


Fundamentals of Cryptography: Algorithms, and Security Services

Professor Guevara Noubir
Northeastern University
noubir@ccs.neu.edu

Cryptography: Theory and Practice, Douglas Stinson, Chapman & Hall/CRC

Network Security: Private Communication in a Public World
Charles Kaufman, Mike Speciner, Radia Perlman, Prentice-Hall


Cryptography and Network Security, William Stallings, Prentice Hall



Outline

- n Introduction to security/cryptography
- n Secret Key Cryptography
 - n DES, IDEA, AES
- n Modes of Operation
 - n ECB, CBC, OFB, CFB, CTR
 - n Message Authentication Code (MAC)
- n Hashes and Message Digest
- n Public Key Algorithms

Network Security Cryptography Overview 2



Why/How?

- n Why security?
 - n Internet, E-commerce, Digi-Cash, disclosure of private information
 - ...
- n Security services:
 - n Authentication, Confidentiality, Integrity, Access control, Non-repudiation, availability
- n Cryptographic algorithms:
 - n Symmetric encryption (DES, IDEA, AES)
 - n Hashing functions
 - n Symmetric MAC (HMAC)
 - n Asymmetric (RSA, El-Gamal)

Network Security Cryptography Overview 3

Terminology

- n Security services:
 - o Authentication, confidentiality, integrity, access control, non-repudiation, availability, key management
- n Security attacks:
 - o Passive, active
- n Cryptography models:
 - o Symmetric (secret key), asymmetric (public key)
- n Cryptanalysis:
 - o Ciphertext only, known plaintext, chosen plaintext, chosen ciphertext, chosen text

Security services

- n Authentication:
 - o assures the recipient of a message the authenticity of the claimed source
- n Access control:
 - o limits the access to authorized users
- n Confidentiality:
 - o protects against unauthorized release of message content
- n Integrity:
 - o guarantees that a message is received as sent
- n Non-repudiation:
 - o protects against sender/receiver denying sending/receiving a message
- n Availability:
 - o guarantees that the system services are always available when needed
- n Security audit:
 - o keeps track of transactions for later use (diagnostic, alarms...)
- n Key management:
 - o allows to negotiate, setup and maintain keys between communicating entities

Security Attacks

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

Kerchoff's Principle

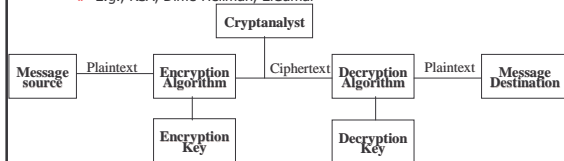
- n The cipher should be secure when the intruder knows all the details of the encryption process except for the secret key
- n "No security by obscurity"
 - n Examples of system that did not follow this rule and failed?


Attacks on Encrypted Messages

- n Ciphertext only:
 - n encryption algorithm, ciphertext to be decoded
- n Known plaintext:
 - n encryption algorithm, ciphertext to be decoded, pairs of (plaintext, ciphertext)
- n Chosen plaintext:
 - n encryption algorithm, ciphertext to be decoded, plaintext (chosen by cryptanalyst) + corresponding ciphertext
- n Chosen ciphertext:
 - n encryption algorithm, ciphertext to be decoded, ciphertext (chosen by cryptanalyst) + corresponding plaintext
- n Chosen text:
 - n encryption algorithm, ciphertext to be decoded, plaintext + corresponding ciphertext (both can be chosen by attacker)

Encryption Models


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





Secret Key Cryptography
 =
 Symmetric Cryptography
 =
 Conventional Cryptography

Network Security
Cryptography Overview
13




Symmetric cryptosystems (conventional cryptosystems)

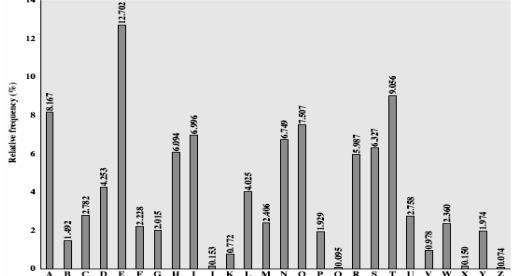
Substitution techniques:

- Caesar cipher
 - 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
 - Key space: 25
 - Brut force attack: try 25 possibilities
- Monoalphabetic ciphers
 - Arbitrary substitution of alphabet letters
 - Key space: $26! > 4 \times 10^{26} >$ 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:12.7, T:9, etc.)
 - compute the relative frequency of 2-letter combinations (e.g., TH)

Network Security
Cryptography Overview
14



English Letters Frequencies



Letter	Relative Frequency (%)
A	8.167
B	1.492
C	2.782
D	4.253
E	12.702
F	2.228
G	2.015
H	6.944
I	6.996
J	0.151
K	0.772
L	4.025
M	2.406
N	6.449
O	7.807
P	1.929
Q	0.095
R	6.907
S	6.427
T	9.054
U	2.758
V	0.978
W	2.360
X	0.150
Y	1.974
Z	0.074

Network Security
Cryptography Overview
15

Symmetric cryptosystems (Continued)

- 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: `deceptive`
 - plain: `wearediscoveredsaveyourself`
 - cipher: `zicvtwqngrzgvttwvzhcqvglmjz`
 - Enhancement: auto-key (key = initial||plaintext)
 - Rotor machines: multi-round monoalphabetic substitution
 - Used during WWII by Germany (ENIGMA) and Japan (Purple)

One-Time Pad

- Introduced by G. Vernam (AT&T, 1918), improved by J. Mauborgne
- Scheme:
 - Encryption: $c_i = p_i \oplus k_i$
 - c_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: `ANKYODKYUREPFJBYQJDSPLREYIUN`
 - 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/Permutation Techniques

- Based on permuting the plaintext letters
- Example: rail fence technique


```
mematrhrtgpry
etefeteoaat
```
- A more complex transposition scheme
 - Key: `4312567`
 - Plain: `attackp`
 - `ostpone`
 - `duntilt`
 - `woamxyz`
 - Cipher: `TTNAAPTMTSUOAODWCOIXKNLYPETZ`
- Attack: letter/diagraph frequency
- Improvement: multiple-stage transposition

Today's Block Encryption Algorithms

- n Key size:
 - o Too short => easy to guess
 - n Block size:
 - o Too short easy to build a table by the attacker: (plaintext, ciphertext)
 - o Minimal size: 64 bits
 - n Properties:
 - o One-to-one mapping
 - o Mapping should look random to someone who doesn't have the key
 - o Efficient to compute/reverse
 - n How:
 - o Substitution (small chunks) & permutation (long chunks)
 - o Multiple rounds
- ⇒ SPN (Substitution and Permutation Networks) and variants

Network Security

Cryptography Overview

19

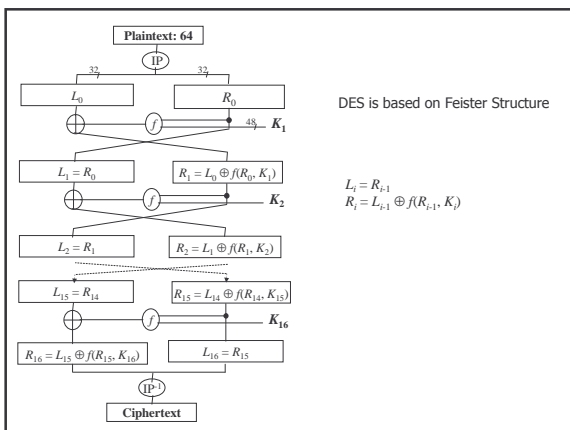
Data Encryption Standard (DES)

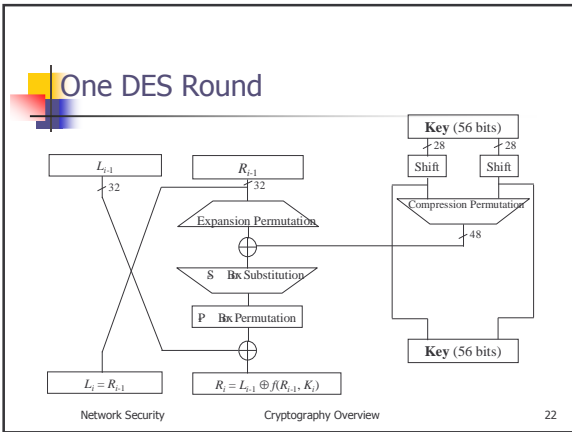
- n Developed by IBM for the US government
- n Based on Lucifer (64-bits, 128-bits key in 1971)
- n To respond to the National Bureau of Standards CFP
 - o Modified characteristics (with help of the NSA):
 - o 64-bits block size, 56 bits key length
 - o Concerns about trapdoors, key size, sbox structure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

