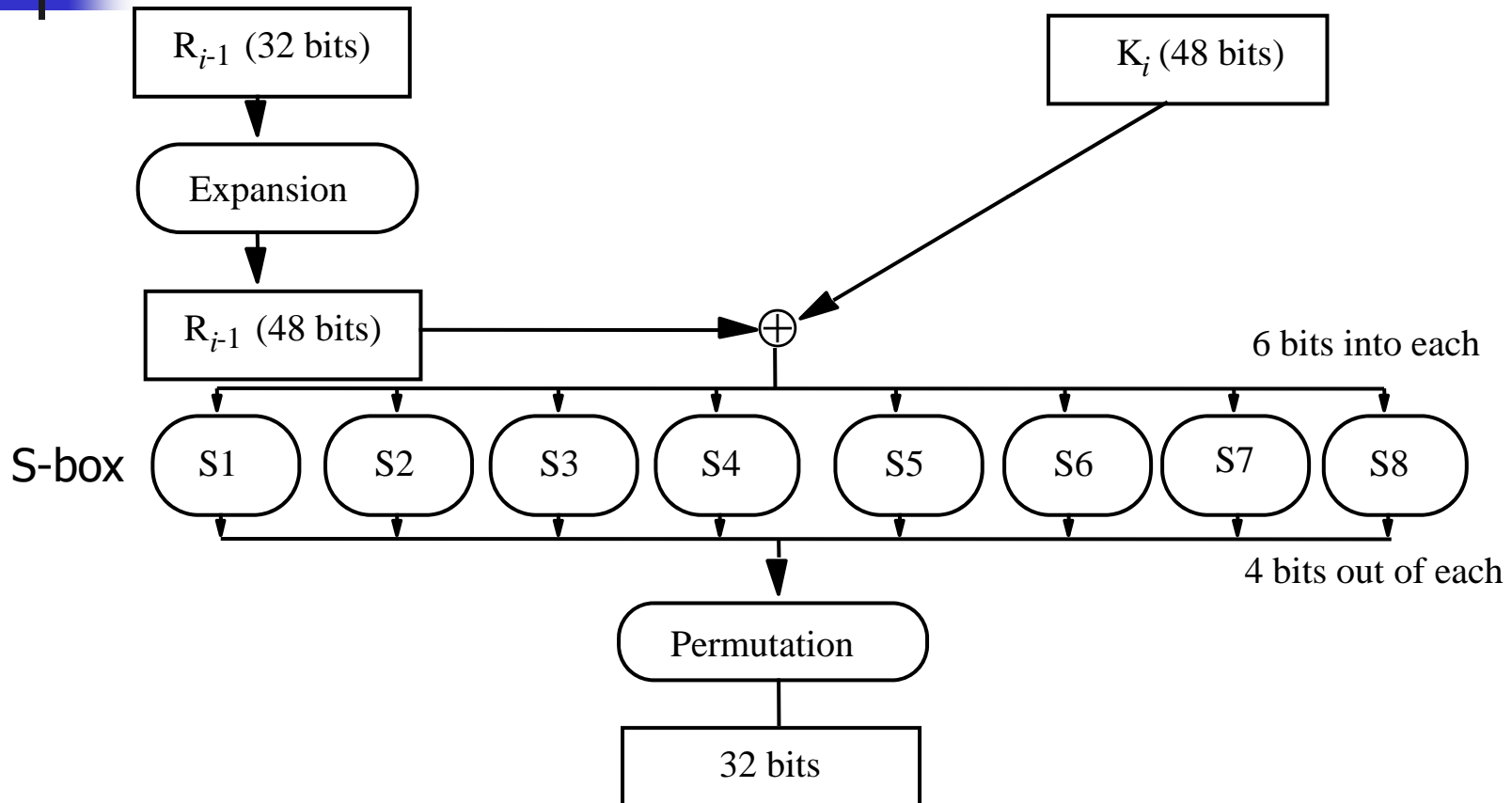


- Round keys are 48 bits each
 - Extracted from 64 bits
 - Permutation applied
- Deciphering involves using round keys in reverse

