

Lecture9:NetworkSecurity

GuevaraNoubir

COM3515

Textbook: ComputerNetworks:ASystemsApproach,
L.Peterson,B.Davie,Morgan Kaufmann
Chapter8.

- Why security?
 - Internet, E-commerce, Digi-Cash, cell-phone cloning/hacking, disclosure of private information...
- Security services:
 - Authentication, Confidentiality, Integrity, Access control, Non-repudiation, availability
- Cryptographical algorithms:
 - Symmetric (DES, IDEA, AES, MD5), asymmetric (RSA, Discrete logarithms)
 - One-way functions, trap-door, hashing functions,

Plan

- Introduction
- Symmetric cryptography
- Asymmetric cryptography
- Security services
- Examples

Terminology

- Security services:
 - authentication, confidentiality, integrity, access control, non repudiation, availability, key management
- Security attacks:
 - passive, active
- Cryptograph models:
 - symmetric, asymmetric
- Cryptanalysis:
 - ciphertext only, known plaintext, chosen plaintext, chosen ciphertext, chosen text

Security services

- **Authentication:**
 - assure the recipient of a message the authenticity of the claim
- **Access control:**
 - limit the access to authorized users
- **Confidentiality=Privacy:**
 - protect against unauthorized release of message content
- **Integrity:**
 - guarantee that a message is received as sent
- **Non-repudiation:**
 - protect against sender/receiver denying sending/receiving a message
- **Availability:**
 - guarantee that the system services are always available when needed
- **Security audit:**
 - keep track of transactions for later use (diagnostic, alarms...)
- **Key management:**
 - allow to negotiate, set up and maintain keys between communicating entities

Security Attacks

- Security attacks:
 - Interception (confidentiality)
 - Interruption (availability)
 - Modification (integrity)
 - Fabrication (authenticity)
- Kent's classification
 - Passive attacks:
 - Release of message content
 - Traffic analysis
 - Active attacks:
 - Masquerade
 - Replay
 - Modification of message
 - Denial of service

Kerchoff's Principle

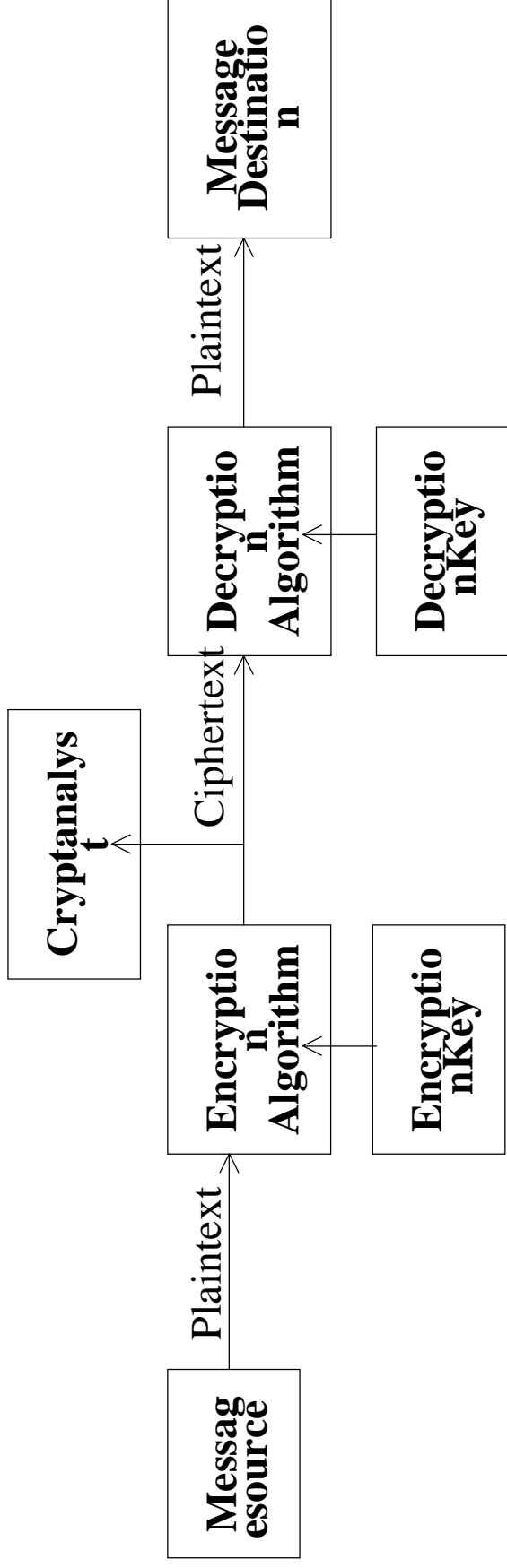
- The cipher should be secure when the intruder knows all the details of the encryption process except for the secret key
- “No security by obscurity”

Attacks on encrypted Messages

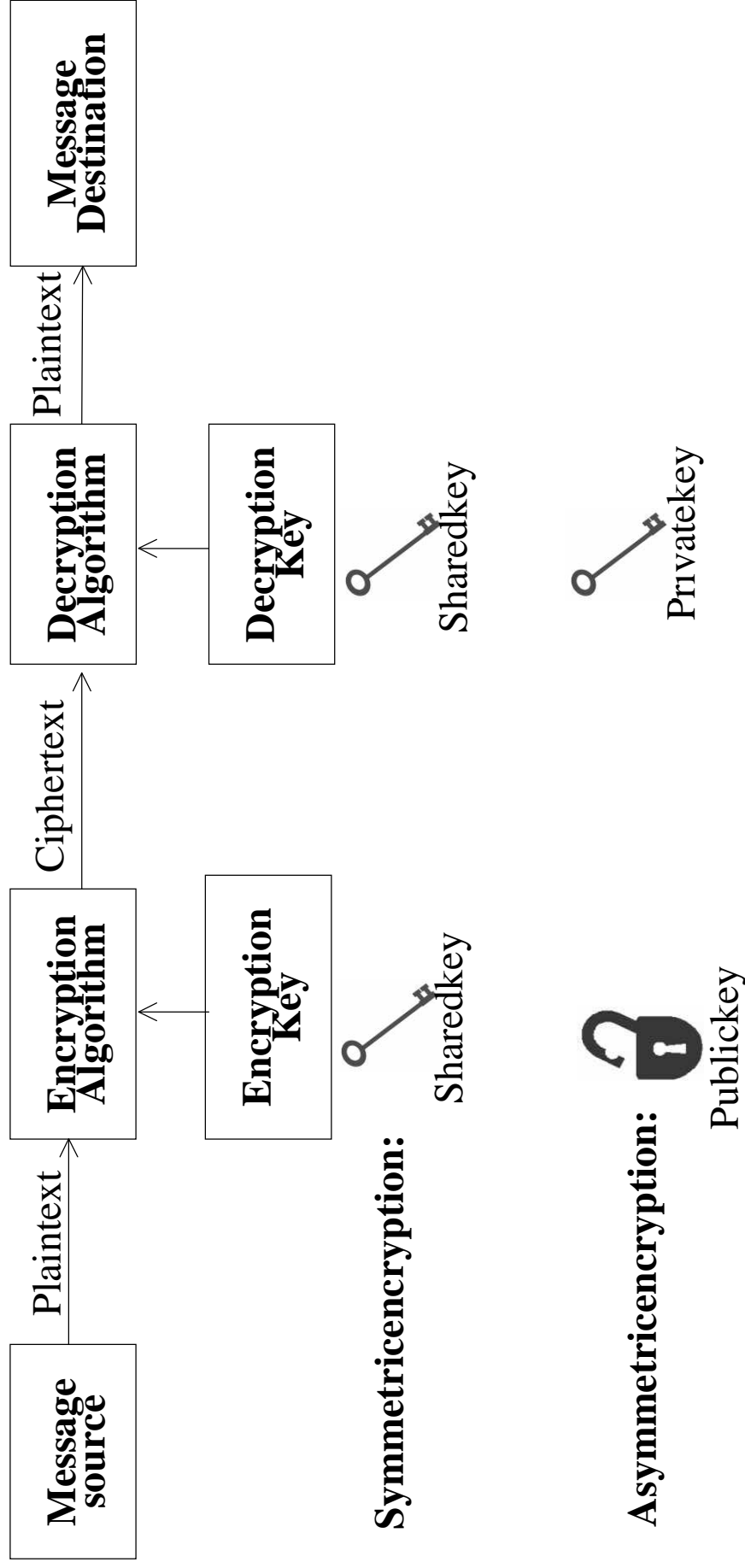
- Ciphertext only:
 - encryption algorithm, ciphertext to be decoded
- Known plaintext:
 - encryption algorithm, ciphertext to be decoded, pair of (plaintext, ciphertext)
- Chosen plaintext:
 - encryption algorithm, ciphertext to be decoded, plaintext (chosen by cryptanalyst) + corresponding ciphertext
- Chosen ciphertext:
 - encryption algorithm, ciphertext to be decoded, ciphertext (chosen by cryptanalyst) + corresponding plaintext
- Chosen text:
 - encryption algorithm, ciphertext to be decoded, plaintext + corresponding ciphertext

Encryption Models

- Symmetric encryption (conventional encryption)
 - Encryption Key = Decryption Key
 - E.g., DES, FEAL, IDEA, BLOWFISH, AES
- Asymmetric encryption
 - Encryption Key \neq Decryption Key
 - E.g., RSA, Diffie-Hellman, DSA



Encryption Models



Building Blocks of Cryptography

- Encryption algorithms
- On-way hashing functions (=message digest, cryptographic checksum, message integrity check, etc.)
 - Input: variable length string
 - Output: fixed length (generally smaller) string
 - Hard to generate a pre-image (input) string that hashes to a given string
- One-way functions
 - $y = f(x)$: easy to compute
 - $x = f^{-1}(y)$: much harder to reverse (it would take a million of years)
 - Example:
 - multiplication of 2 large prime numbers versus factoring
 - discrete exponentiation/discrete logarithms
- Protocols
 - authentication, key management, digital signatures, etc.

