CS 3251- Computer Networks 1: Security

Professor Patrick Traynor

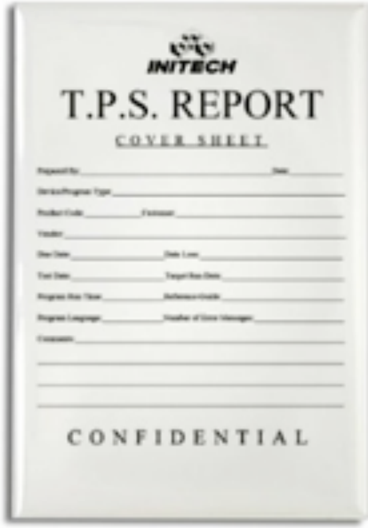
11/14/13

Lecture 24

Announcements

- Homework 3 due 11/19
  - One week from today.
  - We will try to turn this around ASAP for the final.

- Project 4 is due 11/28
  - Last check - did everyone partner-up?
  - You should be moving ahead with this. Do not wait to get started!



Last Time

- We talked about mobility as a matter of context:
  - How is mobility handled as you move around a room? Between rooms in the same building? As your drive down The Connector at 75 MPH?
- Core routers in the Internet could support mobility.
  - Why don't we do this?
- What are the tradeoffs between direct and indirect routing schemes?
- What are the equivalents of HAs and FAs in a cellular network?

Chapter 8: Network Security

Chapter goals:

- understand principles of network security:
  - cryptography and its many uses beyond "confidentiality"
  - authentication
  - message integrity
  - key distribution
- security in practice:
  - firewalls
  - security in application, transport, network, link layers

Chapter 8 roadmap

8.1 What is network security?

8.2 Principles of cryptography

8.3 Authentication

8.4 Integrity

8.5 Key Distribution and certification

8.6 Access control: firewalls

8.7 Attacks and counter measures

8.8 Security in many layers

What is network security?

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

- sender encrypts message
- receiver decrypts message

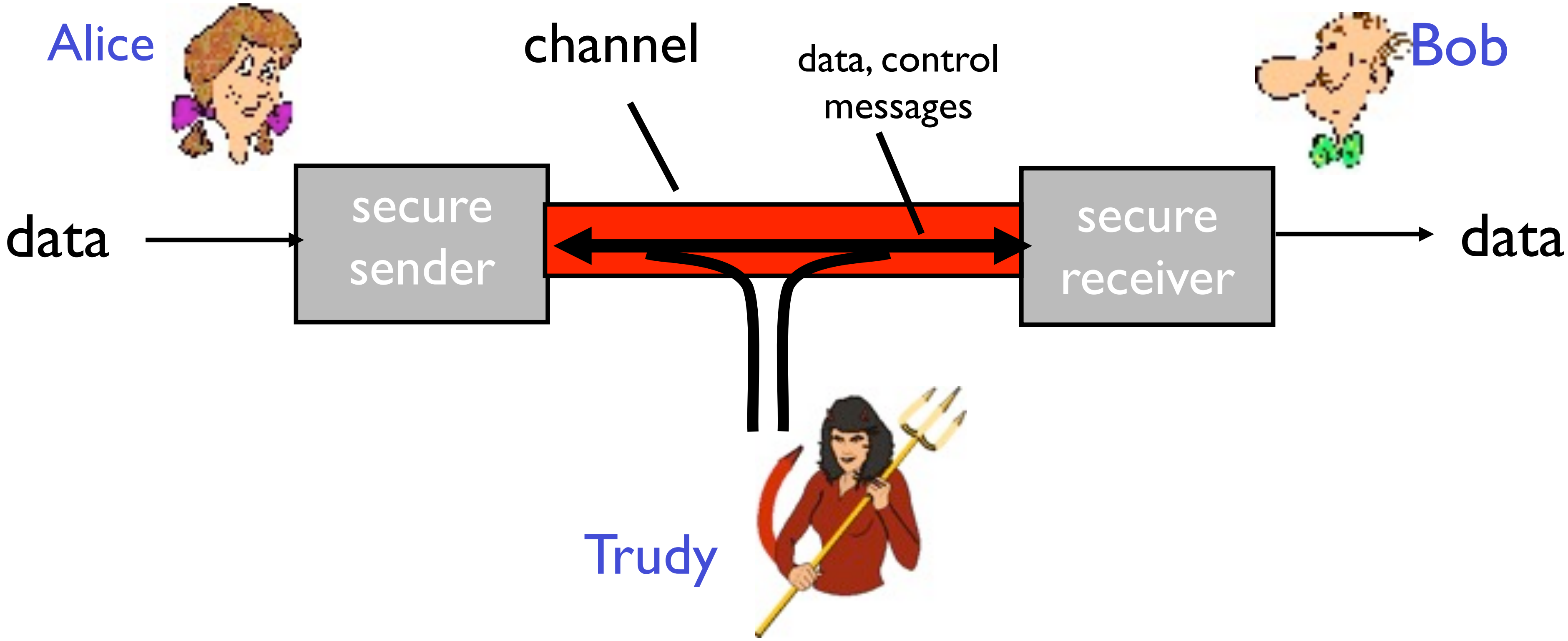
Authentication: sender, receiver want to confirm identity of each other