Encipherment



The f Function



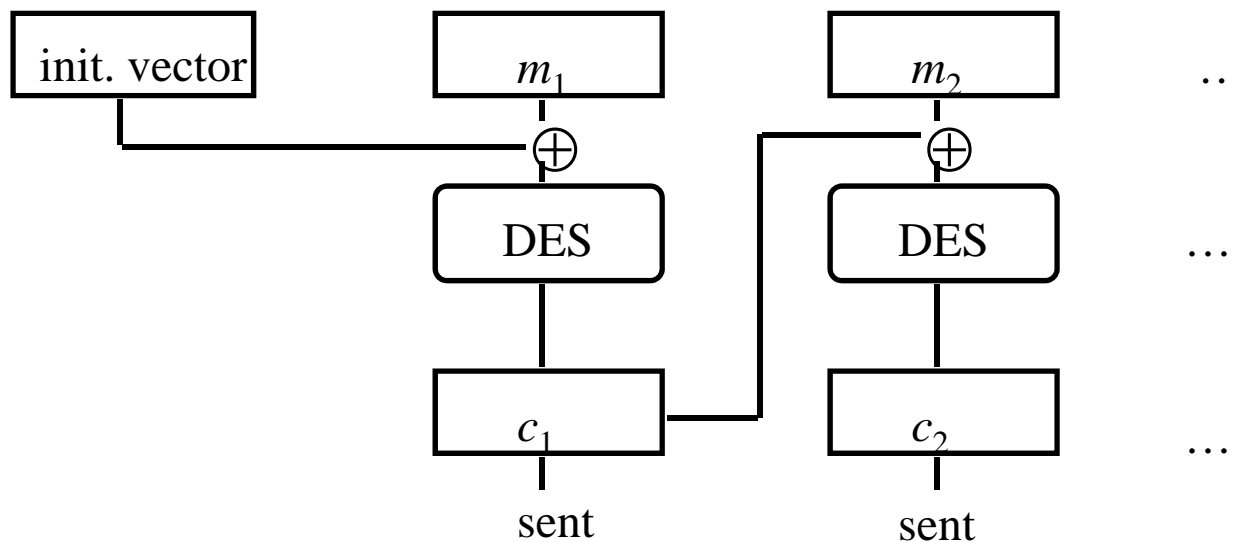


Controversy

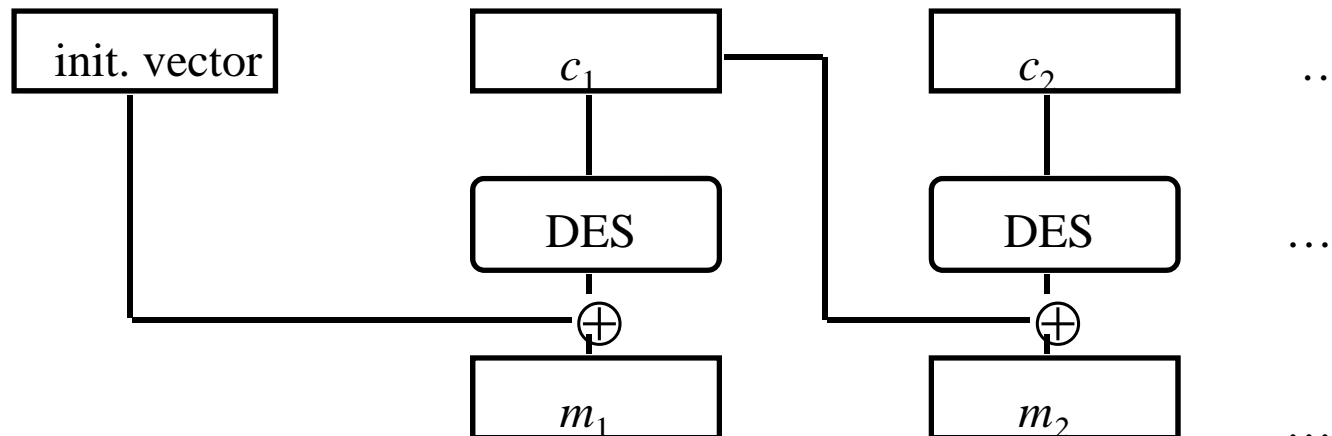
- Considered too weak
 - Design to break it using 1999 technology published
 - Design decisions not public
 - S-boxes may have backdoors
- Several other weaknesses found
 - Mainly related to keys