Symmetric cryptosystems (conventional cryptosystems)

- Substitution techniques:
 - Caesarcipher
 - Replace each letter with the letter standing x places further
 - Example: ($x=3$)
 - plain: meet me after the toga party
 - cipher: phhw ph diwhu wkh wrjd sduwb
 - Keyspace: 25
 - Brute force attack: try 25 possibilities
 - Monoalphabetic ciphers
 - Arbitrary substitution of alphabet letters
 - Keyspace: $26! > 4 \times 10^{26}$ > key-space (DES)
 - Attack if the nature of the plaintext is known (e.g., English text):
 - compute the relative frequency of letters and compare it to standard distribution for English (e.g., E: 12.75, T: 9.25, R: 8.5, N: 7.75, etc.)
 - compute the relative frequency of 2-letter combinations (e.g., TH)

Symmetric cryptosystems (Continued)

- Multiple-Letter Encryption(Playfair cipher)
 - Plaintext is encrypted two -letters at a time
 - Based on a 5x5 matrix
 - Identification of individual digraphs is more difficult (26x26 possibilities)
 - A few hundred letters of ciphertext allow to recover the structure of plaintext (and break the system)
 - Used during World War I & II
- Polyalphabetic Ciphers(Vigenère cipher)
 - 26 Caesar ciphers, each one denoted by a key letter
 - key: **d e c e p t i v e d e c e p t i v e d e c e p t i v e**
 - plain: **w e a r e d i s c o v e r e d s a v e y o u r s e l f**
 - cipher: **Z I C V T W Q N G R Z G V T W A V Z H C Q Y G L M G J**
 - Enhancement: auto -key (key=initial||plaintext)
- Rotor machines: multi -round monoalphabetic substitution
 - Used during WWII by Germany (ENIGMA) and Japan (Purple)

One-TimePad

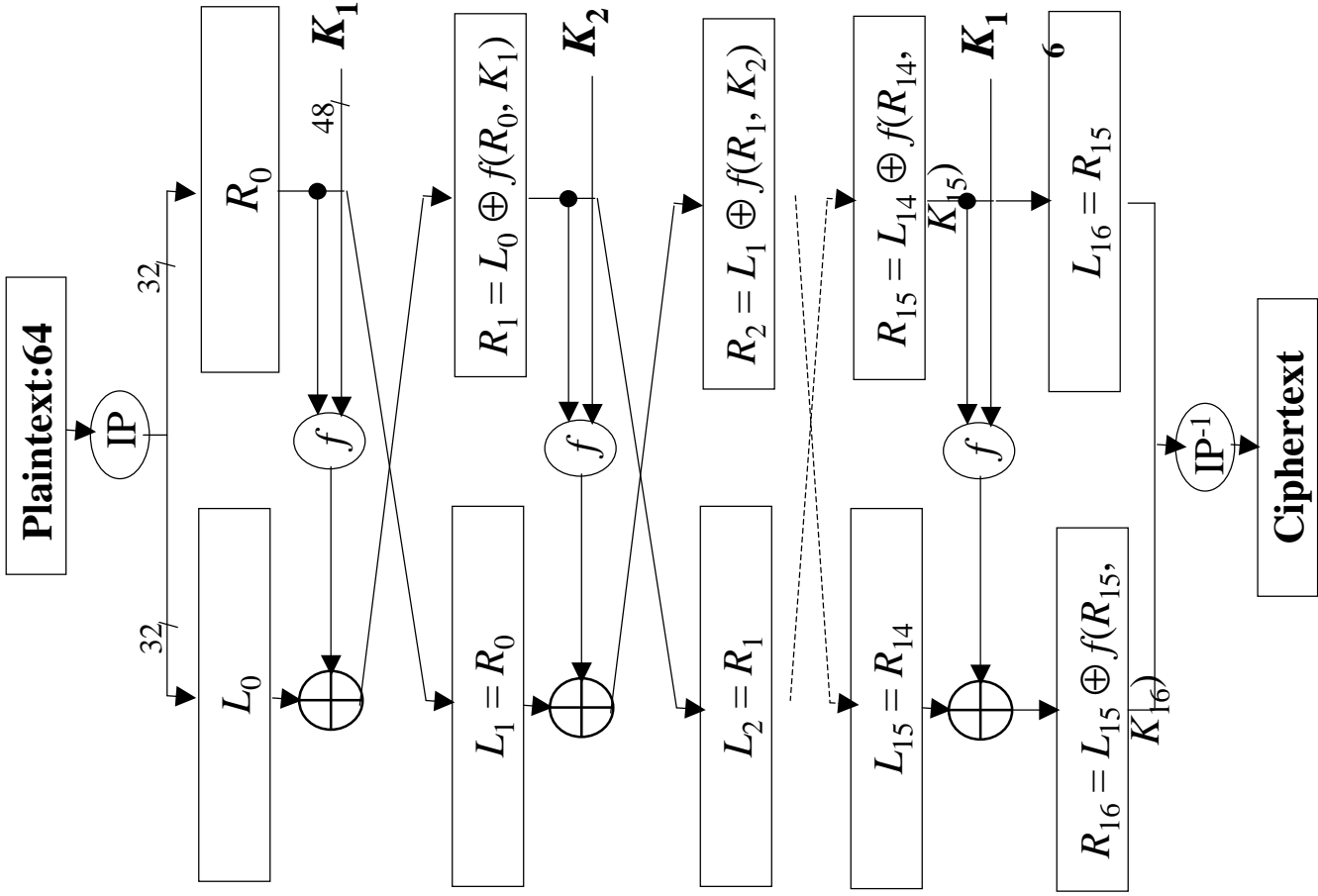
- Introduced by G. Vernam (AT&T, 1918), improved by J. Mauborgne
- Scheme:
 - Encryption: $c_i = p_i \oplus k_i$
 - c_i : i th binary digit of plaintext, p_i : plaintext, k_i : key
 - Decryption: $p_i = c_i \oplus k_i$
 - Key is a random sequence of bits as long as the plaintext
- One-Time Pad is unbreakable
 - No statistical relationship between ciphertext and plaintext
 - Example (Vigenère One-Time Pad):
 - Cipher: **ANKYODKYUREPFJBYOJDSPLREYIUN**
 - Plain-1 (with k1): **MR MUSTARD WITH THE CANDLE**
 - Plain-2 (with k2): **MISS SCARLET WITH THE KNIFE**
- Share the same long key between the sender & receiver

Transposition techniques

- Based on permuting the plaintext letters
- Example: rail fence technique
mematrhtgpry
etefeteoaat
- A more complex transposition scheme
 - Key: **4312567**
 - Plain: **attackp**
ostpone
duntilt
woamxyz
 - Cipher: **TTNAAPTMTSUOAODWCOIXKNLYPETZ**
- Attack: letter/digraph frequency
- Improvement: multiple - stage transposition

Data Encryption Standard (DES)

