

Special Topic Communication Security

These slides are created by Dr. Huang of George Mason University. Students registered in Dr. Huang's courses at GMU can make a single machine readable copy and print a single copy of each slide for their own reference as long as the slide contains the copyright statement, and the GMU facilities are not used to produce the paper copies. Permission for any other use, either in machine-readable or printed form, must be obtained from the author in writing.

1

Common Senses

- ❑ System Safety
 - Check critical updates periodically
 - Every user account has a good password
 - Install virus checking software
 - Install a personal firewall
- ❑ Email Safety
 - Remove MS Outlook
 - Do not click on attachments, unless ...
- ❑ Recommendation:
 - Use Firefox as the browser

2

Security Requirements

- ❑ **Secrecy**: information be accessible for reading only by authorized parties
- ❑ **Authentication**: determining whom you are talking with
- ❑ **Integrity**: information modified/updated only by authorized parties
- ❑ **Availability**: information are available to authorized parties.

3

Enemy Tactics

- ❑ Passive attacks
 - interception
 - traffic analysis
- ❑ Active attacks
 - masquerade (one entity pretends to be a different entity)
 - replay
 - modification of messages
 - denial of service

4

Cryptography



- ❑ In many modern cryptography systems, the encryption/decryption algorithms are known to the public/enemy.
 - this enables large production volumes
- ❑ Security is based on the inherent difficulties in restoring the plaintext from a ciphertext without the key, not on the obscurity of the algorithms.

5

Traditional Cryptography: Substitution Ciphers

- ❑ Example:

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
q w e r t y u I o p a s d f g h j k l z x c v b n m

- ❑ “GMU” will be encrypted as “UDX”
- ❑ The substitution pattern can be parameterized by a key.

6

Transposition Ciphers

□ Example

| 7 | 4 | 5 | 1 | 2 | 8 | 3 | 6 |
|---|---|---|---|---|---|---|---|
| p | l | e | a | s | e | t | r |
| a | n | s | f | e | r | o | n |
| e | m | i | l | l | i | o | n |
| d | o | l | l | a | r | s | t |
| o | m | y | s | w | i | s | s |
| b | a | n | k | a | c | c | o |
| u | n | t | s | i | x | t | w |
| o | t | w | o | a | b | c | d |

- This is also an example of **block ciphers**, where the plaintext is divided into fixed-size blocks and encrypted one block at a time.

7

DES: Data Encryption Standard

- proposed by IBM and adopted by U.S. government in 1997
- a block cipher; block size is 64 bits
- the basic idea is to do a large number of different substitutions, transpositions and permutations