DES Modes

- Electronic Code Book Mode (ECB):
 - Encipher each block independently
- Cipher Block Chaining Mode (CBC)
 - XOR each block with previous ciphertext block
 - Uses an initialization vector for the first one



CBC Mode Decryption



- CBC has self healing property
 - If one block of ciphertext is altered, the error propagates for at most two blocks




Self-Healing Property






- Initial message

- 3231343336353837 3231343336353837
3231343336353837 3231343336353837

- Received as (underlined 4c should be 4b)

- ef7c  b2b4ce6f3b f6266e3a97af0e2c
746ab9a6308f4256 33e60b451b09603d

- Which decrypts to

-  3231  3336353837
3231  3336353837
 -  ined; p  ntext "heals"



Public Key Cryptography

- Two keys
 - *Private key* known only to individual
 - *Public key* available to anyone
- Idea
 - Confidentiality:
 - encipher using public key,
 - decipher using private key
 - Integrity/authentication:
 - encipher using private key,
 - decipher using public one



Requirements

1. Given the appropriate key, it must be computationally easy to encipher or decipher a message
2. It must be computationally infeasible to derive the private key from the public key
3. It must be computationally infeasible to determine the private key from a chosen plaintext attack



Diffie-Hellman

- Compute a common, shared key
 - Called a *symmetric key exchange protocol*
- Based on discrete logarithm problem
 - Given integers n and g and prime number p , compute k such that $n = g^k \bmod p$
 - Solutions known for small p
 - Solutions computationally infeasible as p grows large – hence, choose large p



Algorithm

- Constants known to participants
 - prime p ; integer g other than 0, 1 or $p-1$
- Alice: (private = k_A , public = K_A)
- Bob: (private = k_B , public = K_B)
 - $K_A = g^{k_A} \text{ mod } p$
 - $K_B = g^{k_B} \text{ mod } p$
- To communicate with Bob,
 - Alice computes $S_{A,B} = K_B^{k_A} \text{ mod } p$
- To communicate with Alice,
 - Bob computes $S_{B,A} = K_A^{k_B} \text{ mod } p$
- $S_{A,B} = S_{B,A} ?$



Example

- Assume $p = 53$ and $g = 17$
- Alice chooses $k_A = 5$
 - Then $K_A = 17^5 \bmod 53 = 40$
- Bob chooses $k_B = 7$
 - Then $K_B = 17^7 \bmod 53 = 6$
- Shared key:
 - $K_B^{k_A} \bmod p = 6^5 \bmod 53 = 38$
 - $K_A^{k_B} \bmod p = 40^7 \bmod 53 = 38$

Exercise:

Let $p = 5$, $g = 3$
 $k_A = 4$, $k_B = 3$

$K_A = ?$, $K_B = ?$,
 $S = ?$,



RSA

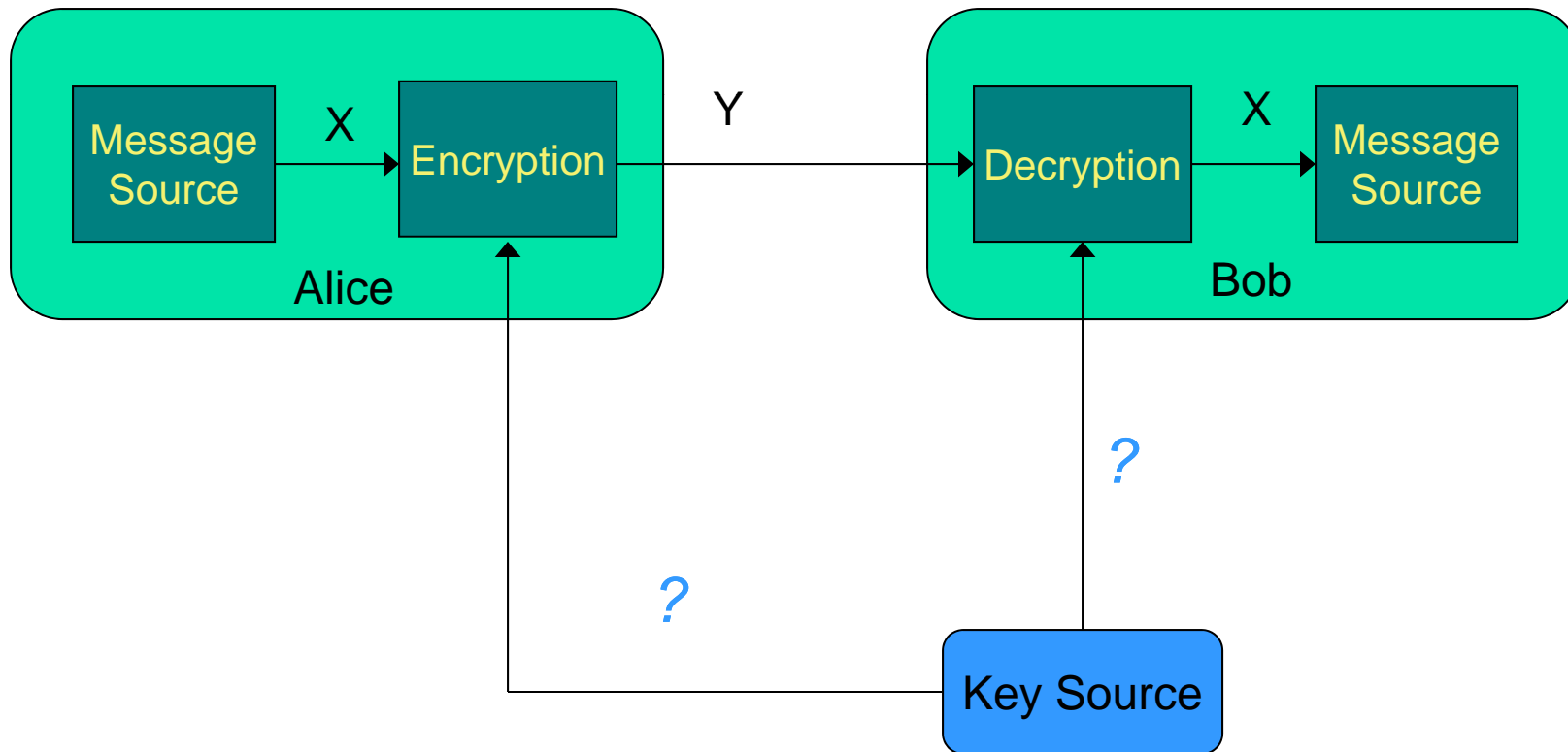
- Relies on the difficulty of determining the number of numbers relatively prime to a large integer n
- **Totient** function $\phi(n)$
 - Number of + integers less than n and relatively prime to n
- Example: $\phi(10) = 4$
 - What are the numbers relatively prime to 10?
- $\phi(77)$?
- $\phi(p)$? When p is a prime number
- $\phi(pq)$? When p and q are prime numbers