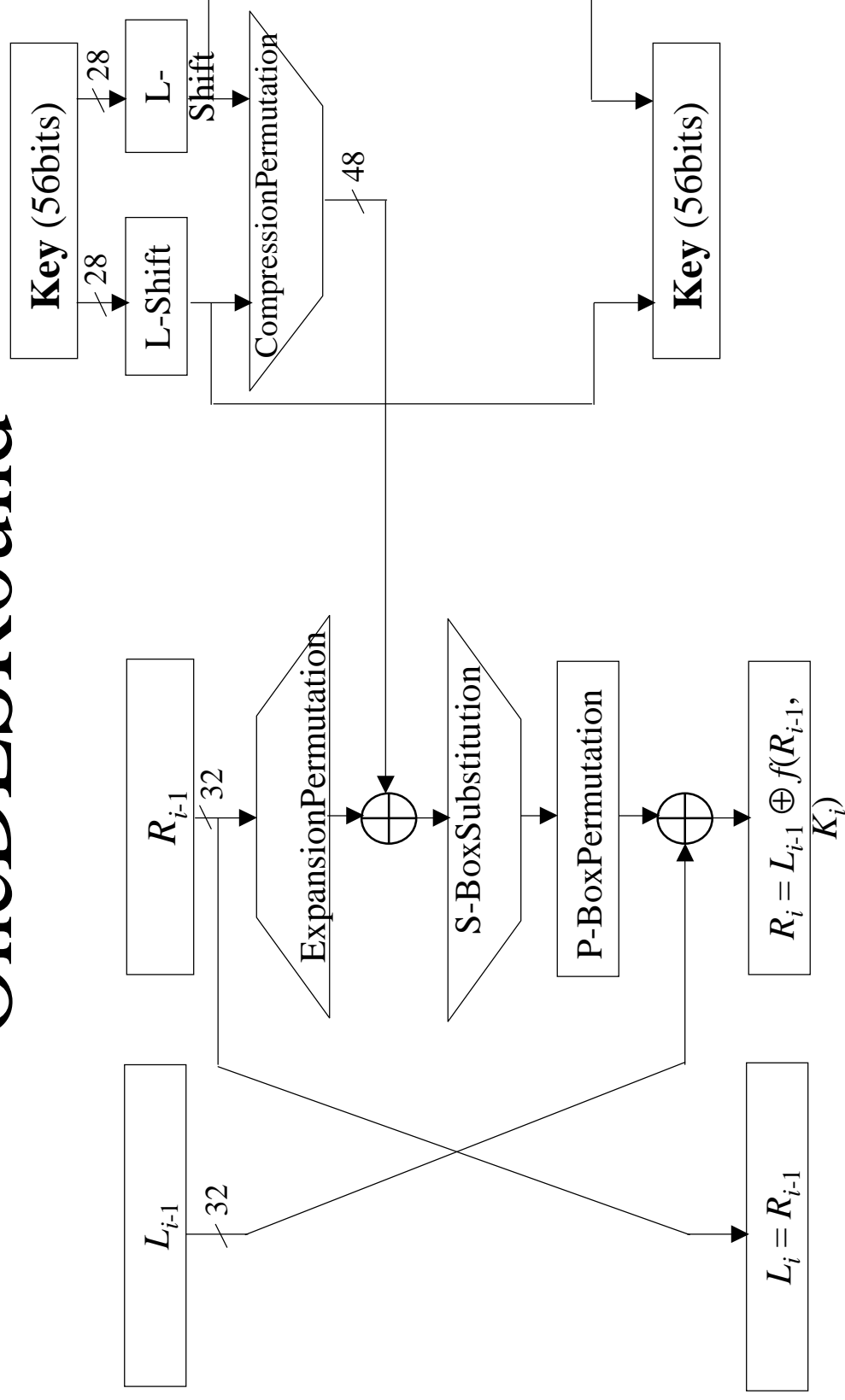
- Developed by IBM for the US government
- Based on Lucifer (64 -bits, 128 -bits key in 1971)
- To respond to the National Bureau of Standards CFP
 - Modified characteristics (with help of the NSA):
 - 64-bit block size, 56 bit key length
 - Concerns about trapdoors!
- Adopted in 1977 as the DES (FIPS PUB 46, ANSI X3.92) and reaffirmed in 1994 for 5 more years
- Most widely used encryption scheme



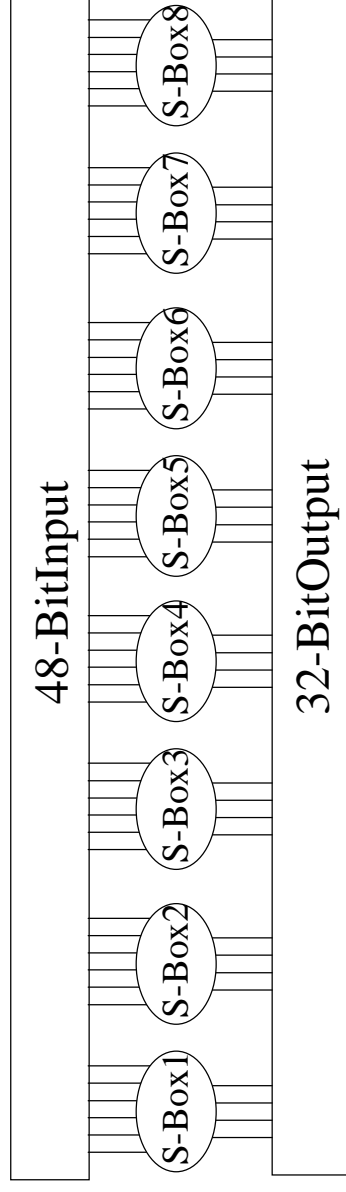
$$L_i = R_{i-1}$$

$$R_i = L_{i-1} \oplus f(R_{i-1}, K_i)$$

OneDES Round



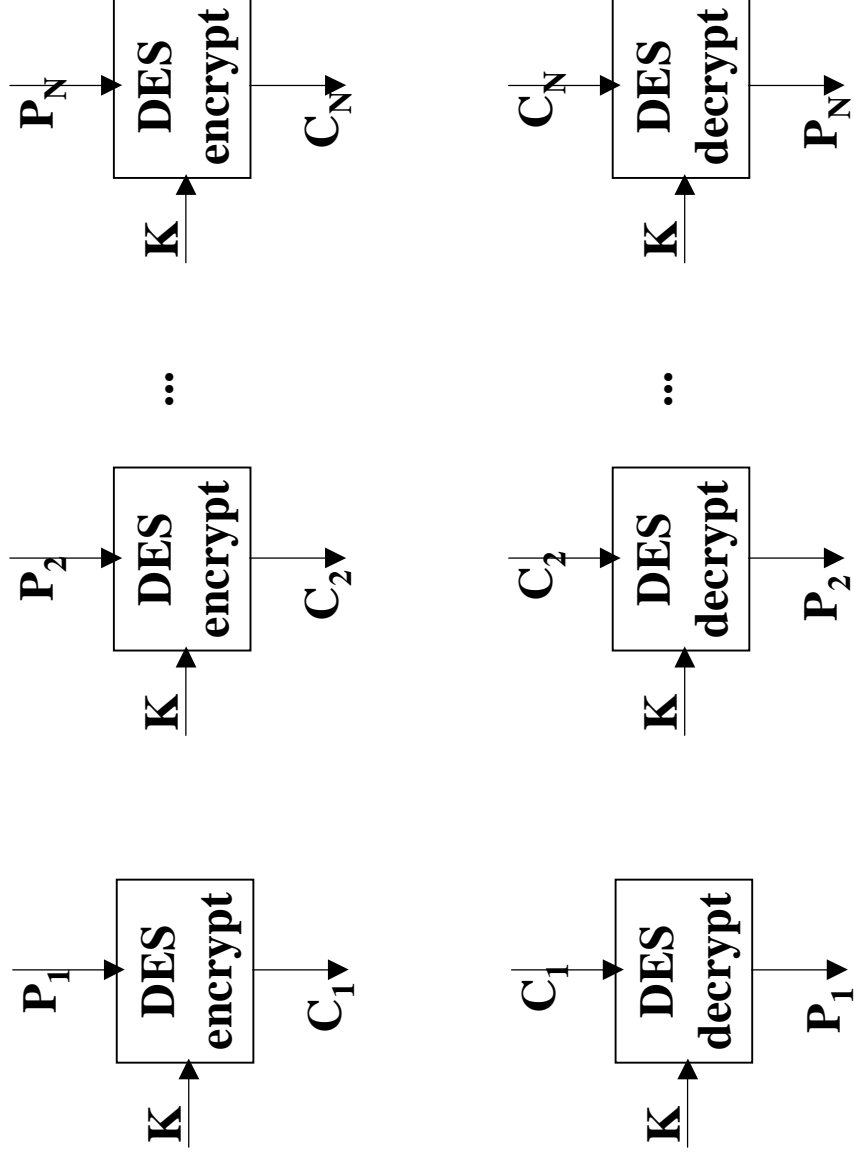
S-Box Substitution



- S-Box heart of DES security
- S-Box: 4x16 entry table
 - Input: 6 bits
 - 2 bits: determine the table (1/4)
 - 4 bits: determine the table entry
 - Output: 4 bits
- S-Boxes are optimized against Differential cryptanalysis