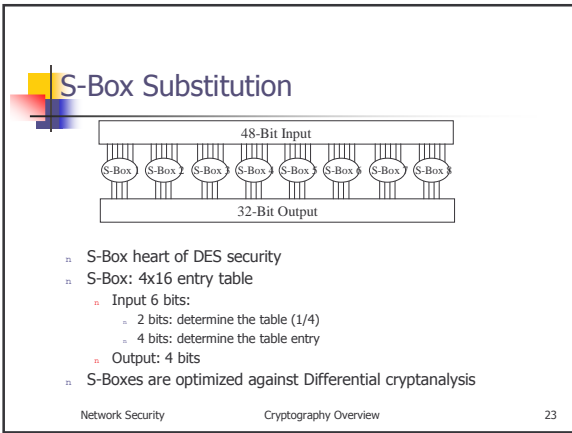
Network Security

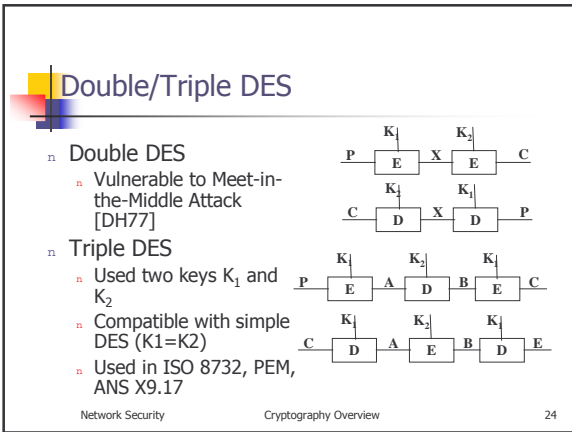
Cryptography Overview

20









Linear/Differential Cryptanalysis

- Differential cryptanalysis
 - "Rediscovered" by E. Biham & A. Shamir in 1990
 - Based on a chosen-plaintext attack:
 - Analyse the difference between the ciphertexts of two plaintexts which have a known fixed difference
 - The analysis provides information on the key
 - 8-round DES broken with 2^{14} chosen plaintext
 - 16-round DES requires 2^{47} chosen plaintext
- 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 to this kind of attacks
- GSM A3 algorithm is sensitive to this kind of attacks
 - SIM card secret key can be recovered => GSM cloning

Network Security

Cryptography Overview

25

Breaking DES

- Electronic Frontier Foundation built a "DES Cracking Machine" [1998]
 - Attack: brute force
 - Inputs: two ciphertext
 - Architecture:
 - PC
 - array of custom chips that can compute DES
 - 24 search units/chip x 64chips/board x 27 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)

Network Security

Cryptography Overview

26

International Data Encryption Algorithm (IDEA)

- Developed by Xu Lai & James Massey (ETH Zurich, Switzerland)
- Characteristics:
 - 64-bits block cipher
 - 128-bits key length
 - Uses three algebraic groups: XOR, $+$ mod 2^{16} , x mod $2^{16}+1$
 - 17 rounds (or 8 rounds according to the description)
- Speed: software: 2 times faster than DES
- Used in PGP
- Patented (expires in 2011)

Network Security

Cryptography Overview

27

The Advanced Encryption Standard (AES) Cipher - Rijndael

- Designed by Rijmen-Daemen (Belgium)
- Key size: 128/192/256 bit
- Block size: 128 bit data
- Properties: **iterative** rather than **Feistel** cipher
 - Treats data in 4 groups of 4 bytes
 - Operates on an entire block in every round
- Designed to be:
 - Resistant against known attacks
 - Speed and code compactness on many CPUs
 - Design simplicity