Message Integrity: sender, receiver want to ensure message not altered (in transit, or afterwards) without detection

Access and Availability: services must be accessible and available to users

Friends and enemies: Alice, Bob, Trudy

- well-known in network security world
- Bob, Alice want to communicate "securely"
- Trudy (intruder) may intercept, delete, add messages



Who might Bob, Alice be?

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

There are bad guys (and girls) out there!

Q: What can a "bad guy" do?

A: a lot!

- eavesdrop: intercept messages
- actively insert messages into connection
- impersonation: can fake (spoof) source address in packet (or any field in packet)
- hijacking: "take over" ongoing connection by removing sender or receiver, inserting himself in place
- denial of service: prevent service from being used by others (e.g., by overloading resources)

more on this later ……

Chapter 8 roadmap

8.1 What is network security?

8.2 Principles of cryptography

8.3 Authentication

8.4 Integrity

8.5 Key Distribution and certification

8.6 Access control: firewalls

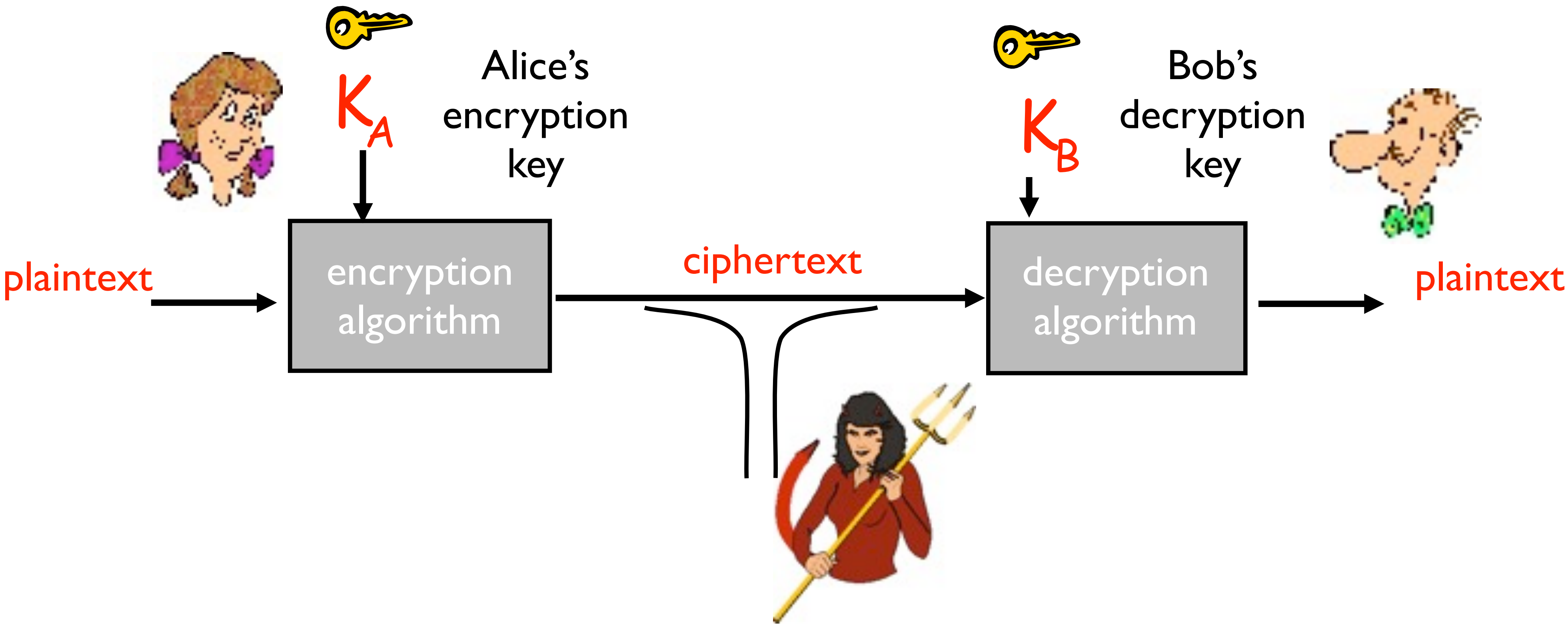
8.7 Attacks and counter measures

8.8 Security in many layers

Notes on Cryptography

- Cryptography in and of itself *is not security*. It is a tool that helps us achieve security.