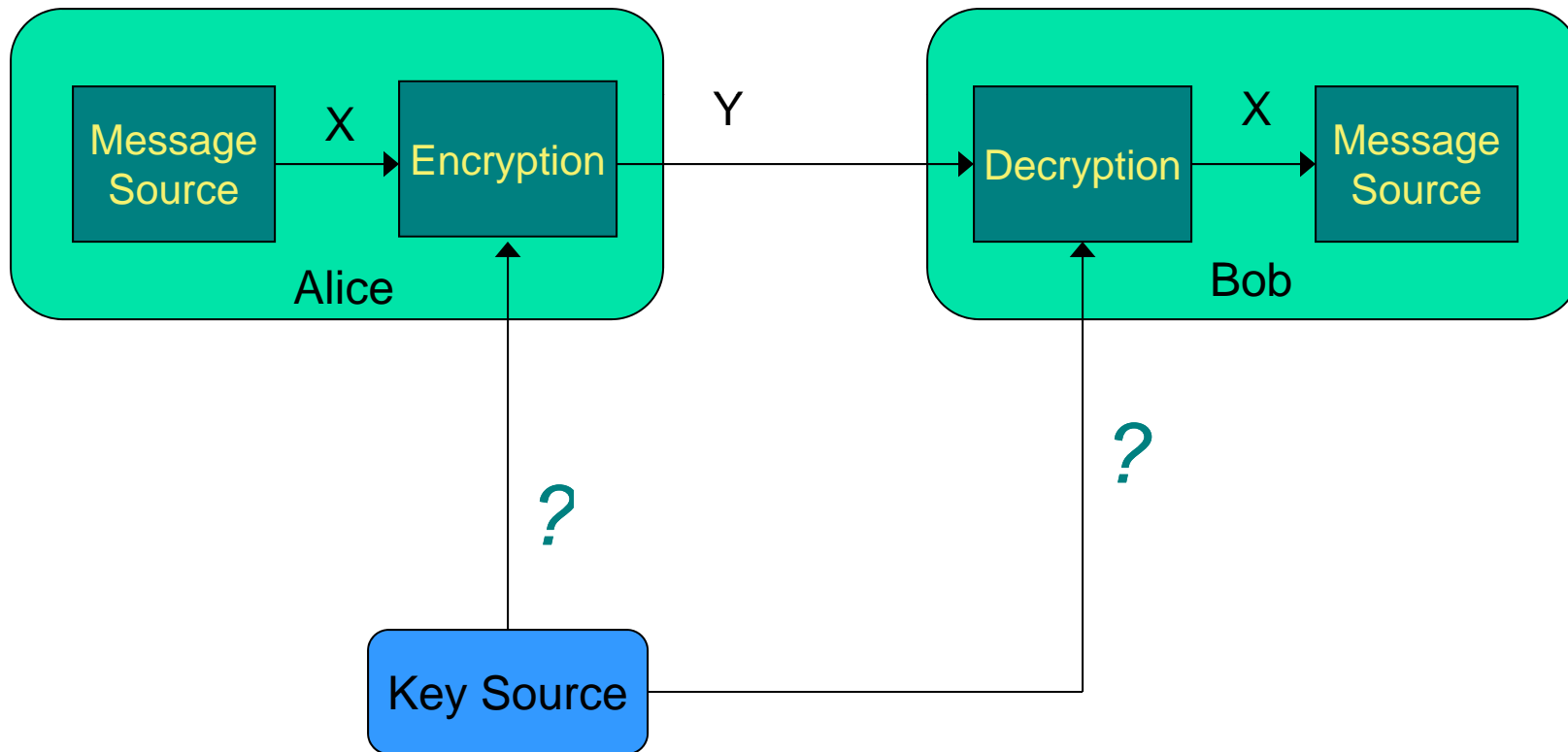
Algorithm

- Choose two large prime numbers p, q
 - Let $n = pq$; then $\phi(n) = (p-1)(q-1)$
 - Choose $e < n$ relatively prime to $\phi(n)$.
 - Compute d such that $ed \bmod \phi(n) = 1$
 - Public key: (e, n) ;
 - private key: d (or (d, n))
- Encipher: $c = m^e \bmod n$
- Decipher: $m = c^d \bmod n$