Network Security

Cryptography Overview

28

AES

- State: 16 bytes structured in a array

$S_{0,0}$	$S_{0,1}$	$S_{0,2}$	$S_{0,3}$
$S_{1,0}$	$S_{1,1}$	$S_{1,2}$	$S_{1,3}$
$S_{2,0}$	$S_{2,1}$	$S_{2,2}$	$S_{2,3}$
$S_{3,0}$	$S_{3,1}$	$S_{3,2}$	$S_{3,3}$

- Each byte is seen as an element of $\mathbf{F}_{2^8} = \text{GF}(2^8)$
 - \mathbf{F}_{2^8} finite field of 256 elements
 - Operations
 - Elements of \mathbf{F}_8 are viewed as polynomials of degree 7 with coefficients $\{0, 1\}$
 - Addition: polynomials addition \Rightarrow XOR
 - Multiplication: polynomials multiplication modulo $x^8 + x^4 + x^3 + x + 1$

Network Security

Cryptography Overview

29

AES Outline

- Initialize State $\leftarrow x \oplus \text{RoundKey}$;
- For each of the $N_r - 1$ rounds:
 - SubBytes(State);
 - ShiftRows(State);
 - MixColumns(State);
 - AddRoundKey(State);
- Last round:
 - SubBytes(State);
 - ShiftRows(State);
 - AddRoundKey(State);
- Output $y \leftarrow$ State

Network Security

Cryptography Overview

30

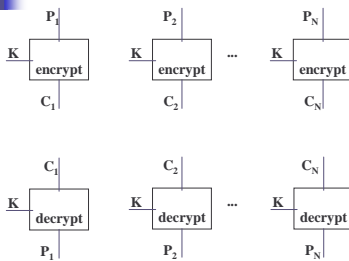
Implementation Aspects

- Can be efficiently implemented on 8-bit CPU
 - byte substitution works on bytes using a table of 256 entries
 - shift rows is a simple byte shifting
 - add round key works on byte XORs
 - mix columns requires matrix multiply in $GF(2^8)$ which works on byte values, can be simplified to use a table lookup

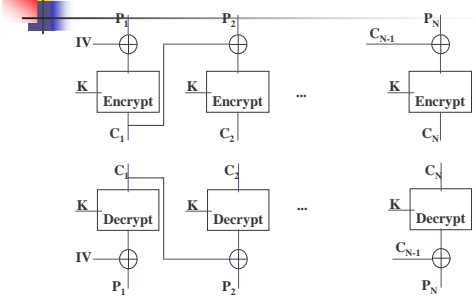
Implementation Aspects

- Can be efficiently implemented on 32-bit CPU
 - redefine steps to use 32-bit words
 - can pre-compute 4 tables of 256-words
 - then each column in each round can be computed using 4 table lookups + 4 XORs
 - at a cost of 16Kb to store tables
- Designers believe this very efficient implementation was a key factor in its selection as the AES cipher

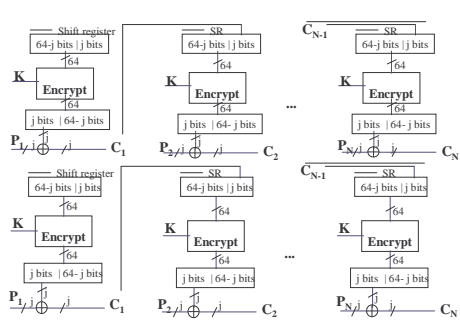
Encryption Modes: Electronic Codebook (ECB)



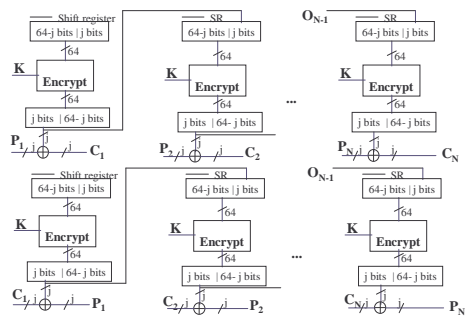
Encryption Modes: Cipher Block Chaining (CBC)



Encryption Modes: Cipher Feedback (CFB)



Encryption Modes: Output Feedback (OFB)



Counter (CTR)

- A "new" mode, though proposed early on
- Similar to OFB but encrypts counter value rather than any feedback value
- Must have a different key & counter value for every plaintext block (never reused)
 - $C_i = P_i \text{ XOR } O_i$
 - $O_i = \text{DES}_{K1}(i)$
- Uses: high-speed network encryptions, random access to files

Inside vs. Outside CBC-3DES

- What is the impact of using 3DES with CBC on the outside vs. inside?