  - Think of this as the difference using a hammer and designing a building (an architect probably needs to be skilled in both).
- Do not, under any circumstances, attempt to "roll your own" crypto.
  - In the last 40 years, cryptography has become a science. Most work before this time can be broken with ease.
- You must assume that your enemy knows the algorithm you are using.
  - It must be secure anyhow. This is Kerchoffs' Principle.

The language of cryptography



symmetric key crypto: sender, receiver keys identical

public-key crypto: encryption key *public*, decryption key *secret* (private)

Caesar Cipher

- The earliest know encryption scheme, this cipher simply shifts all letters in the alphabet to the right by 3 places.

  abcdefghijklmnopqrstuvwxyz
  defghijklmnopqrstuvwxyzabc

- KRZ ZHOO GRHV WKLV ZRUN?

- This example belongs to a basic class of rotation substitution ciphers.
  - e.g., ROT3, ROT13

Symmetric key cryptography

substitution cipher: substituting one thing for another

- monoalphabetic cipher: substitute one letter for another

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

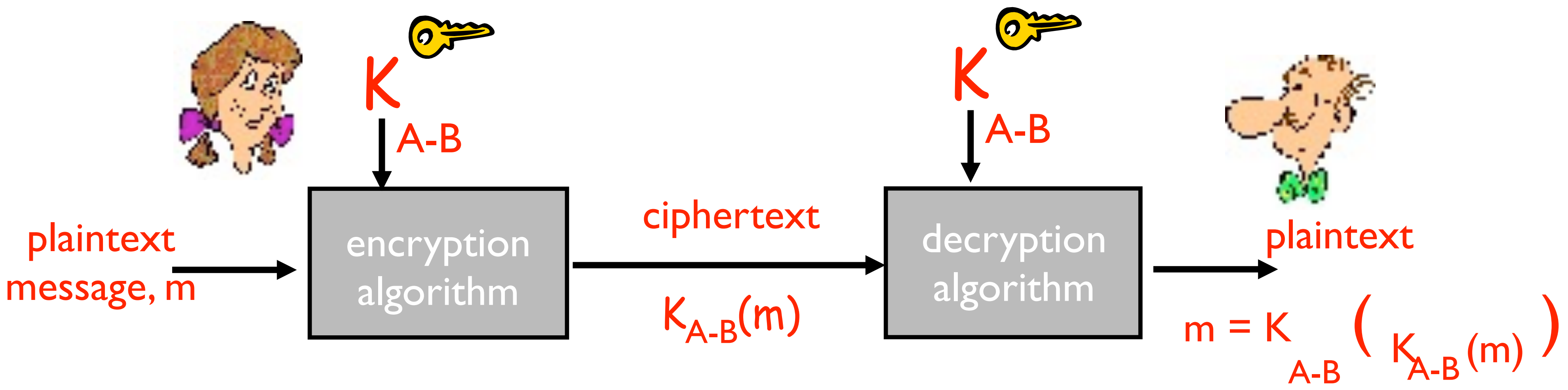
E.g.:

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

Q: How hard to break this simple cipher?:

- brute force (how hard?)
- other?