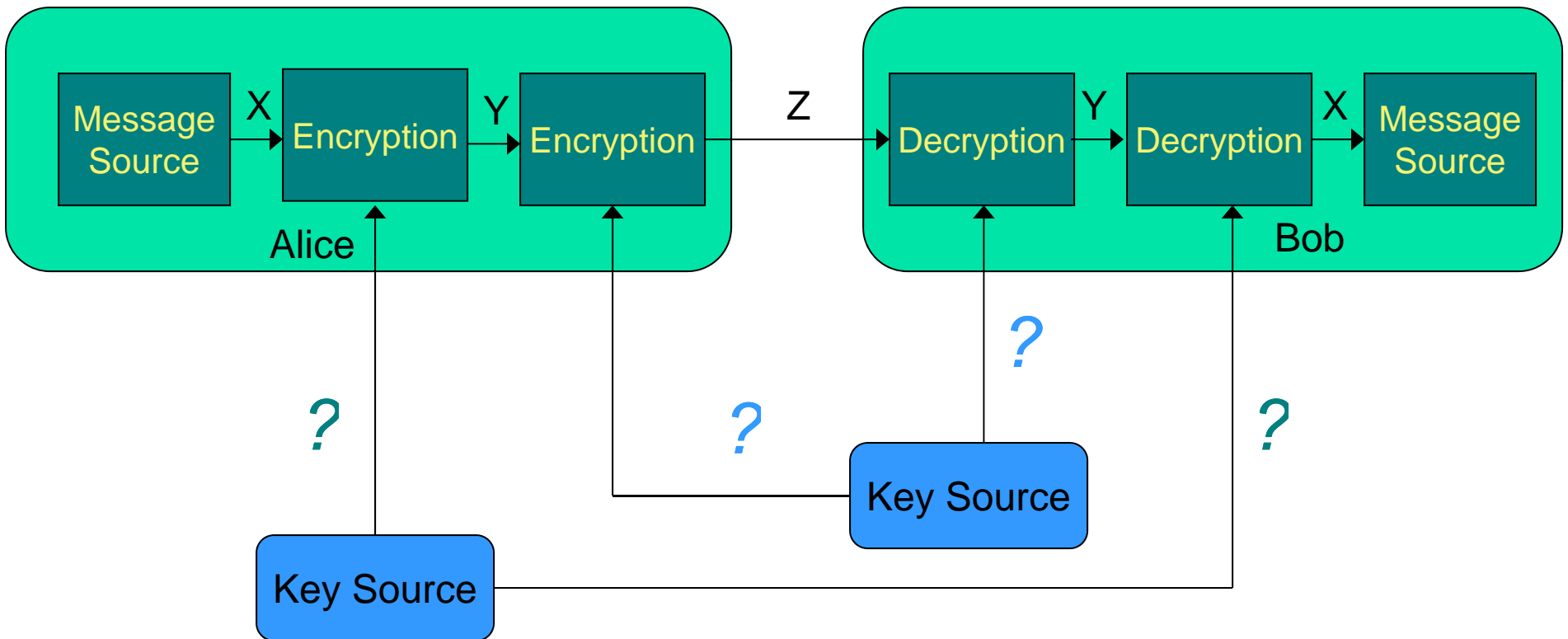
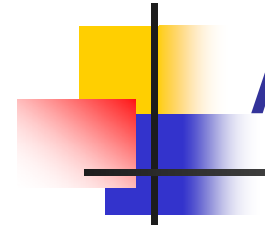
Confidentiality using RSA



Authentication using RSA



Confidentiality + Authentication





Warnings

- Encipher message in blocks considerably larger than the examples here
 - If 1 character per block, RSA can be broken using statistical attacks (just like classical cryptosystems)
 - Attacker cannot alter letters, but can rearrange them and alter message meaning
 - Example: reverse enciphered message: ON to get NO



Cryptographic Checksums

- Mathematical function to generate a set of k bits from a set of n bits (where $k \leq n$).
 - k is smaller than n except in unusual circumstances
 - Keyed CC: requires a cryptographic key
$$h = C_{key}(M)$$
 - Keyless CC: requires no cryptographic key
 - Message Digest or One-way Hash Functions
$$h = H(M)$$
- Can be used for message authentication
 - Hence, also called Message Authentication Code (MAC)



