本讲主题

网络安全基本概念

什么是网络安全？

- 网络安全是指网络系统的硬件、软件及其系统中的数据受到保护，不因偶然的或者恶意的原因而遭受到破坏、更改、泄露，系统连续可靠正常地运行，网络服务不中断。

网络安全基本属性

机密性(confidentiality): 只有发送方与预定接收方能够理解报文内容

- 发送方加密报文
- 接收方解密报文

身份认证(authentication): 发送方与接收方希望确认彼此的真实身份

信息完整性(message integrity): 发送方与接收方希望确保信息未被篡改（传输途中或者后期），发生篡改一定会被检测到

可访问与可用性(access and availability): 网络服务必须对被授权用户可访问与可用

网络安全的基本特征

- 相对性
  - 只有相对的安全，没有绝对的安全
- 时效性
  - 新的漏洞与攻击方法不断发现
- 相关性
  - 新配置、新系统组件可能会引入新的安全问题
- 不确定性
  - 攻击时间、攻击者、攻击目标和攻击发起的地点都具有不确定性
- 复杂性
  - 网络安全是一项系统工程，需要技术的和非技术的手段
- 重要性
  - 网络安全关乎国家、政府、企业、个人的安全

网络安全

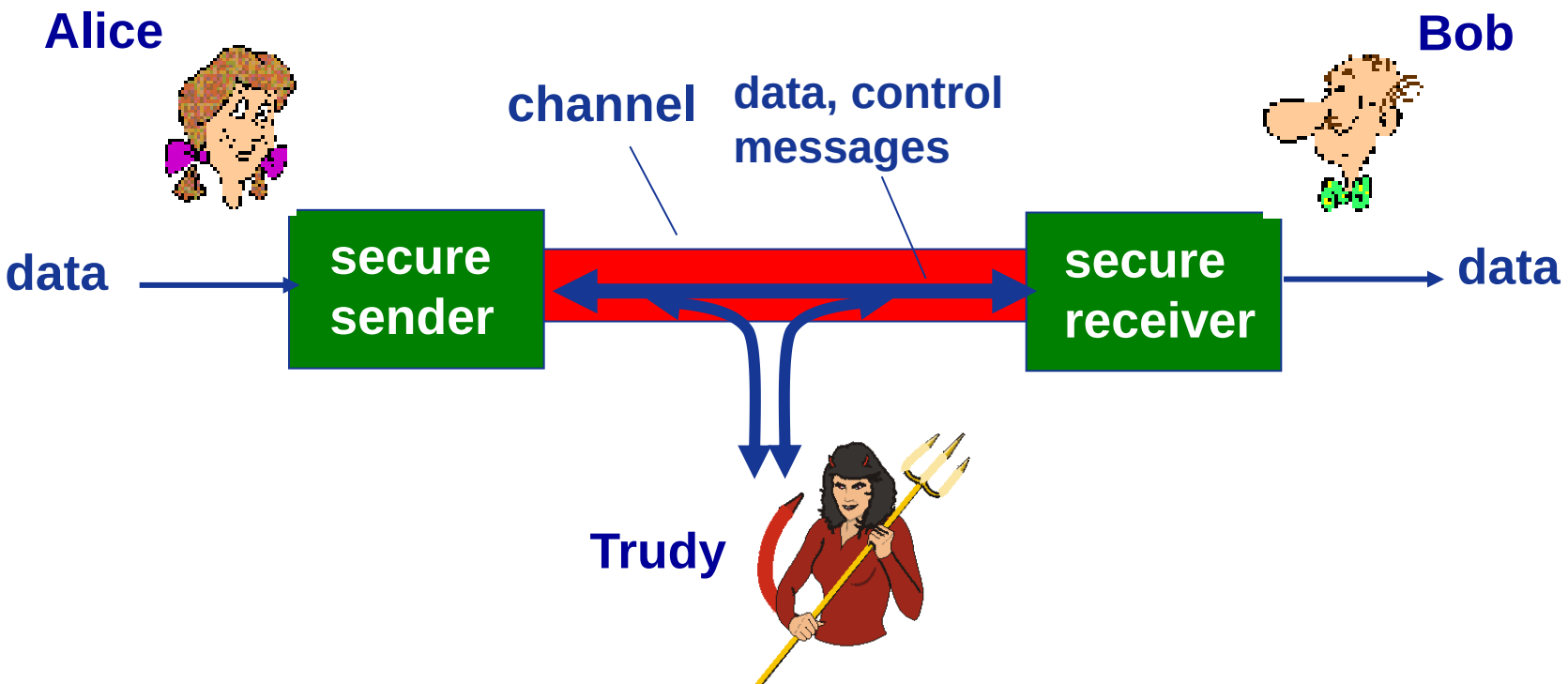
- 网络安全研究领域:
  - 入侵者（bad guys）如何攻击计算机网络
  - 如何防护网络对抗攻击
  - 如何设计网络体系结构免疫（immune）攻击
- Internet最初设计几乎没考虑安全性
  - 最初愿景:“一组彼此信任的互助用户连接到一个透明网络”进行信息共享 ☺
  - Internet协议设计者扮演了“追赶者”（catch-up）角色
  - 网络安全需要在网络各个层次考虑!