Symmetric key cryptography



symmetric key crypto: Bob and Alice share know same (symmetric) key: K_{A-B}

- e.g., key is knowing substitution pattern in mono alphabetic substitution cipher
- Q: how do Bob and Alice agree on key value?

Symmetric key crypto: DES

DES: Data Encryption Standard

- US encryption standard [NIST 1993]
- 56-bit symmetric key, 64-bit plaintext input
- How secure is DES?
  - DES Challenge: 56-bit-key-encrypted phrase ("Strong cryptography makes the world a safer place") decrypted (brute force) in 4 months
  - no known "backdoor" decryption approach
- making DES more secure:
  - use three keys sequentially (3-DES) on each datum
  - use cipher-block chaining

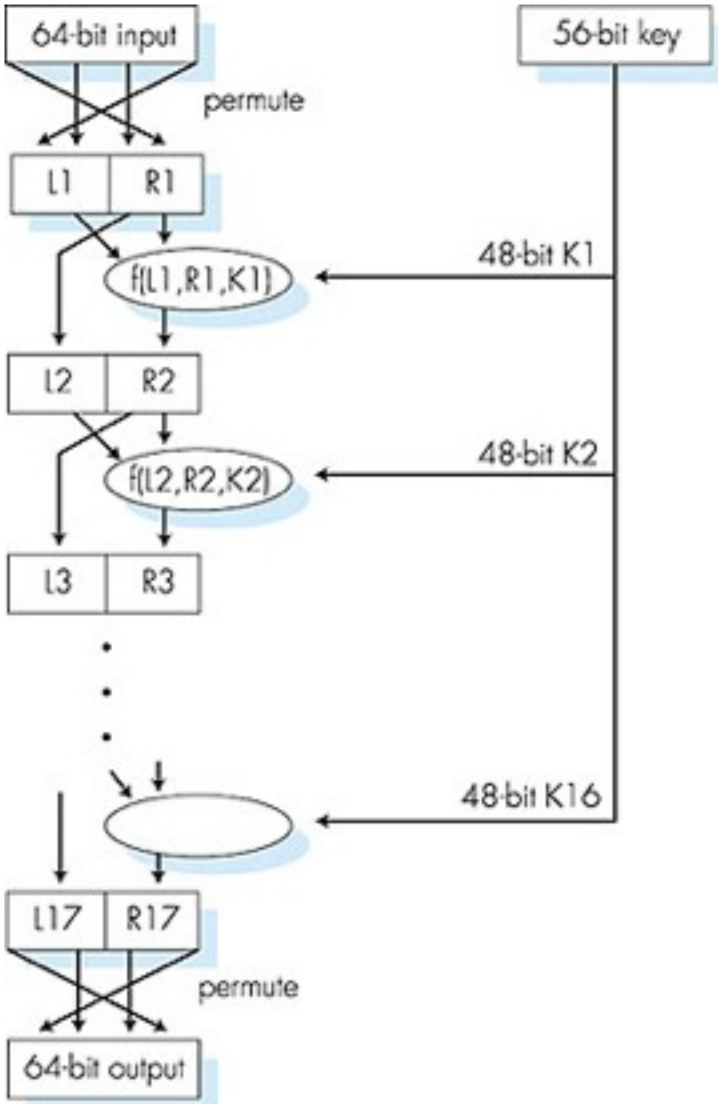
Symmetric key crypto: DES

DES operation

initial permutation

16 identical "rounds" of function application, each using different 48 bits of key

final permutation



AES: Advanced Encryption Standard

- new (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, takes 149 trillion years for AES