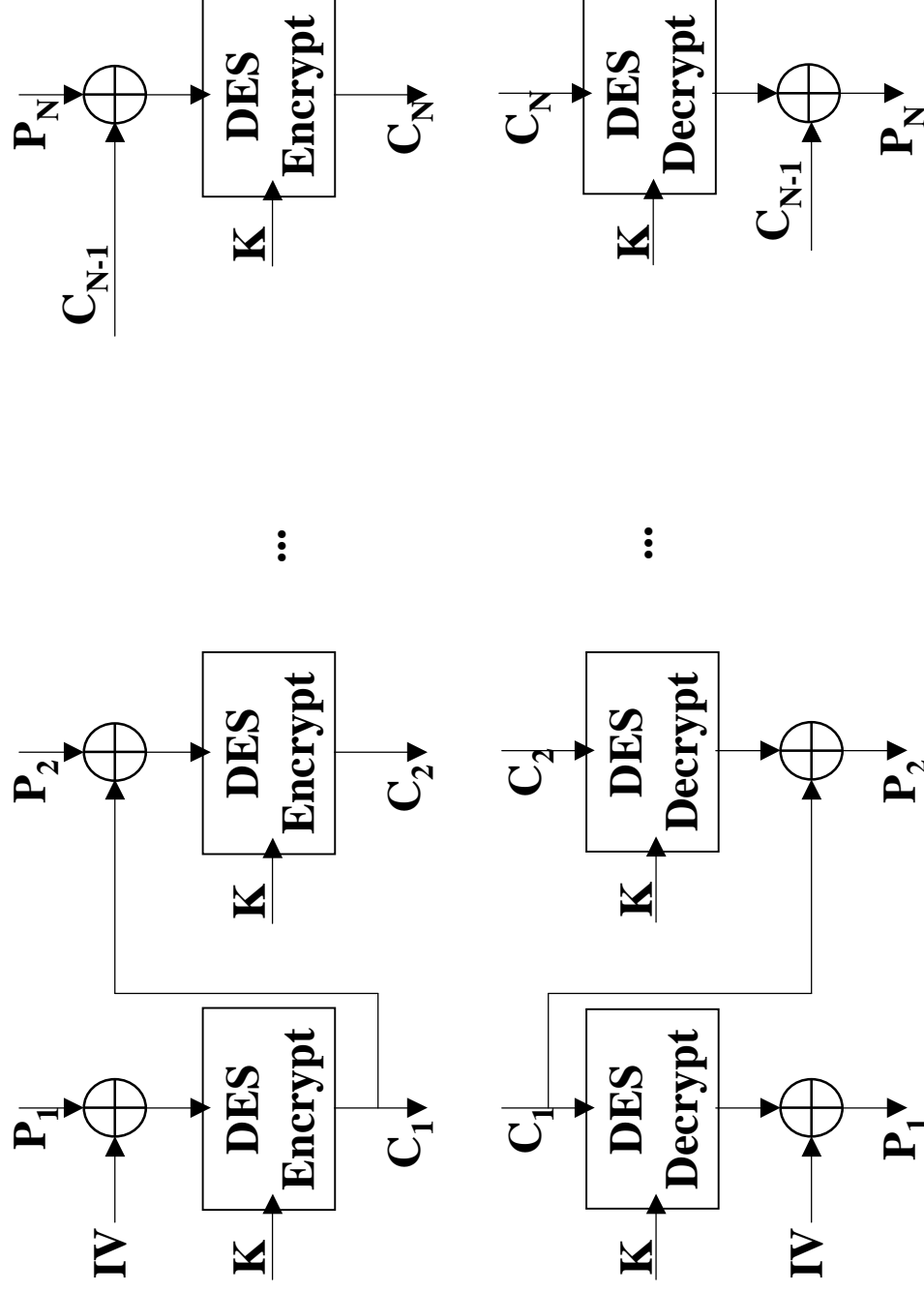
DES Modes: Electronic Codebook

(ECB)



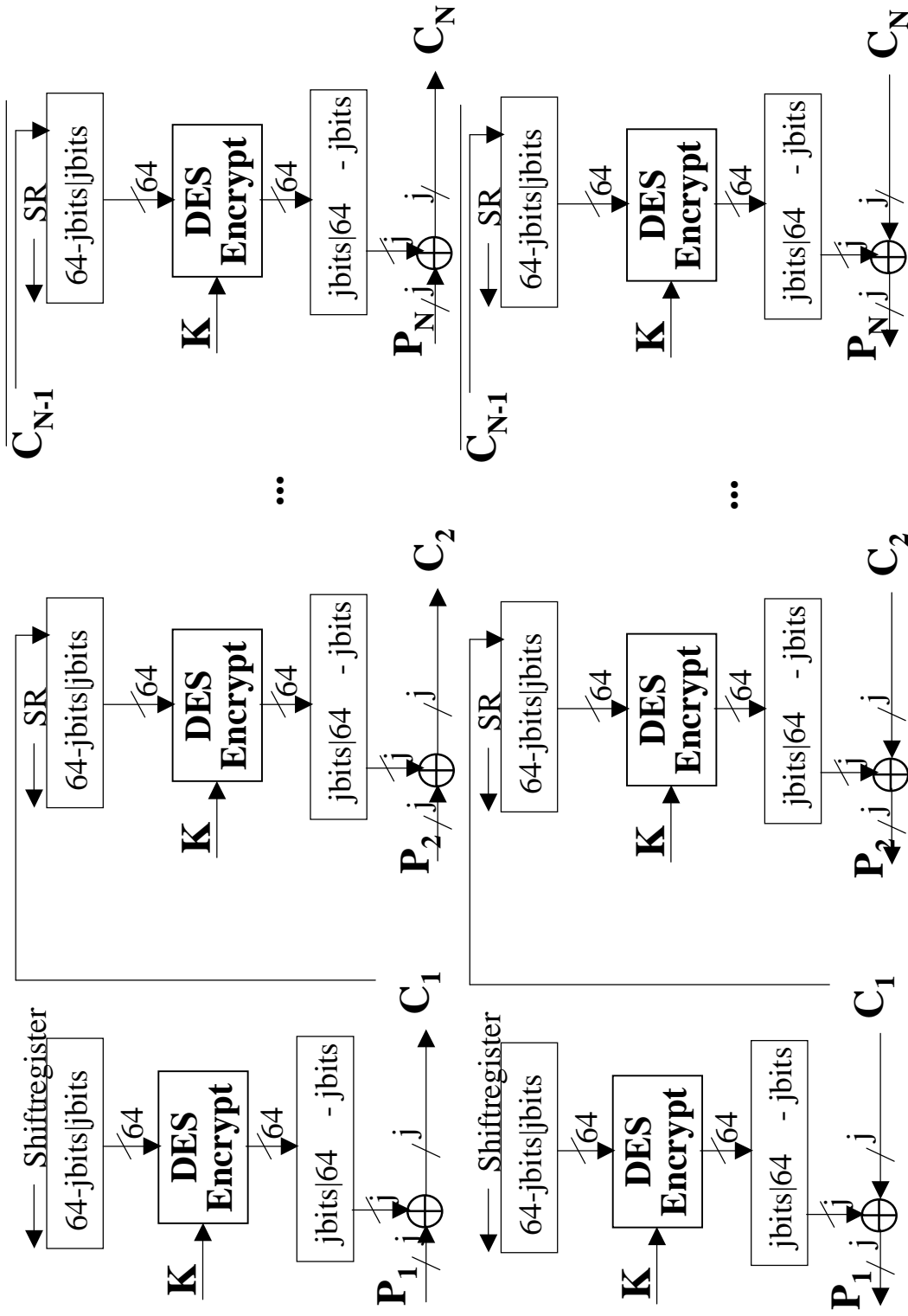
DES Modes: CipherBlock

Chaining(CBC)