本讲主题

网络安全拟人模型

拟人场景: Alice、Bob、Trudy

- 网络安全领域的著名拟人模型
- Bob与Alice是期望进行安全通信的情侣
- Trudy是企图破坏Bob和Alice通信的入侵者 (intruder)，试图拦截、删除或添加信息



网络中的Bob、Alice？

Bob、Alice：

- 电子交易过程的Web浏览器/服务器 (e.g., 网购)
- 网络银行的客户/服务器
- DNS服务器
- 路由器之间交换路由表更新
- ……

网络中的Trudy?

Trudy：Bad Guys

- 通过Internet向主机植入恶意软件（malware）
  - 病毒（virus）
  - 蠕虫（worm）
  - 间谍软件（spyware）：记录键盘输入、web站点访问、向收集站点上传信息等
  - ……
- 被感染主机可能加入僵尸网络（botnet），用于发送垃圾邮件、DDoS攻击等

本讲主题

网络安全威胁（1）

“坏蛋”们可能做什么？

Q: “坏蛋”们能做什么?

A: 很多！

- 窃听(eavesdrop): 窃听信息
- 插入(insert):主动在连接中插入信息
- 假冒(impersonation): 可以通过伪造(spoof)分组中的源地址(或者分组的任意其他字段）
- 劫持(hijacking): 通过移除/取代发送发或者接收方“接管”(take over)连接
- 拒绝服务DoS(denial of service): 阻止服务器为其他用户提供服务(e.g., 通过过载资源)

Internet安全威胁

映射(Mapping):

- 发起攻击前:“探路”(case the joint) – 找出网络上在运行什么服务
- 利用ping命令确定网络上主机的地址
- 端口扫描(Port-scanning): 依次尝试与每个端口建立TCP连接
- nmap (http://www.insecure.org/nmap/)，广为使用的国外端口扫描工具之一

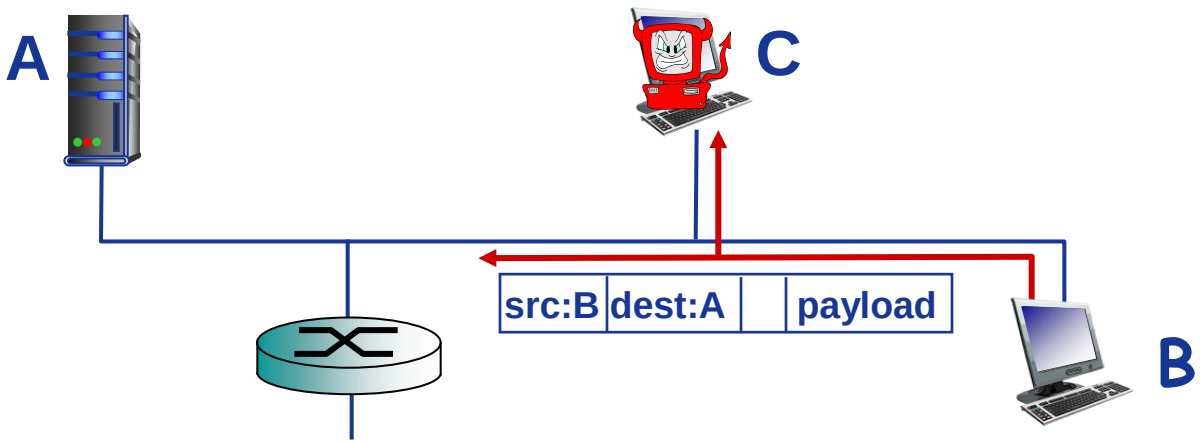
对策(Countermeasures)?

