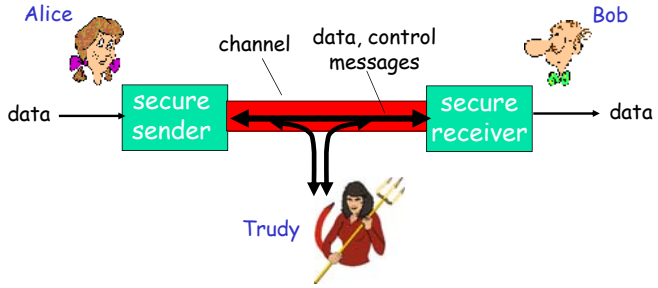
# Principles of Network Security

Kai Shen



## The Network Security Model

- Bob and Alice want to communicate "securely".
- Trudy (the adversary) has access to the channel.






## Who might Bob and Alice be?

- Web browser/server for electronic transactions (e.g., on-line purchases/banking)
- DNS servers
- Routers exchanging routing table updates
- … well, *real-life* Bobs and Alices!



## What can an adversary do?

- Eavesdrop: understand the content of messages
- Actively changing messages
- Impersonation: fake (spoof) identity
- Denial of service: prevent service from being used by others (e.g., by overloading resources)

## What is Network Security?

Confidentiality: only sender, intended receiver should "understand" message contents.

Authentication: sender, receiver want to confirm identity of each other.

Message Integrity: sender, receiver want to ensure message not altered (in transit, or afterwards).

Access and Availability: services must be accessible and available to (and only to) legitimate users.



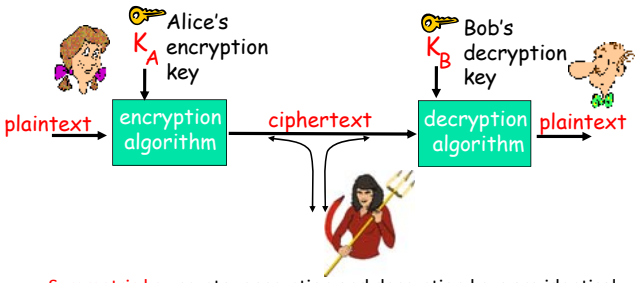
## Principles of Network Security

- Confidentiality: cryptography
- Authentication
- Integrity
- Key distribution and certification



## The Language of Cryptography

First goal of cryptography: confidentiality.




- Symmetric key crypto: encryption and decryption keys are identical. (both are secret)
- Public key crypto: encryption key is public, decryption key is secret.



## Symmetric Key Cryptography: Monoalphabetic Cipher

Monoalphabetic cipher: substitute one letter for another.

```
plaintext:  abcdefghijklmnopqrstuvwxyz
            |                       |
ciphertext: mnbvcxzasdfghjklpoiuytrewq
```

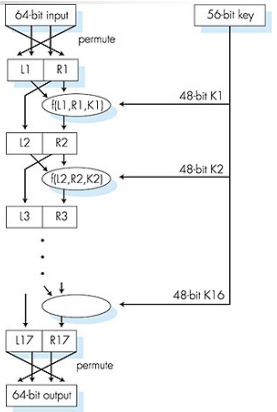
Example: `Plaintext: bob. i love you. alice`
`ciphertext: nkn. s gktc wky. mgsbc`

Q1: How hard to break this simple cipher?
- brute force?
- other?

Q2: How to make it more difficult to break?

## Symmetric Key Cryptography: DES

- DES: Data Encryption Standard
  - US encryption standard [NIST 1993]
  - 56-bit symmetric key, 64-bit plaintext input
  - encryption: initial permutation ⇒ 16 "rounds", each using different 48 bits of key ⇒ final permutation
  - decryption: reverse operation using the same key
- How secure is DES?
  - DES Challenge (1999): 56-bit-key-encrypted phrase decrypted (brute force) in 22 hours 15 minutes
- Making DES more secure:
  - use three keys sequentially (3-DES)
  - use more bits






## AES: Advanced Encryption Standard

- Newer (Nov. 2001) symmetric-key NIST standard, replacing DES
- Processes data in 128 bit blocks
- 128, 192, or 256 bit keys
- Brute force decryption (try each key) taking 1 sec on DES, would take 149 trillion years for 128-bit AES
- Cost
  - 128-bit AES takes a few hundred microseconds to encrypt 4KB data on modern mobile processors
- Block cypher



## Stream Cypher

- Much faster than block cypher like AES
- Fixed-size seed key K expanded (through some kind of shuffled bit shifts) to an infinite key stream C(K)
- For data X, it uses a portion of the key stream (equal length to data X), then xor the data to produce encrypted data
- Same key cannot be used twice, otherwise

  Encrypted(X) = X xor C(K)

  Encrypted(Y) = Y xor C(K)

  ⇨ Encrypted(X) xor Encrypted(Y) = X xor Y

  ⇨ If you know X (or Y), then you know the other

  ⇨ Even if you know neither, you can guess X/Y quite well from X xor Y if X and Y are in a natural language



## Stream Cypher

- How to guarantee the same key is not reused?
  - A shared master key known secretly by the two sides
  - An initialization vector, communicated publicly for each session, augment the master key to create the seed key and key stream for the session
- WEP wireless encryption
  - Employ an initialization vector of 24-bit, but insufficient (doesn't take long to observe sessions with the same initialization vector)

Sources: http://en.wikipedia.org/wiki/Stream_cipher_attack

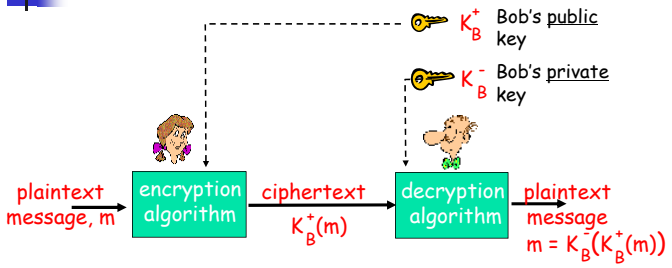
## Public Key Cryptography

Symmetric key cryptography
- requires sender, receiver know shared secret key
- Q: how to agree on key in the first place? (particularly difficult if Trudy is eavesdropping on all communication)

Public key cryptography
- encryption key is different from decryption key
- encryption key is public, known to everyone, also called public key
- decryption key is secret, known only to receiver, also called private key



## Public Key Cryptography




Principle for choosing the public/private key pair:
One should not be able to derive the private key from the public key.



## Public Key Cryptography: RSA (Ron Rivest, Adi Shamir and Len Adleman)

- Choosing keys:
  - Choose two large prime numbers p, q. (e.g., 1024 bits each)
  - Compute n = pq, z = (p-1)(q-1)
  - Choose e (with e<n) that has no common factors with z
  - Choose d such that ed-1 is exactly divisible by z
  - Public key is (n,e). Private key is (n,d)
- To encrypt a message, m (<n): do $c = m^e \bmod n$
- To decrypt a received ciphertext, c: do $m = c^d \bmod n$
- Reason: for any m (relatively prime with n)
  - $m^z \bmod n = 1$; therefore $m^{ed-1} \bmod n = 1$
- Another property: $(m^d \bmod n)^e \bmod n = m$
- RSA is much slower than the symmetric key cryptos (block and certainly stream cyphers)



## Disclaimer

- Parts of the lecture slides contain original work of James Kurose, Larry Peterson, and Keith Ross. The slides are intended for the sole purpose of instruction of computer networks at the University of Rochester. All copyrighted materials belong to their original owner(s).