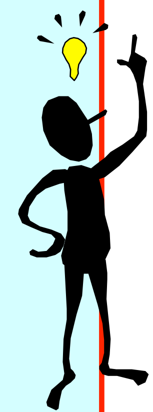
Public Key Cryptography

symmetric key crypto

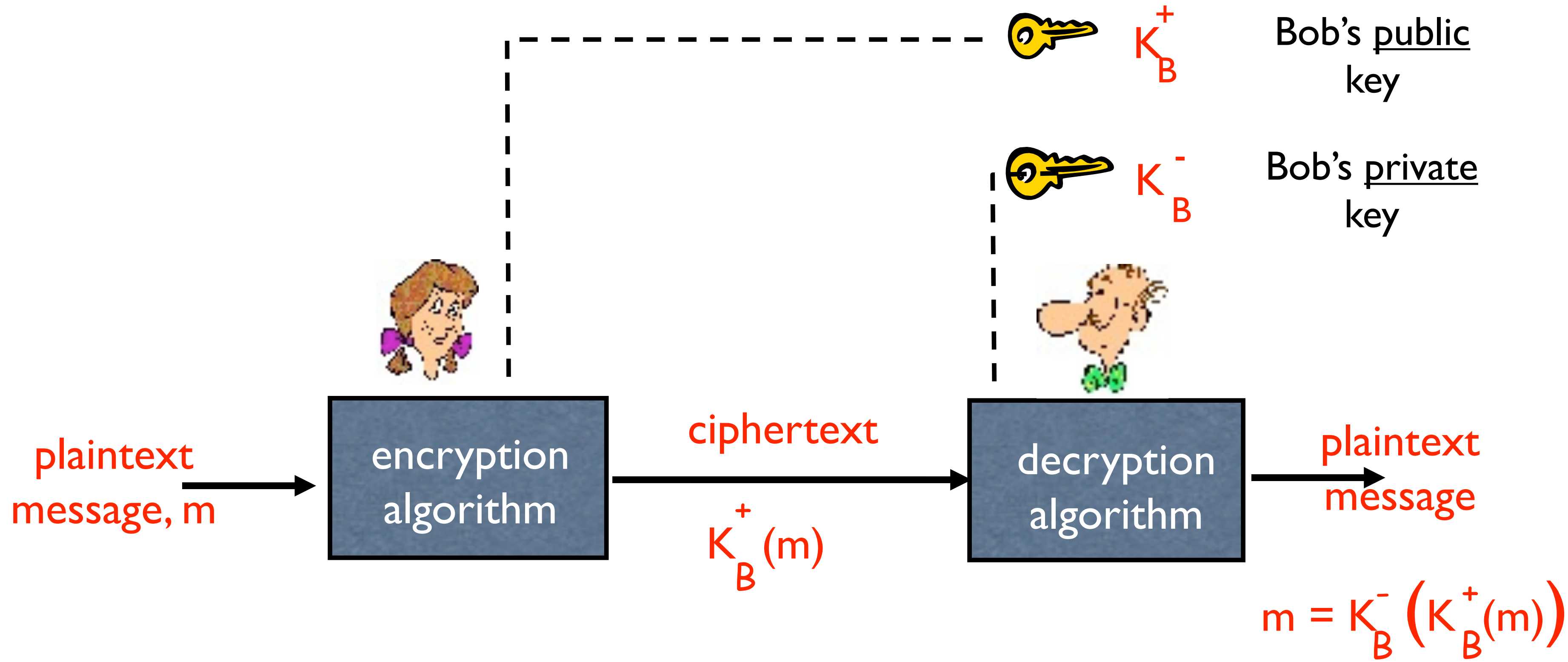
- requires sender, receiver know shared secret key
- Q: how to agree on key in first place (particularly if never "met")?

public key cryptography

- radically different approach [Diffie-Hellman76, RSA78]
- sender, receiver do not share secret key
- public encryption key known to all
- private decryption key known only to receiver



Public key cryptography



Public key encryption algorithms

Requirements:

1. need $K_B^+(\cdot)$ and $K_B^-(\cdot)$ such that

$$K_B^-(K_B^+(m)) = m$$

2. given public key K_B^+, it should be impossible to compute private key K_B^-

RSA: Rivest, Shamir, Adelson algorithm

RSA: Choosing keys

1. Choose two large prime numbers p, q. (e.g., 1024 bits each)

2. Compute n = pq, z = (p-1)(q-1)

3. Choose e (with e<n) that has no common factors with z. (e, z are "relatively prime").

4. Choose d such that ed-1 is exactly divisible by z. (in other words: ed mod z = 1).

5. Public key is $\underbrace{(n,e)}_{K_B^+}$. Private key is $\underbrace{(n,d)}_{K_B^-}$.

RSA: Encryption, decryption

0. Given (n,e) and (n,d) as computed above

1. To encrypt bit pattern, m, compute $c = m^e \bmod n$
   (i.e., remainder when m^e is divided by n)

2. To decrypt received bit pattern, c, compute

$$m = c^d \bmod n$$

(i.e., remainder when c^d is divided by n)

Magic happens!

$$m = (\underbrace{m^e \bmod n}_{c})^d \bmod n$$

RSA example:

Bob chooses p=5, q=7. Then n=35, z=24.

e=5 (so e, z relatively prime).
d=29 (so ed-1 exactly divisible by z).

	letter	m	m^e	$c = m^e \bmod n$
encrypt:	l	12	248832	17

	c	c^d	$m = c^d \bmod n$	letter
decrypt:	17	481968572106750915091411825223071697	12	l

Now You Try

- Choosing Keys
  - $p = 7$ and $q = 11$
  - $n = pq = ?$; $z = (p-1)(q-1) = ?$
  - Choose e (no common factors with z): 7
  - Choose d such that $7 \times d = 1 \bmod z$: 43
- Our message is 9, what is the ciphertext?
- Our ciphertext is 37, what is the message?

$c = m^e \bmod n$

$m = c^d \bmod n$

RSA: another important property

The following property will be very useful later:

$$\underbrace{K_B^-(K_B^+(m))}_{} = m = \underbrace{K_B^+(K_B^-(m))}_{}$$

use public key first, followed by private key

use private key first, followed by public key

Result is the same!

Why is RSA secure?

- suppose you know Bob's public key (n,e). How hard is it to determine d?
- essentially need to find factors of n without knowing the two factors p and q
- fact: factoring a big number is hard

Diffie-Hellman

- Encrypting data with RSA is computationally expensive
  - Much faster to use a symmetric cipher
- Later, we will see examples where a symmetric key is choosen by sender and encrypted by RSA
- ... but, what about network communications?
  - We want two hosts to agree on a symmetric key
- Public key crypto was really started by Diffie and Hellman
  - Goal: negotiate a secret over an insecure (e.g., public) medium
    - seems impossible

Diffie-Hellman Protocol

- We have two participants: Alice (A) and Bob (B)
- Setup: pick a prime number p and a base g ($<p$)
  - This information is public, e.g., $p=13$, $g=4$
- Step 1: Each principal picks a private value x ($<p-1$)
- Step 2: Each principle generates and communicates a new value:

$$y = g^x \bmod p$$

- Step 3: Each principle generates the secret shared key z

$$z = y^x \bmod p$$

- z is used as the symmetric key for communication

Diffie-Hellman: Exchange

- Public Info: p = 17, g = 5
- Everyone choose an x

Alice picks
her x: S_A

$T_A = g^{SA}$ mod p

$T_B = g^{SB}$ mod p

Bob picks
his x: S_B

Alice computes:
$K = T_B^{SA}$ mod p

$T_B^{SA} = (g^{SB})^{SA} = g^{SB\ SA} = g^{SA\ SB} = (g^{SA})^{SB} = T_A^{SB}$ mod p

Alice computes:
$K = T_A^{SB}$ mod p

Encrypted communication with K

- How does Alice know Bob sent T_A?
  - Stay tuned ...

Next Time

- Read Sections 8.3 and 8.4
  - Authentication
- Start working on the homework and the last project.