Message Authentication Code (MAC) Using an Encryption Algorithm

- Also called Message Integrity Code (MIC)
- Goal:
 - Detect any modification of the content by an attacker
- Some techniques:
 - Use CBC mode, send only the last block (residue) along with the plaintext message
 - For confidentiality + integrity:
 - Use two keys (one for CBC encryption and one for CBC residue computation)
 - Append a cryptographic hash to the message before CBC encryption
 - New technique: use a Nested MAC technique such as HMAC



Hashes and Message Digests

- Goal:
 - Input: long message
 - Output: short block (called *hash* or *message digest*)
 - Property: given a hash h it is computationally infeasible to find a message that produces h
 - Examples: <http://www.slavasoft.com/quickhash/links.htm>
 - Secure Hash Algorithm (SHA-1, SHA-2) by NIST
 - MD2, MD4, and MD5 by Ron Rivest [RFC1319, 1320, 1321]
 - SHA-1: output 160 bits
 - SHA-2: output 256-384-512 believed to be more secure than others
 - Uses:
 - MAC: How? Problems? ... HMAC
 - Authentication: how?
 - Encryption: how?



HMAC

- $$\text{HMAC}_K(x) = \text{SHA-1}((K \oplus \text{opad}) \parallel \text{SHA-1}((K \oplus \text{ipad}) \parallel x))$$
 - $\text{ipad} = 3636\dots36$; $\text{opad} = 5\text{C}5\text{C}\dots5\text{C}$
 - Assumption:
 - SHA-1 restricted to one application is a secure MAC



Message Digest 5 (MD5) by R. Rivest [RFC1321]

- Input: message of arbitrary length
 - Output: 128-bit hash
 - Message is processed in blocks of 512 bits (padding if necessary)
 - Security:
 - Designed to resist to the Birthday attack
 - Collisions where found in MD5, SHA-0, and SHA-1
 - Near-Collisions of SHA-0
 - Eli Biham, Rafi Chen
 - Proceedings of Crypto 2004
 - <http://www.cs.technion.ac.il/~biham/publications.html>
 - Collisions for Hash Functions MD4, MD5, HAVAL-128 and RIPEMD
 - Xiaoyun Wang and Dengguo Feng and Xuejia Lai and Hongbo Yu
 - <http://eprint.iacr.org/2004/199.pdf>

Birthday Attacks

- Is a 64-bit hash secure?
 - Brute force: 1ns per hash => 10^{13} seconds over 300 thousand years
- But by **Birthday Paradox** it is not
- Example: what is the probability that at least two people out of 23 have the same birthday? $P > 0.5$
- Birthday attack technique**
 - opponent generates $2^{m/2}$ variations of a valid message all with essentially the same meaning
 - opponent also generates $2^{m/2}$ variations of a desired fraudulent message
 - two sets of messages are compared to find pair with same hash (probability > 0.5 by birthday paradox)
 - have user sign the valid message, then substitute the forgery which will have a valid signature
- Need to use larger MACs

Public Key Systems

Asymmetric cryptosystems

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



Modular Arithmetic

- Modular addition:
 - E.g., $3 + 5 = 1 \pmod{7}$
- Modular multiplication:
 - E.g., $3 * 4 = 5 \pmod{7}$
- Modular exponentiation:
 - E.g., $3^3 = 6 \pmod{7}$
- Group, Rings, Finite/Galois Fields ...

RSA Cryptosystem [RSA78]

- $E(M) = M^e \pmod{n = C}$ (Encryption)
- $D(C) = C^d \pmod{n = M}$ (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)
 - $d = e^{-1} \pmod{\phi(n)}$ (private, calculated)
- $D(E(M)) = M^{ed} \pmod{n} = M^{\phi(n)+1} = M$ (Euler's theorem)