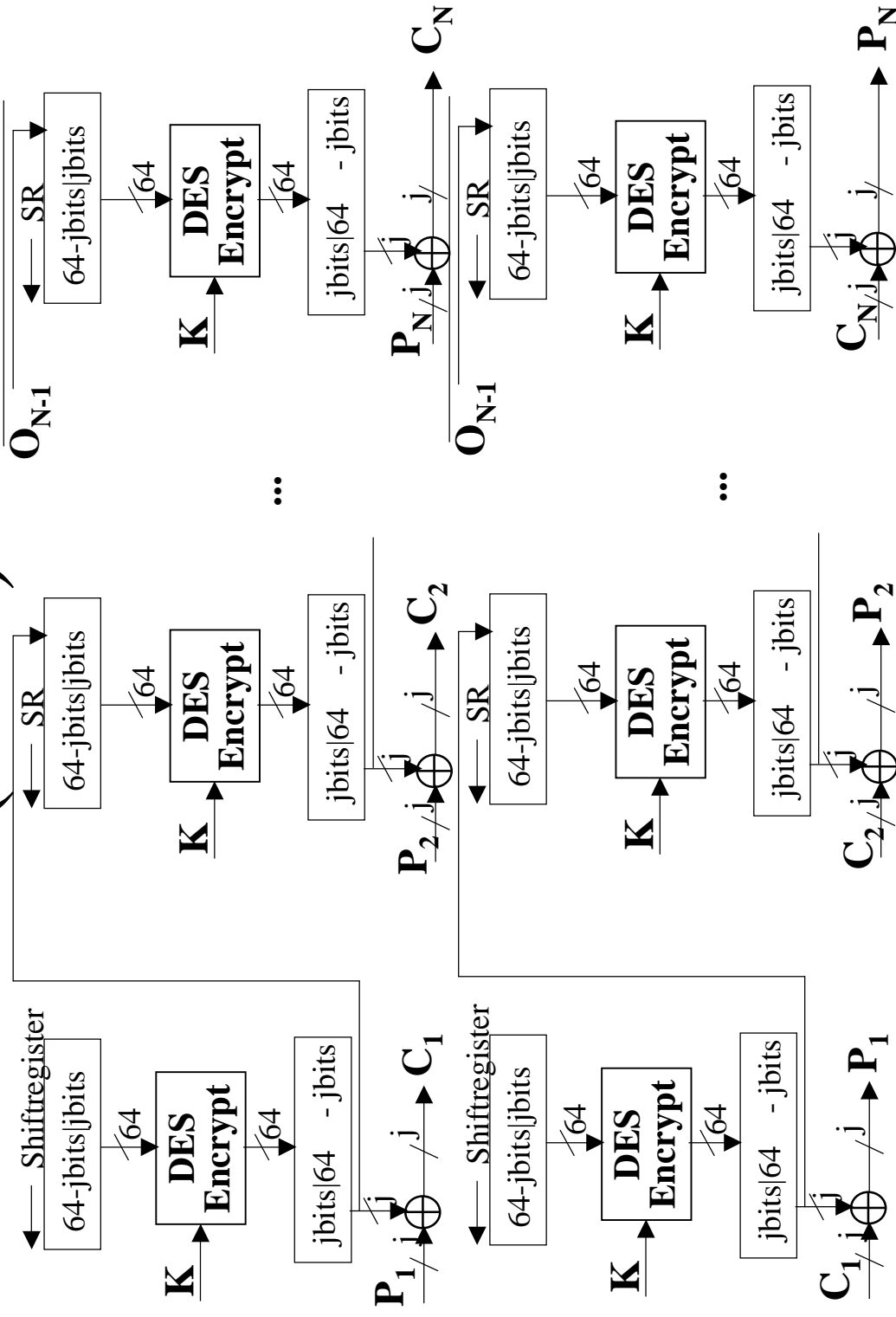
DES Modes: Cipher Feedback

(CFB)



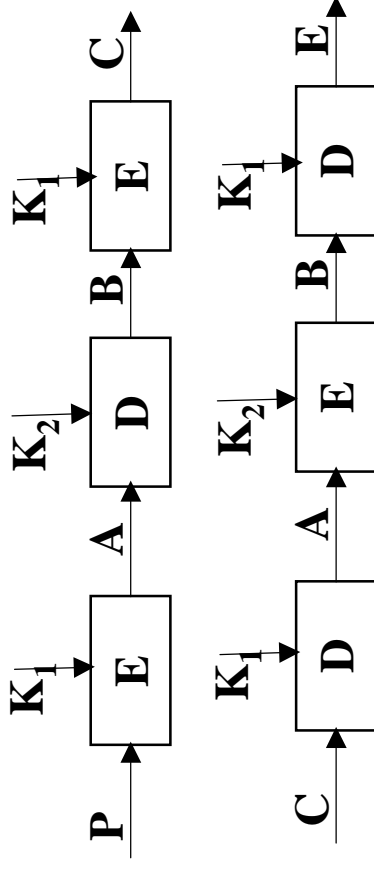
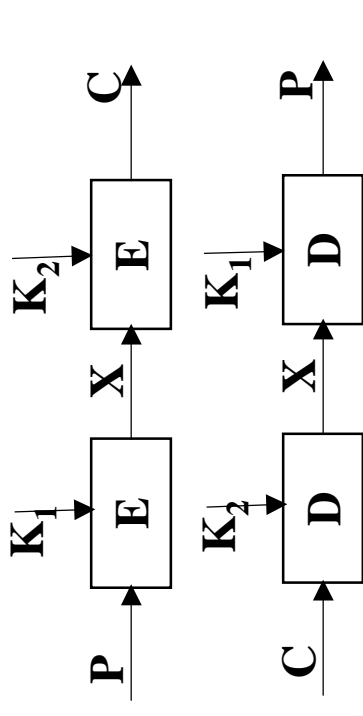
DES Modes: Output Feedback

(OFB)



Double/TripleDES

- DoubleDES
 - Vulnerable to Meet-in-the-Middle Attack [DH77]
- TripleDES
 - Used two keys K_1 and K_2
 - Compatible with simple DES ($K_1 = K_2$)
 - Used in ISO 8732, PEM, ANS X9.17



Linear/Differential Cryptanalysis

- Differential cryptanalysis
 - “Rediscovered” by E. Biham & A. Shamir in 1990
 - Based on chosen -plaintext attack:
 - Analyzed the difference between the ciphertexts of two plaintexts which have a known fixed difference
 - The analysis provided information on the key
 - 8-round DES broken with 2^{14} chosen plaintext and complexity 2^9
 - 16-round DES requires 2^{47} chosen plaintext and complexity 2^{37}
- DES design took into account this kind of attacks
- Linear cryptanalysis
 - Uses linear approximations of the DES cipher (M. Matsui 1993)
- IDEA first proposal (PES) was modified to resist this kind of attacks
- GSM A3 algorithm is sensitive to this kind of attacks
 - SIM card secret key can be recovered \Rightarrow GSM cloning

Breaking DES

- Electronic Frontier Foundation built a “DES Cracking Machine”
 - Attack: brute force
 - Inputs: two ciphertext
 - Architecture:
 - PC
 - array of custom chips that can compute DES
24 search units / chip x 64 chips / board x 24 boards
 - Power:
 - searches 92 billion keys per second
 - takes 4.5 days for half the key space
 - Cost:
 - \$130'000 (all the material: chips, boards, cooling, PC etc.)
 - \$80'000 (development from scratch)

International Data Encryption Algorithm (IDEA)

- Developed by Xu Lai & James Massey
- Characteristics:
 - 64-bit block cipher
 - 128-bit key length
 - Uses three algebraic groups: XOR, $+ \text{mod } 2^{16}$, $\times \text{mod } 2^{16} + 1$
- Speed:
 - software: 2 times faster than DES (2.4 Mbits/second)
 - hardware: VLSI chips reach 177 Mbits/s
- Used in PGP

Advanced Encryption Algorithm (AES)

- NIST has selected Rijndael as the proposed AES algorithm – 2001 – Q2
- Rijndael was developed by cryptographers from Belgium
Joan Daemen of Proton World International and Vincent Rijmen from Katholieke Universiteit Leuven
- Characteristics:
 - Key size: 128, 192, 256 bits
 - Block length: 128, 192, 256 bits
 - Efficient implementation both in HW and SW
 - Low memory requirements
 - Good key agility

MessageDigest5(MD5) [RFC1321]

- Input: message of arbitrary length
- Output: 128-bit hash
- Message is processed in blocks of 512 bits (padding if necessary)
- Designed to resist to the Birthday attack
- Developed by R. Rivest

MD5

- Current digest: d_0, d_1, d_2, d_3 (32bit each)
- Message: m_0, m_1, \dots, m_{15}
- Algorithm = 4 similar passes
- Pass 1 has 16 steps. The first six steps are:
 - $d_0 = (d_0 + F(d_1, d_2, d_3)) + m_0 + T_1$) Rotate -Left 7
 - $d_3 = (d_3 + F(d_0, d_1, d_2)) + m_1 + T_2$) Rotate -Left 12
 - $d_2 = (d_2 + F(d_3, d_2, d_3)) + m_2 + T_3$) Rotate -Left 17
 - $d_1 = (d_1 + F(d_2, d_3, d_0)) + m_3 + T_4$) Rotate -Left 22
 - $d_0 = (d_0 + F(d_1, d_2, d_3)) + m_4 + T_5$) Rotate -Left 7
 - $d_1 = (d_3 + F(d_0, d_1, d_2)) + m_5 + T_6$) Rotate -Left 12
- Pass 2, 3, 4:
 - Change F, T_i , rotation values, sequence of m_i
- Fis based on bitwise operations OR, AND, NOT

Asymmetric cryptosystems

- Invented by Diffie and Hellman [DH76]
 - When DES was proposed for standardization
- Asymmetric systems are much slower than symmetric ones (~1000 times)
- Advantages:
 - does not require a shared key
 - simpler security architecture (no need for a trusted third party)

PublicKey **EncryptedMessage** **PrivateKey**



RSACryptosystem[RSA78]