Mathematical characteristics

- Every bit of the message digest function potentially influenced by every bit of the function's input
- If any given bit of the function's input is changed, every output bit has a 50 percent chance of changing
- Given an input file and its corresponding message digest, it should be computationally infeasible to find another file with the same message digest value



Definition

- Cryptographic checksum function $h: A \rightarrow B$:
 1. For any $x \in A$, $h(x)$ is easy to compute
 - Makes hardware/software implementation easy
 2. For any $y \in B$, it is computationally infeasible to find $x \in A$ such that $h(x) = y$
 - *One-way property*
 3. It is computationally infeasible to find $x, x' \in A$ such that $x \neq x'$ and $h(x) = h(x')$
 4. Alternate form: Given any $x \in A$, it is computationally infeasible to find a different $x' \in A$ such that $h(x) = h(x')$.



Collisions

- If $x \neq x'$ and $h(x) = h(x')$, x and x' are a collision
 - Pigeonhole principle: if there are n containers for $n+1$ objects, then at least one container will have 2 objects in it.
 - Application: suppose $n = 5$ and $k = 3$. Then there are 32 elements of A and 8 elements of B, so
 - each element of B has at least 4 corresponding elements of A



Keys

- Keyed cryptographic checksum:
requires cryptographic key
 - DES in chaining mode: encipher message, use last n bits. Requires a key to encipher, so it is a keyed cryptographic checksum.
- Keyless cryptographic checksum:
requires no cryptographic key
 - MD5 and SHA-1 are best known; others include MD4, HAVAL, and Snefru



Message Digest

- MD2, MD4, MD5 (Ronald Rivest)
 - Produces 128-bit digest;
 - MD2 is probably the most secure, longest to compute (hence rarely used)
 - MD4 is a fast alternative; MD5 is modification of MD4
- SHA, SHA-1 (Secure Hash Algorithm)
 - Related to MD4; used by NIST's Digital Signature
 - Produces 160-bit digest
 - SHA-1 may be better
- SHA-256, SHA-384, SHA-512
 - 256-, 384-, 512 hash functions designed to be use with the Advanced Encryption Standards (AES)
- Example:
 - MD5(There is \$1500 in the blue bo) = f80b3fde8ecbac1b515960b9058de7a1
 - MD5(There is \$1500 in the blue box) = a4a5471a0e019a4a502134d38fb64729

Hash Message Authentication Code (HMAC)

- Make keyed cryptographic checksums from keyless cryptographic checksums
- h be keyless cryptographic checksum function that takes data in blocks of b bytes and outputs blocks of l bytes. k' is cryptographic key of length b bytes (from k)
 - If short, pad with 0s' to make b bytes; if long, hash to length b
- $ipad$ is 00110110 repeated b times
- $opad$ is 01011100 repeated b times
- $HMAC-h(k, m) = h(k' \oplus opad || h(k' \oplus ipad || m))$
 - \oplus exclusive or, $||$ concatenation



Protection Strength

- Unconditionally Secure
 - Unlimited resources + unlimited time
 - Still the plaintext CANNOT be recovered from the ciphertext
- Computationally Secure
 - Cost of breaking a ciphertext exceeds the value of the hidden information
 - The time taken to break the ciphertext exceeds the useful lifetime of the information

Average time required for exhaustive key search

| Key Size (bits) | Number of Alternative Keys | Time required at 10^6 Decryption/ μ s |
|-----------------|--------------------------------|---|
| 32 | $2^{32} = 4.3 \times 10^9$ | 2.15 milliseconds |
| 56 | $2^{56} = 7.2 \times 10^{16}$ | 10 hours |
| 128 | $2^{128} = 3.4 \times 10^{38}$ | 5.4×10^{18} years |
| 168 | $2^{168} = 3.7 \times 10^{50}$ | 5.9×10^{30} years |



Key Points

- Two main types of cryptosystems: classical and public key
- Classical cryptosystems encipher and decipher using the same key
 - Or one key is easily derived from the other
- Public key cryptosystems encipher and decipher using different keys
 - Computationally infeasible to derive one from the other