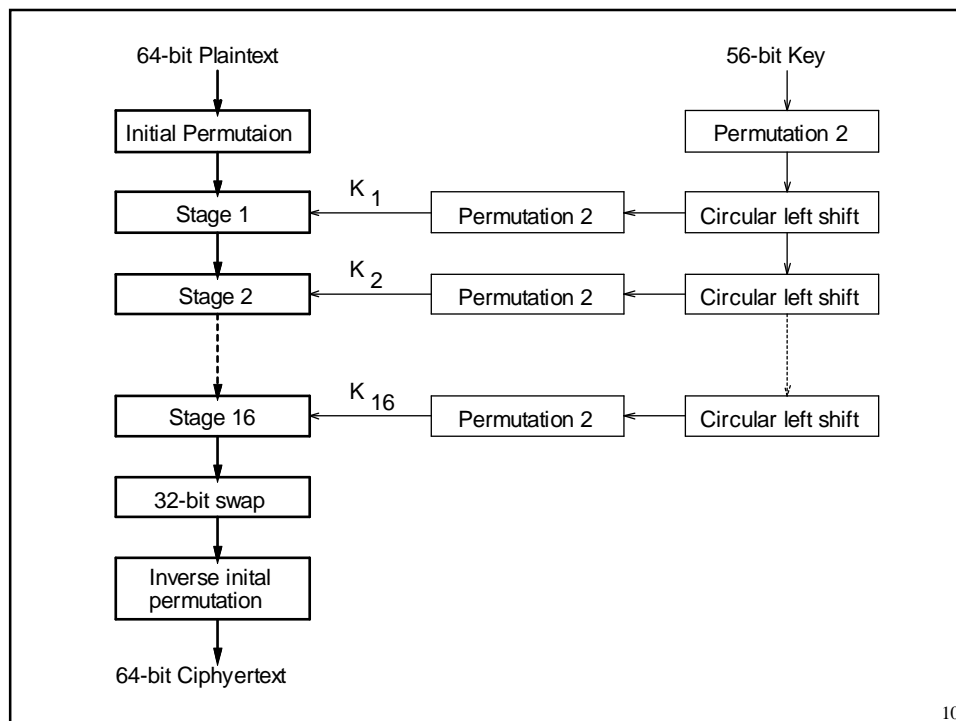
8

□ 19 distinct stages

- 3 stages of transposition or permutation independent of the key
- 16 more stages: each stage i is a combination of Exclusive OR with the key, K_i , a substitution function and a transposition function determined by K_i , plus a fixed-pattern permutation.

□ To decrypt, reverse the order of the 19 stages.

9



10

Is DES Safe ?

- ❑ With the increased computing powers, the initial key size of 56 bits causes concerns today.
- ❑ Other than that, there seem no *inherent* problems with the algorithm.

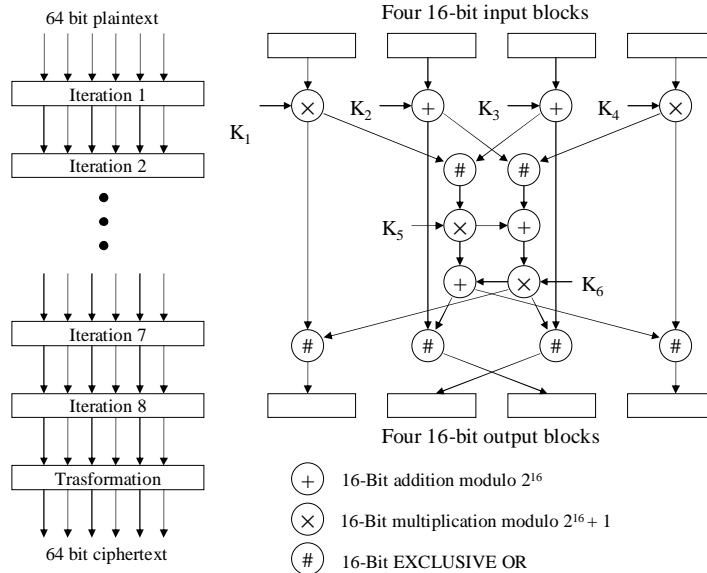
11

IDEA: International Data Encryption Algorithm

- ❑ uses 128-bit keys
- ❑ a block cipher; block size is 64 bits
- ❑ 8 iteration encryption; each iteration is a combination of integer addition, multiplications, exclusive OR, and permutation
- ❑ like DES, each iteration uses a different key derive from the input key

12

IDEA Block Diagram



13

RSA Public-Key Algorithm

- ❑ By Ron Rivest, Adi Shamir, and Len Adleman
- ❑ A block cipher in which the plaintext and ciphertext are treated integers between 0 and $n-1$.
- ❑ Each user
 - Choose two “large” primes, p and q ($\geq 10^{100}$)
 - Let $n=p \times q$ and $z=(p-1)(q-1)$
 - Choose a number e relatively prime to z .
 - Find d such that $e \times d = 1 \pmod{z}$.
 - The (e, n) pair is called the **public key**.
 - The (d, n) pair is called the **private key**.

14

Example

- Choose $p=7$ and $q=17$
 - We have $n=7 \times 17=119$ and $z=(7-1)(17-1)=96$
 - Let $e=5$ (note that $\gcd(5,96)=1$)
 - Let $d=77$ (note that $5 \times 77=385=4 \times 96+1$)
- $(5,119)$ is the public key of the user.
 - The key is made known to the world.
- $(77,119)$ is the private key of the user
 - The user keeps the key to him/herself.

15

Encryption and Decryption

- Consider a pair of public/private keys: (e,n) and (d,n) .
- To encrypt a plaintext block $M < n$: $C = M^e \bmod n$
- To decrypt a ciphertext block $C < n$: $M = C^d \bmod n$
- Why does this recover the original M ?