- $E(M) = M^e \bmod n = C$ (Encryption)
- $D(C) = C^d \bmod n = M$ (Decryption)
- RSA parameters:
 - p, q , two big prime numbers (private, chosen)
 - $n = pq, \phi(n) = (p-1)(q-1)$ (public, calculated)
 - e , with $\gcd(\phi(n), e) = 1, 1 < e < \phi(n)$ (public, chosen)
 - $d = e^{-1} \bmod \phi(n)$ (private, calculated)
- $D(E(M)) = M^{ed} \bmod n = M^{k\phi(n)+1} = M$ (Euler's theorem)

Prime Numbers Generation

- Density of primes (prime number theorem):
 $\pi(x) \sim x / \ln(x)$
- Sieve of Eratostène
 - Try if any number less than \sqrt{n} divides n
- Fermat's Little Theorem:
 - $b^{n-1} = 1 \pmod n$
- Solovay-Strassen primality test
 - If n is not prime at least 50% of b fail to satisfy the following:
 - $b^{(n-1)/2} \neq \pm 1 \pmod n$
- Rabin-Müller primality test
 - If n is not prime then it is not pseudoprime to at least 75% of $b < n$:
 - $n-1 = 2^s t$, $b^t = \pm 1 \pmod n$ or $(b^{t2^k} = -1 \pmod n \text{ and } b^{t2^{k+r}} \neq 1)$
 - probabilistic test, deterministic if the Generalized Riemann Hypothesis is true
- Adleman-Pomerance-Rumely and Cohen - Lenstra test
 - Use an analog of Fermat's Little theorem in extension fields of the rational numbers

Use of RSA

- Encryption (A wants to send a message to B):
 - A uses the public key of B and encrypts M (i.e., $E_B(M)$)
 - Since only B has the private key, only B can decrypt M (i.e., $M = D_B(M)$)
- Digital signature (A wants to send a signed message to B):
 - Based on the fact that $E_A(D_A(M)) = D_A(E_A(M))$
 - A encrypts M using its private key (i.e., $D_A(M)$) and sends it to B
 - B can check that $E_A(D_A(M)) = M$
 - Since only A has the decryption key, only A can generate this message

Security Services

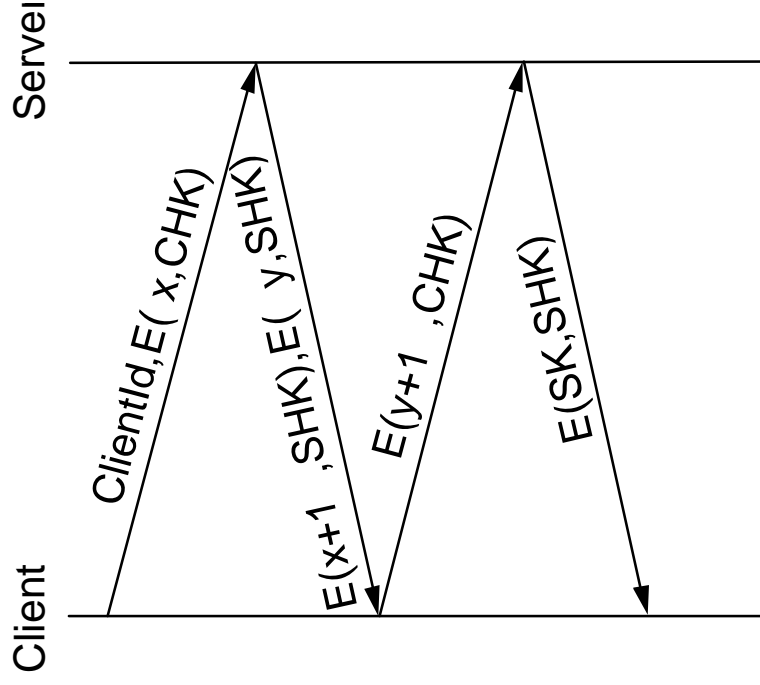
- Confidentiality:
 - Use an encryption algorithm
 - Generally a symmetrical algorithm
- Integrity:
 - Combine a One-way hash function with an encryption function
- Authentication
- Access control:
 - Use access control tables

Message Integrity Protocols

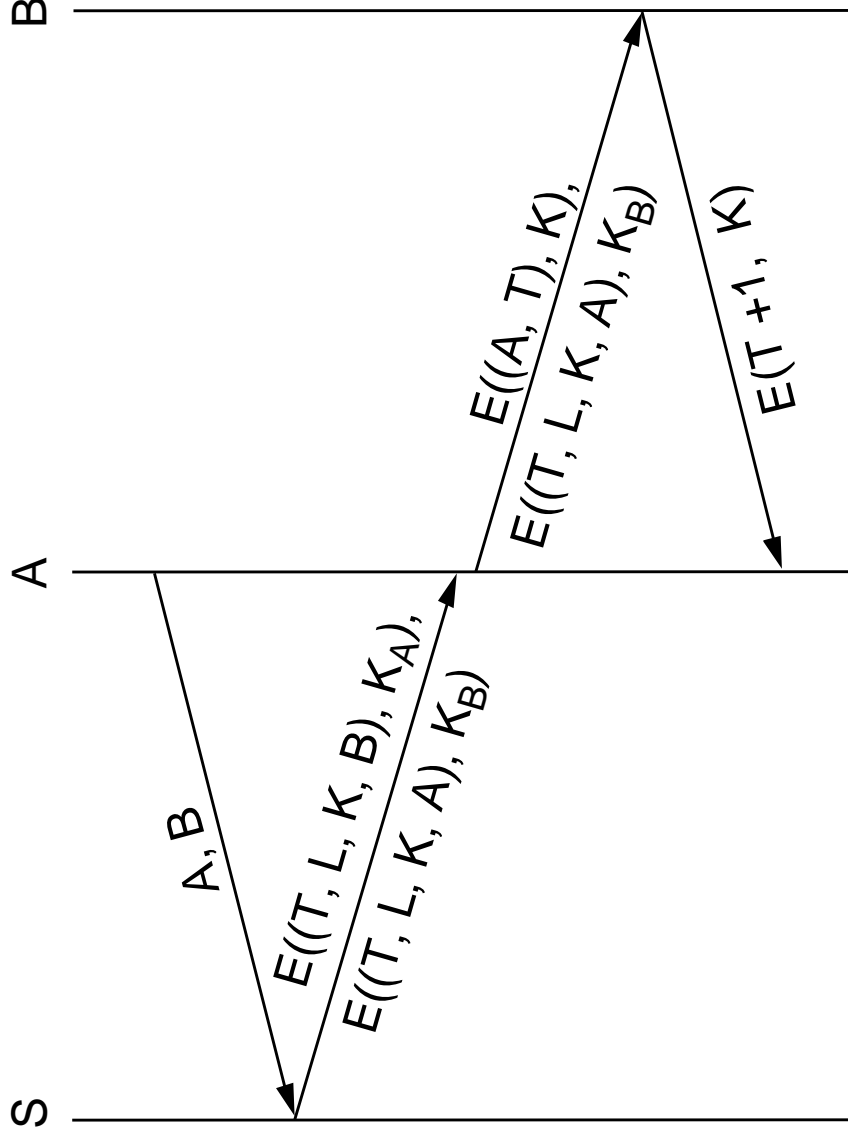
- Digital signature using RSA
 - special case of a message integrity where the code can only have been generated by one participant
 - computes signature with private key and verify with public key
- Keyed MD5
 - sender: $m + \text{MD5}(m + k) + E(k, \text{private})$
 - receiver
 - recovers random key using the sender's public key
 - applies MD5 to the concatenation of this random key and message
- MD5 with RSA signature
 - sender: $m + E(\text{MD5}(m), \text{private})$
 - receiver
 - decrypts signature with sender's public key
 - compares result with MD5 checksum sent with message

Authentication Protocols

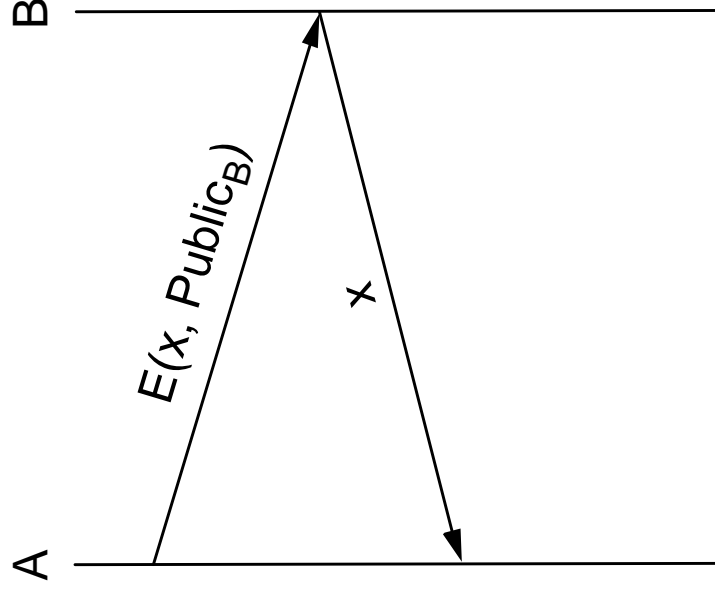
- Three-way handshake



- Trustedthirdparty(Kerberos)