Prime Numbers Generation

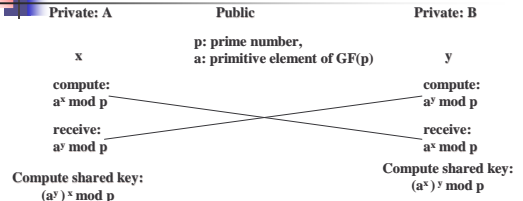
- Density of primes (prime number theorem):
 - $\pi(x) \sim x/\ln(x)$
- Sieve of Erathostène
 - Try if any number less than $\sqrt{RT(n)}$ divides n
- Fermat's Little Theorem does not detect Carmichael numbers
 - $b^{n-1} = 1 \pmod{n}$
- Solovay-Strassen primality test
 - If n is not prime then at least 50% of b fail to satisfy the following:
 - $b^{(n-1)/2} = \pm 1 \pmod{n}$
- Rabin-Miller primality test
 - If n is not prime then it is not pseudoprime to at least 75% of $b < n$:
 - $n-1 = 2^t \cdot b', b' = \pm 1 \pmod{n}$ or $(b'^2 = -1 \pmod{n}$ and $b'^{2^{t-1}} \neq \pm 1)$
 - probabilistic test, deterministic if the Generalized Riemann Hypothesis is true
- Deterministic polynomial time primality test [Agrawal, Kayal, Saxena 2002]

Use of RSA

- n Encryption (A wants to send a message to B):
 - n A uses the public key of B and encrypts M (i.e., $E_B(M)$)
 - n Since only B has the private key, only B can decrypt M (i.e., $M = D_B(E_B(M))$)

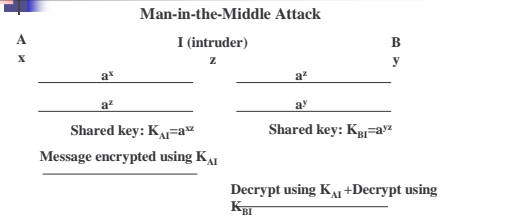
- n Digital signature (A want to send a signed message to B):
 - n Based on the fact that $E_A(D_A(M)) = D_A(E_A(M))$
 - n A encrypts M using its private key (i.e., $D_A(M)$) and sends it to B
 - n B can check that $E_A(D_A(M)) = M$
 - n Since only A has the decryption key, only can generate this message

Diffie-Hellman Key Exchange



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

Attack on Diffie-Hellman Scheme: Public Key Integrity



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

El Gamal Scheme

- Parameters:
 - p : prime number (public, chosen)
 - $g < p$: random number (public, chosen)
 - $x < p$: random number (private, chosen)
 - $y = g^x \text{ mod } p$ (public, computed)
- Encryption of message M :
 - choose random $k < p-1$
 - $a = g^k \text{ mod } p$
 - $b = y^k M \text{ mod } p$
- Decryption:
 - $M = b / y^k \text{ mod } p = b / g^{kx} \text{ mod } p = b / a^x$
- Message signature
 - choose random k relatively prime with $p-1$
 - find b : $M = (xa + kb) \text{ mod } (p-1)$ (extended Euclid algorithm)
 - signature(M) = (a, b)
 - verify signature: $y^a a^b \text{ mod } p = g^M \text{ mod } p$

Knapsack

- Introduced by R. Merkle
- Based on the difficulty of solving the Knapsack problem in polynomial time (Knapsack is an NP-complete problem)
 - cargo vector: $a = (a_1, a_2, \dots, a_n)$ (seq. Int)
 - plaintext msg: $x = (x_1, x_2, \dots, x_n)$ (seq. Bits)
 - ciphertext: $S = a_1x_1 + a_2x_2 + \dots + a_nx_n$
 - $a_i = wa'_i$ such that $a'_i > a'_1 + \dots + a'_{i-1}$, $m > a_1 + \dots + a_n$
 - w is relatively prime with m
- One-round Knapsack was broken by A. Shamir in 1982
- Several variations of Knapsack were broken

Others

- Elliptic Curve Cryptography (ECC)
- Zero Knowledge Proof Systems

Security Services

- Confidentiality:
 - Use an encryption algorithm
 - Generally a symmetric algorithm
- Integrity:
 - MAC algorithm
- Access control:
 - Use access control tables
- Authentication
 - Use authentication protocols
- Non-repudiation

Questions

- How many keys are derived in DES?
- Is the decryption process exactly the same as the encryption process?
- If a bit error occurs in the transmission of a ciphertext character in 8-bit CFB mode how far does it propagate?