- 记录到达的网络流量
- 分析、识别出可疑活动(IP地址和端口被依次扫描)

Internet安全威胁

分组“嗅探”(sniffing):

- 广播介质(共享式以太网，无线网络)
- 混杂(promiscuous)模式网络接口可以接收/记录所有经过的分组/帧
- 可以读到所有未加密数据(e.g., 包括口令！)

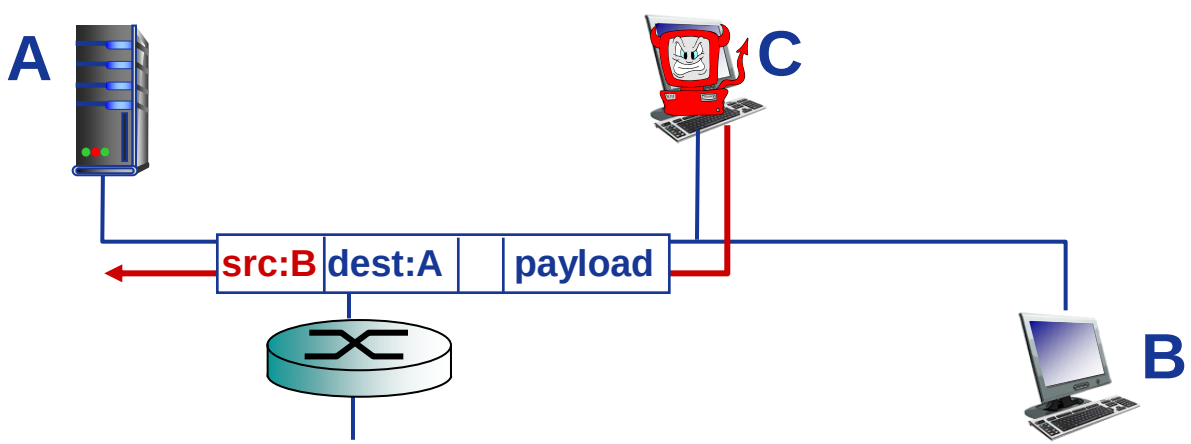


- Wireshark就是一个典型免费的分组嗅探软件

Internet安全威胁

IP欺骗(Spoofing):

- 直接由应用生成“原始”IP分组，可以设置分组的源IP地址字段为任意值
- 接收方无法判断源地址是否被欺骗
- e.g.: C冒充B



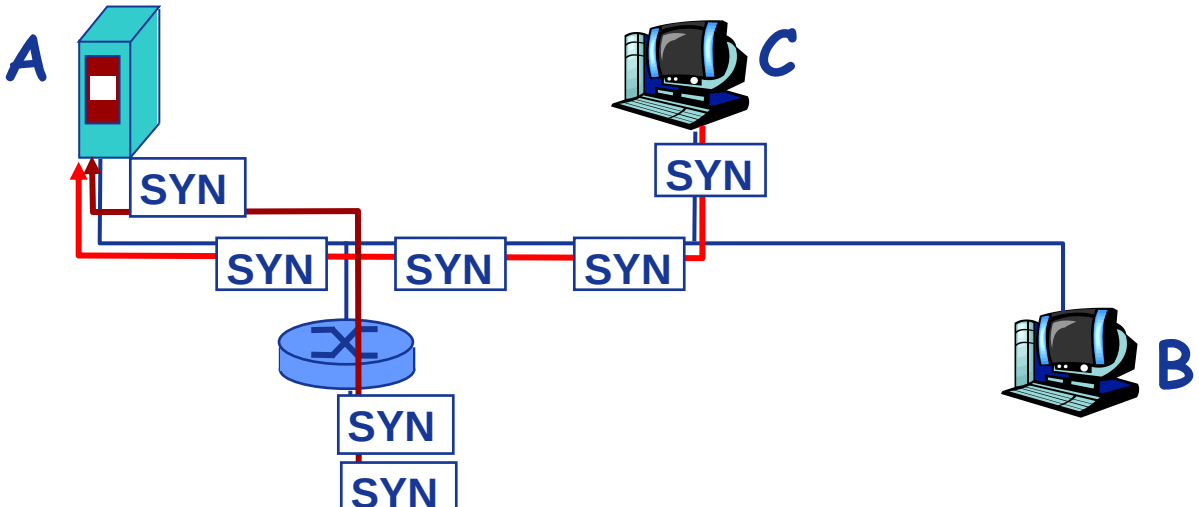
本讲主题

网络安全威胁（2）

Internet安全威胁

拒绝服务DOS(Denial of service):