- You can reverse the roles of the two keys and get the same result.
- A pair of public/private keys always cancel out each other.

16

Using RSA to Ensure Secrecy

- When Alice sends a message M to Bob, she sends

$$C = E_{\text{public_key}(B)}(M)$$

- When Bob receives C , he recovers the plaintext by

$$M = D_{\text{private_key}(B)}(C)$$

- Only Bob can read the message because only he knows the private key.
- However, since everyone knows the Bob's public key, he cannot be sure if C is from Alice or not.

17

Authentication: RSA Digital Signature

- When Alice sends a message M to Bob, she sends

$$C = E_{\text{private_key}(A)}(M)$$

- When Bob receives C , he recovers M by

$$M = D_{\text{public_key}(A)}(C)$$

- The fact C can be decrypted by the public key of Alice proves that it is from Alice.
- However, since everyone knows Alice's public key, everyone can read the ciphertext C .

18

Providing Secrecy and Authentication

- When Alice sends a message M to Bob, she sends

$$C = E_{\text{public_key}(B)}(E_{\text{private_key}(A)}(M))$$

- When Bob receives C , he recovers M by

$$M = D_{\text{public_key}(A)}(D_{\text{private_key}(B)}(C))$$

- How is secrecy enforced ?
- How is authentication enforced ?

19

Discussion

- RSA is much slower than secret key algorithms.
 - It is in general 10 to 100 times slower than DES.
- Thus, it is appropriate only for encrypting critical information.
 - One common application of RSA is in the communication of a secret session key which is used to encrypt the bulk of data.

20

Message Authentication

- ❑ **Purpose:** to verify, upon the receipt of a message, that the source is authentic and the content has not been altered.
- ❑ Message authentication using encryption:
 - The fact a message can be successfully decrypted can be taken as a guarantee that the message is from the correct source.
 - However, cryptography algorithms are overkills when secrecy is not a concern.
 - It is desirable to develop efficient, authentication-only technologies.

21

One-Way Hash Functions

- ❑ A hash function H accepts a variable-size message M as input and produces a fixed-size block, $H(M)$, called the **digest** of M .
 - H must be computationally efficient.
 - For any digest d , it must be computationally infeasible to find M' such that $H(M')=d$.
 - For a message M , it is computationally infeasible to find M' such that $H(M')=H(M)$.
 - The last two properties make sure that faking a message is impossible.
- ❑ Example of hash functions: **MD5**
 - Digest size is 128 bits.

22

Message Authentication Using A Hash Function and Secret Key

- Alice wishes to send a message M to Bob, assuming that they both know a secret key K .
 - Alice computes digest $d=H(K \parallel M)$, where notation \parallel stands for concatenation.
 - Alice sends (M,d) to Bob.
 - Upon receiving (M,d) , Bob computes $H(K \parallel M)$ to check if the result matches d .

23

Message Authentication Using A Hash Function and Public Key

- Alice computes the digest of M and encrypts the digest using her private key, that is,

$$d' = E_{\text{private_key(A)}}(H(M)).$$

- Alice sends (M,d') to Bob.
- Upon receiving (M,d') , Bob checks whether or not

$$H(M) = D_{\text{public_key(A)}}(d')$$

- Notice that RSA is applied only to a small piece of data, namely the digest, thus will not cause performance problems

24

PGP: Pretty Good Privacy

- ❑ Developed by Phil Zimmermann.
- ❑ A complete email security package that provides secrecy, authentication, digital signatures, and compression.
- ❑ Freely available on almost all platforms.
- ❑ A technology NOT controlled by US government.

25

A Message M from Alice to Bob

- ❑ Alice computes the MD5 digest of M, call it D.
- ❑ Alice encrypts D using her private key, call the result D'.
- ❑ Alice produces $M' = M \parallel D'$.
- ❑ Alice compress M' by zip, call the result Z.
- ❑ Alice encrypts Z using IDEA with a randomly selected key K, call the result Z'.
- ❑ Alice encrypts K using Bob's public key, call the result K'.
- ❑ Alice computes and sends the Base64 representation of $K' \parallel Z'$.

26

- ❑ Upon reception, Bob converts the Base64 representation to its original form.
- ❑ How does Bob read the message ?

- ❑ How does Bob authenticate the message ?