INTRODUCTION TO COMPUTER AND COMMUNICATIONS SECURITY

Security Objective

Protection of information to ensure:

(i) privacy

(ii) authenticity

Introduction

· in operating systems "*access control*" methods are used

· "*subjects*" (such as users, processes) are given certain access privileges to "*objects*" (such as files, directories, and drives) on an individual, group, and system basis

· individual users authenticate themselves by using a password

Intruder Oscar:

- passive eavesdropping

- active insertion, deletion, and modification of data

Consider user Bob logon to computer system A

- Example scenarios
  - (i) *passive attack*
    - password recorded by Oscar
      - ➔ Oscar now has access to Bob's files

  - (ii) *active attack*
    - Oscar intercepts logon and responds to Bob with "system A down"
      - ➔ Bob's logon prevented

Conclusion:

- password and access control methods can be inadequate

Solution:

- ***CRYPTOGRAPHY*** (Science of Secret Writing)

= Symmetric (or Private) Key Cryptography
+ Public (or Asymmetric) Key Cryptography

Public (or Asymmetric) Key Cryptography → One secret key and one public key.

Symmetric (or Private) Key Cryptography → One cryptographic key known only to txer/rxer.

- cryptographic ***key*** selects parameters of encryption/decryption algorithm or ***cipher***

Classical Cryptography

Transposition Ciphers

- divide message into blocks and transpose characters within block

Substitution Ciphers

- one approach called "shift" cipher: equate each letter to a number from 0 to 25 and add key to each letter in message

Modern Cryptography

Claude Shannon (1949):
- principles of confusion and diffusion

Feistel (1973):
- practical cipher structure⇒ Substitution-Permutation Network

Data Encryption Standard (1975):
- designed by IBM, used extensively in banking

Diffie and Hellman and others (late 1970s):
- invention of public key cryptography and RSA

Explosion of New Research and Applications (late 1980s to present)
- due to rapid growth of distributed computing, wireless networks, and the Internet, importance and interest in cryptographic applications has taken off

Advanced Encryption Standard (2001):
- algorithm to replace DES adopted by NIST after extensive public process

Symmetric Key Cryptosystems

cipher = encryption (E_K) + decryption (D_K) algorithms

· key K ➔ selects parameters of algorithm
➔ key kept private or secret by distributing over secure communication channel
➔ must be kept large enough to prevent exhaustive search on all possible keys

· in order to "break" cipher, cryptanalyst (intruder) will try to find K given some amount of C

· often some amount of C plus corresponding P can be known
➔ cipher not secure if K can be found from knowledge of P and C

· ideally ciphertext C looks random and there is no
(i) statistical
or (ii) mathematical
relationship between C and K or C and P

- two general types of private key ciphers
(1) block ciphers (2) stream ciphers

(Symmetric Key) Block Ciphers

· encrypts/decrypts in blocks of bits

· eg. Data Encryption Standard (DES)
  → most widely applied cipher today
  → blocksize = 64 bits
  → key size = 56 bits

  Advanced Encryption Standard (AES)
  → will become predominant block cipher in next 10 years
  → blocksize = 128 bits
  → key size ≥ 128 bits

· DES breakable using exhaustive seach on special purpose hardware or by distributing work across Internet

· AES should be secure for decades

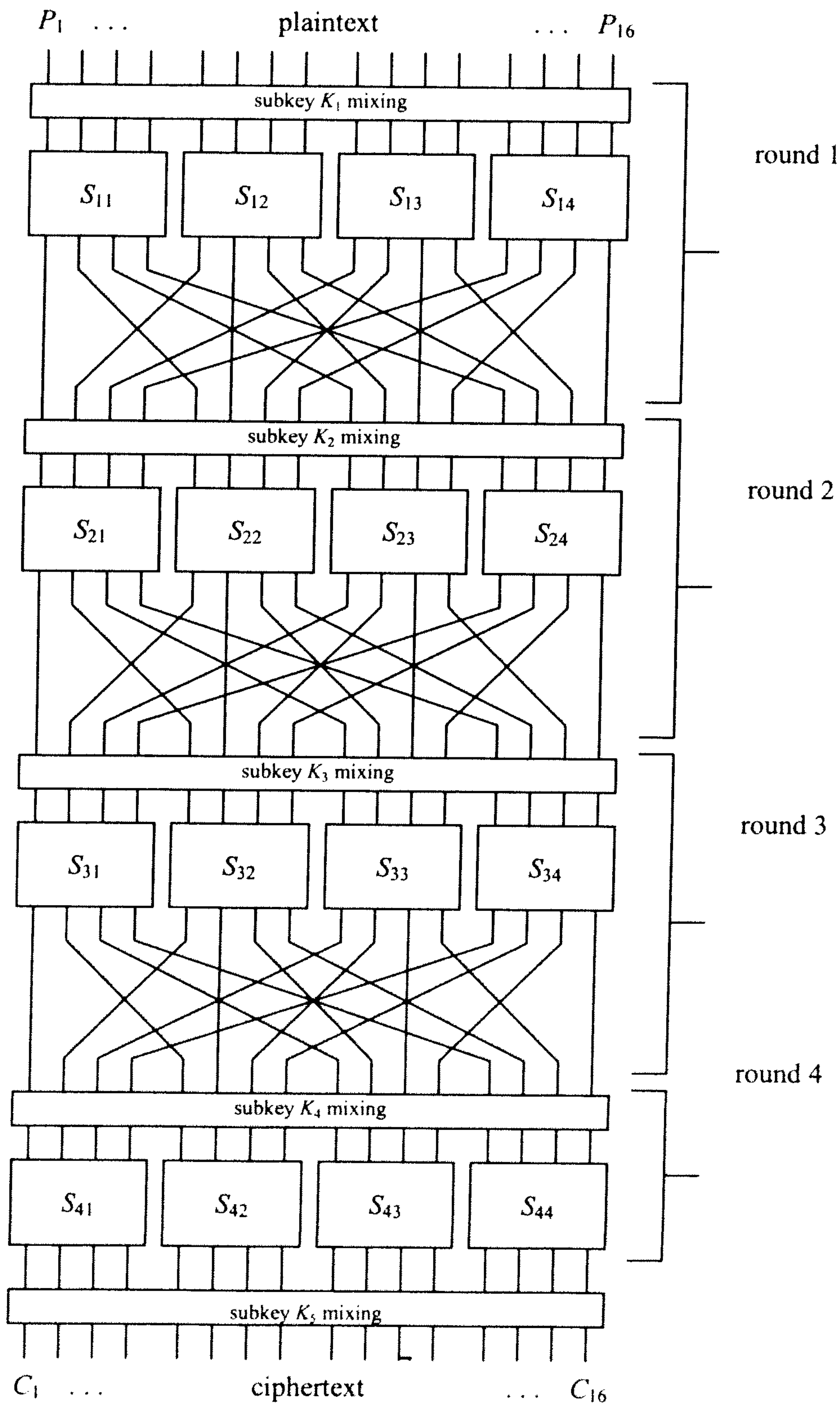
· most symmetric key block ciphers are based on Shannon's fundamental principles:

  → *confusion* - complex mathematical relationship (eg. nonlinear relationship of *P* and *C*)
  → *diffusion* - local effects in plaintext block spread across all ciphertext bits (essentially statistical strength)

· using these principles, "product ciphers" are constructed by executing a sequence of rounds of simple cryptographic operations

· eg. Substitution-Permutation Networks (SPNs)

Substitution-Permutation Networks



S-boxes

· small $n \times n$ nonlinear mapping provides "confusion"

Permutation

· transposition of bits between rounds of S-boxes provides "diffusion"

Keying

· key bits XORed to data bits at S-box inputs
· applied according to key scheduling algorithm

· DES and AES are similar to SPNs

But there is a problem with symmetric key systems

➔ key distribution is a difficult problem since a reliable, secure channel is required

Can we distribute keys over insecure channels like the Internet?

➔ YES! Using Public Key Cryptography

Public Key Ciphers

K_P = public key, K_S = secret key

- K_S only known to receiver
- K_P can be known to everyone including intruder
- K_S cannot be determined from K_P because only receiver knows relationship
- based on "hard" number theory problems
- best known public key cipher is RSA

RSA

(1) receiver chooses two large primes p and q

(2) receiver picks K_S and computes K_P from
$K_P K_S = 1 \bmod (\Phi(N))$
where $N = p \cdot q$ and $\Phi(N) = (p-1)(q-1)$
(Euler Totient Function)

(3) receiver sends N and K_P to transmitter

(4) transmitter can send encrypted information to receiver using exponentiation

Encryption: $C = P^{K_P} \bmod N$ (*)

Decryption: $P = C^{K_S} \bmod N$ (**)

· anyone who acquires K_P can encrypt but only receiver knows K_S and can decrypt

But if (*) is true, how do we know (**) will work?

➔ Let ψ be given by
$\psi = C^{K_S} \bmod N$

➔ Hence ψ

$$\begin{aligned} \psi &= (P^{K_P})^{K_S} \bmod N \\ &= P^{K_P K_S} \bmod N \\ &= P^{1+l\Phi(N)} \bmod N \\ &= P \cdot P^{l\Phi(N)} \bmod N \\ &= P \end{aligned}$$

- since $P^{l\Phi(N)} \bmod N = 1$ (Fermat's Theorem)

Why is RSA secure?

- Difficult to factor large composites and difficult to compute discrete logarithms.

Factoring:

$N = p \cdot q$
"Given N, what are p and q?" ➔ Hard Problem

Discrete Log:

$y = a^x \bmod N$
"Given y, a, and N, what is x?" ➔ Hard Problem

RSA math:

≥ 512 bits to be secure - quite slow!

Authentication

· various requirements

→ user, file, computer, etc.

Consider 3 techniques:

(1) Message Authentication Code

(2) Challenge-Response Protocol

(3) RSA Digital Signature

Message Authentication Code (MAC)

· used to verify authenticity of a message
(i.e., no alterations, correctness of origin)

· uses private key block cipher with both parties in communication knowing key K

· message divided into blocks and chained through block cipher

· can only be generated and verified knowing *K*

· MAC can be attached to end of unencrypted message

Challenge-Response Protocol

Consider

· Alice wants to communicate with Bob

· How can Bob ensure that he is talking to Alice given that they both know the same private key *K*?

· Alice cannot reveal key *K* because someone might be impersonating Bob or eavesdropping

Bob challenges Alice!

RSA Digital Signature

- digital signature must verify that particular user originated a particular electronic document

  ➔ originator cannot deny signature

  ➔ recipient and others cannot forge signature

- hence, signature must be unique to person signing and must change for each document

Consider using RSA public key cipher for Alice to sign contract with Bob:

Let M = contract

· "digest(M)" publicly known digest or hash function to produce small block out of large message

· to ensure that Alice cannot deny, public key K_P should be kept on file by central authority