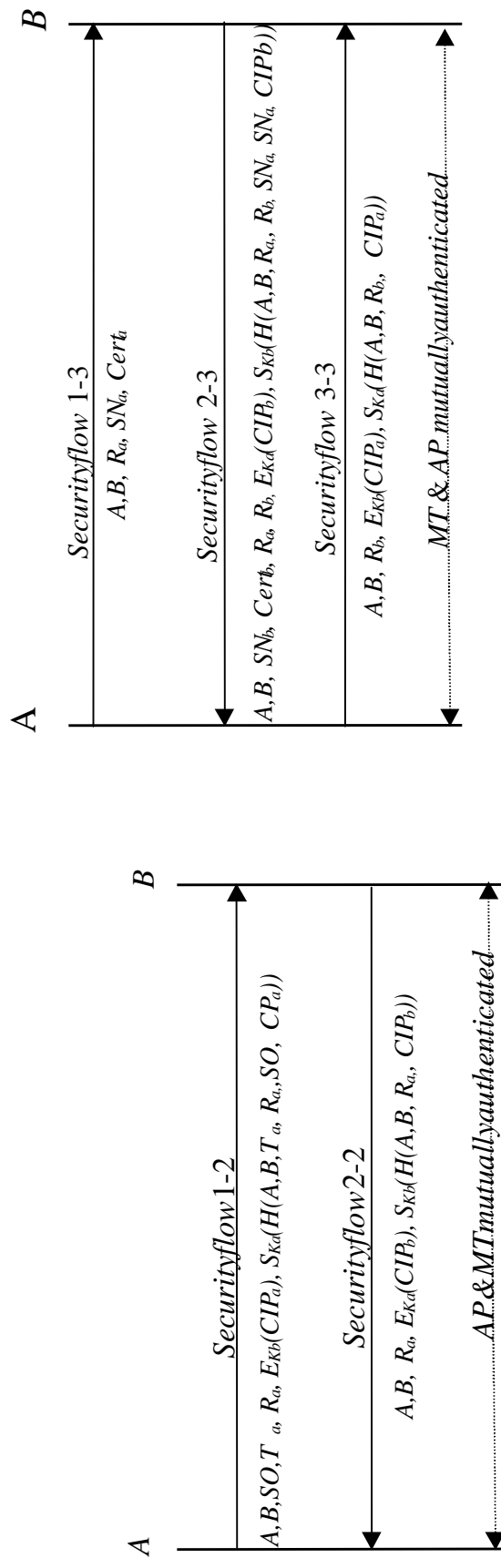


- Publickeyauthentication



Authentication [X.509]

X.509 authentication framework: two/three flow authentication



SO: SecurityOptions (security services to be provided)

T_a: Timestamp

R_i: Random number (nonce) generated by x

CIP_x: Confidentiality & Integrity Parameters x
 masterkey, first integrity sessionkey

E_{K_i}: Encryption function using key (public when asymmetric)

S_{K_i}: Signature function (private key when asymmetric encryption)

H: one-way Hash function (e.g., MD5)

SO: SecurityOptions (security services to be provided)

T_a: Timestamp

R_i: Random number (nonce) generated by x

CIP_x: Confidentiality & Integrity Parameters x
 masterkey, first integrity sessionkey

SN_x: Security Services Negotiation parameters

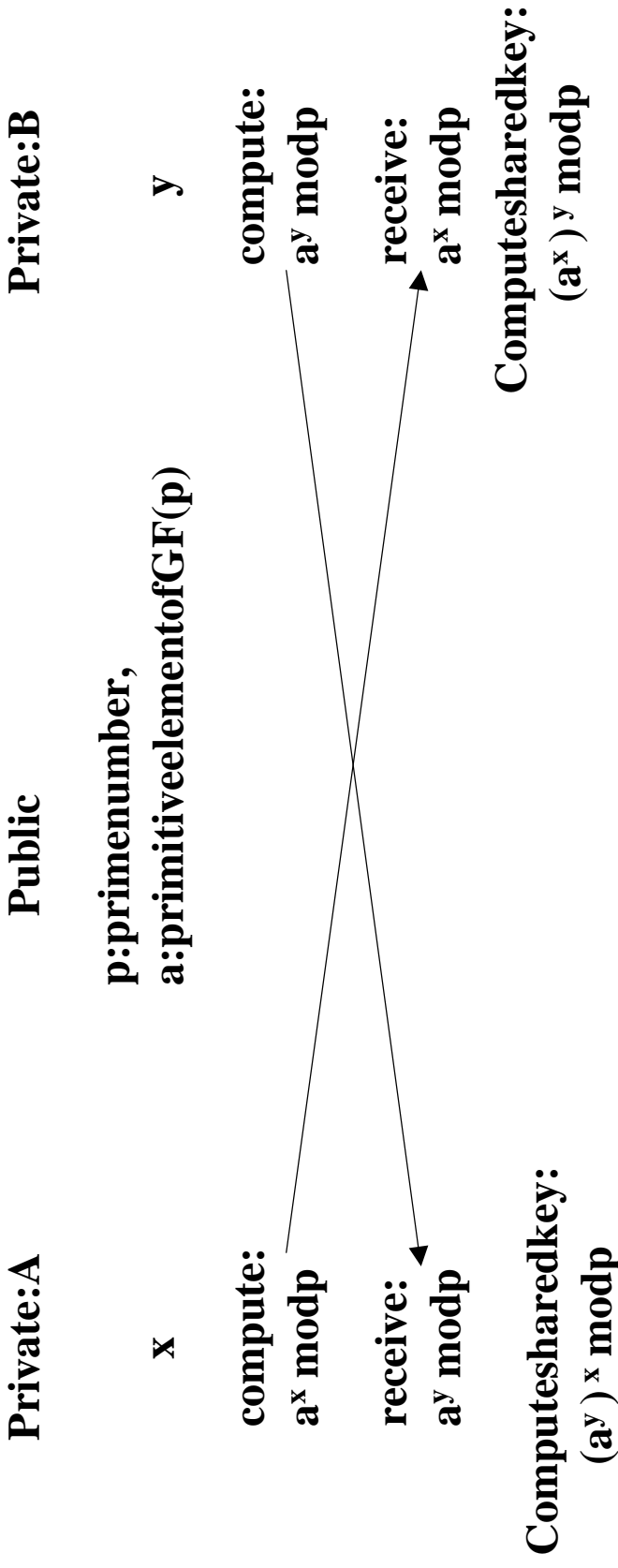
E_{K_i}: Encryption function using key (public when asymmetric)

S_{K_i}: Signature function (private key when asymmetric encryption)

H: one-way Hash function (e.g., MD5)

Cert_x: x Certificate

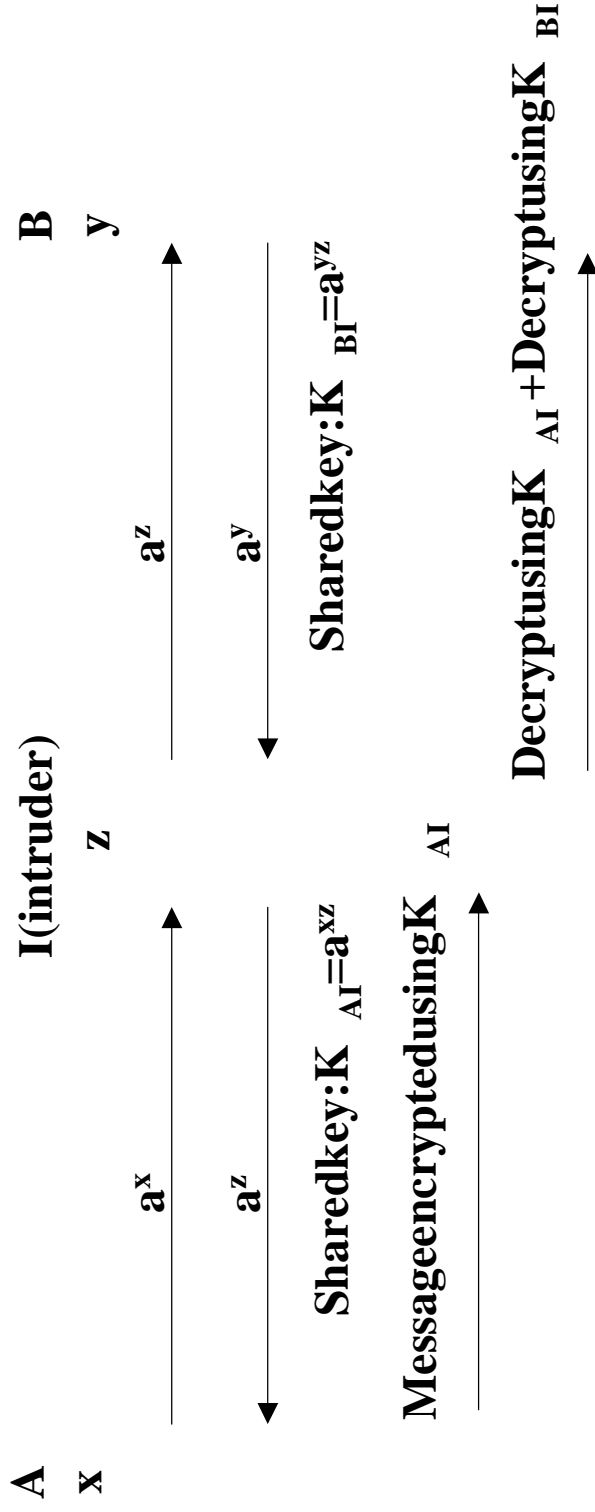
Diffie-Hellman Key Exchange



- Based on the difficulty of computing discrete logarithms
- Works also in extension Galois fields: GF(p^q)

Attack on Diffie-Hellman Scheme: Public Key Integrity

Man-in-the-Middle Attack



- Need for a means to verify the public information: certification
- Another solution: the Interlock Protocol (Rivest & Shamir 1984)

Key Distribution

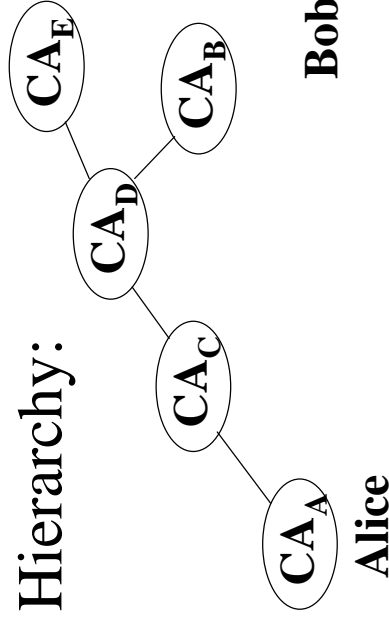
- Certificate
 - special type of digitally signed document:
 - “*I certify that the public key in this document belongs to the entity named in this document, signed X.*”
 - the name of the entity being certified
 - the public key of the entity
 - the name of the certifying authority
 - a digital signature
- Certified Authority (CA)
 - administrative entity that issues certificates
 - useful only to someone that already holds the CA’s public key.

Key Distribution(cont)

- Chain of Trust
 - if X certifies that a certain public key belongs to Y , and Y certifies that another public key belongs to Z , then there exists a chain of certificates from X to Z
 - someone that wants to verify Z 's public key has to know X 's public key and follow the chain
- Certificate Revocation List

Certification

- Problem:
 - A wants to send a message to B
 - How can A be sure that she does not have a fake public key of A (man-in-the-middle attack)?
- Use a certification hierarchy (e.g., X.509)
- Certificate:
 - is a message signed by an authority (which public key is known)
 - contains the public key of a user, identity, validity, algorithm ...
- Hierarchy: **CA: Certification Authority**



Pretty Good Privacy (PGP)

- Provides confidentiality and authentication for electronic mail
- Combines several algorithms:
 - Confidentiality:
 - Select key K_s and encrypt it using RSA public key of the receiver
 - Encrypt the email content with IDEA using K_s
 - Digital signature:
 - Use MD5 for hashing the email content
 - Encrypt the hash code using RSA private key of the sender

Securing Internet Connection

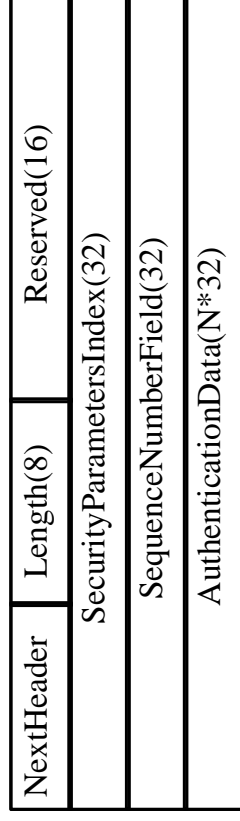
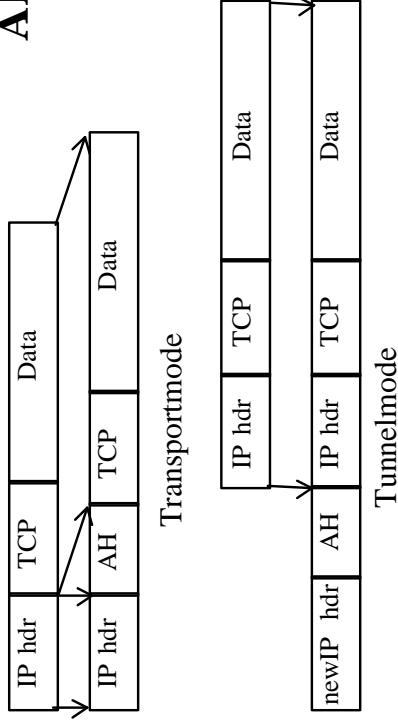
- SecureSocketLayer(SSL)
- IPsec protocols[RFC]
- IEEE802.10

IPsec Protocol Suite (IETF Draft)

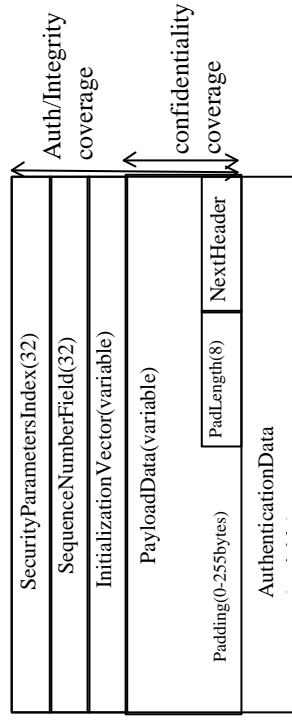
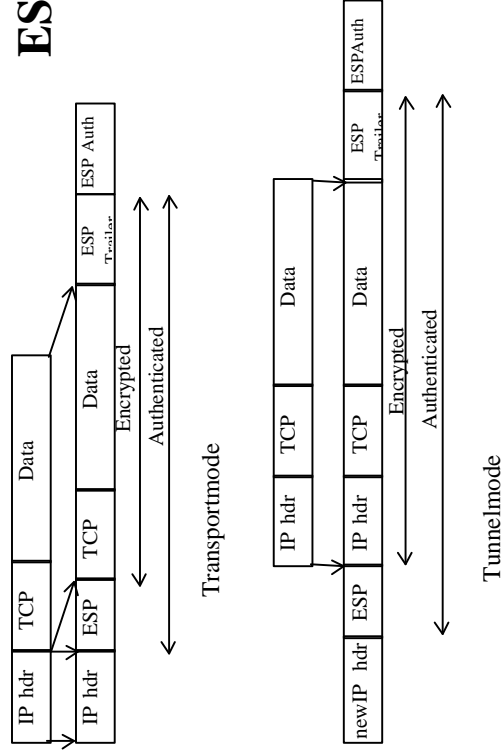
- Provides interoperable cryptographically based security services :
 - **Services: confidentiality, authentication, integrity, and non-repudiation, key management**
 - **Protocols: Authentication Header (AH), Encapsulated Security Payload (ESP) and Internet Key Exchange (IKE), ISAKMP, Oakley**
 - **Environments: IPv4 and IPv6**
 - **Modes: transport (between two hosts) or tunnel (between hosts/security gateways).**

AH and ESP Formatting

AH



ESP



Key Management

- Internet Security Association Establishment and Key Management Protocol (ISAKMP) [1998]:
 - defines procedures and packet format to establish, negotiate, modify and delete Security Association (SA)
 - not bound to any cryptographic, key exchange, key generation algorithm
- Oakley [1998]:
 - describes a series of key exchange protocols
- Internet Key Exchange (IKE) [1998]:
 - combines ISAKMP and Oakley modes

Bibliography

- B. Schneier, “Applied Cryptography”, 1996, John Wiley & Sons, second edition.
- G. Simmons, “Contemporary Cryptology: The Science of Information Integrity”, 1992, IEEE Press.
- W. Stallings, “Network and Internet Security”, 1995, Prentice-Hall.
- Electronic Frontier Foundation, “Cracking DES”, 1998, O’Reilly & Associates.
- J. B. Lacy, D. P. Mitchell, and W. M. Schell. “CryptoLib: Cryptography in Software”, 1993.