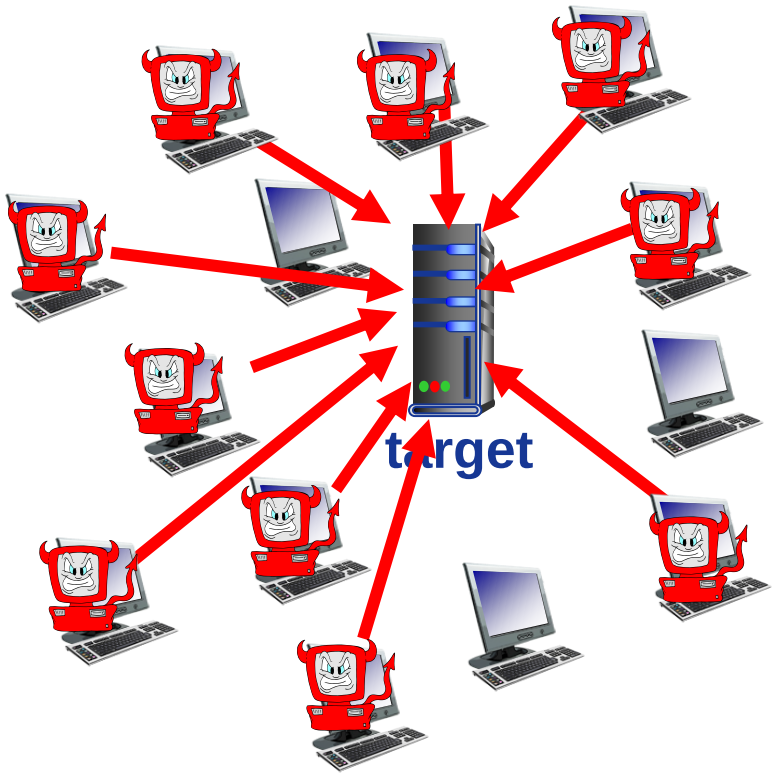
- 向接收方恶意泛洪(flood)分组，淹没(swamp)接收方
  - 带宽耗尽
  - 资源耗尽
- 分布式拒绝服务攻击 (DDOS): 多个源主机协同淹没接收方
- e.g., C与另一个远程主机协同对A进行SYN攻击



Internet安全威胁

DDoS攻击过程:

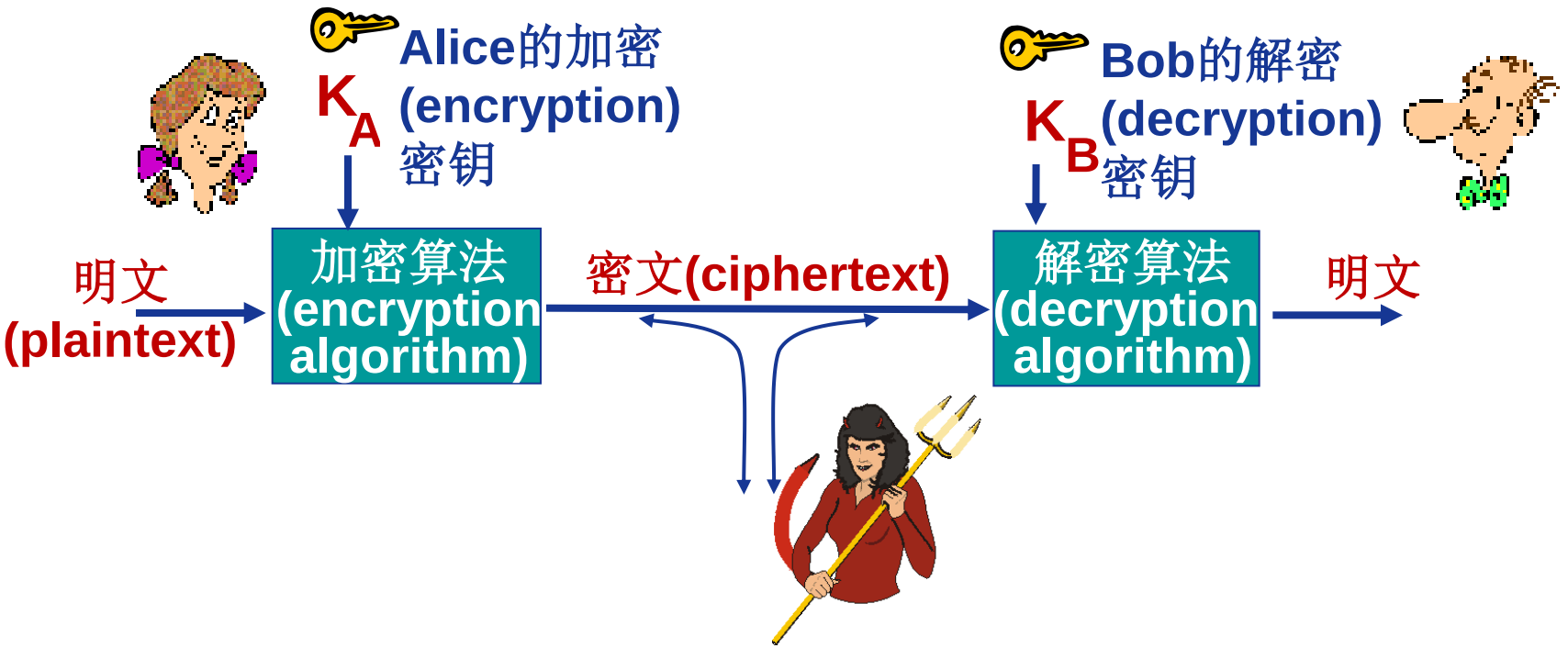
1. 选择目标
2. 入侵(break into)网络中主机（构建僵尸网络）
3. 控制僵尸主机向目标发送分组



本讲主题

密码学基础（1）

密码学(cryptography)术语

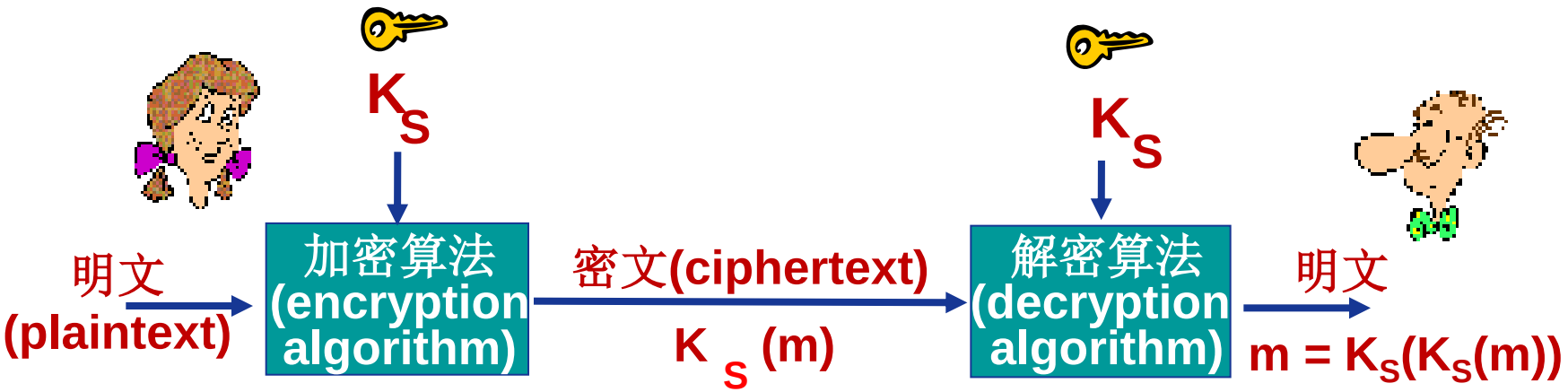


m：明文

$K_A(m)$：密文，利用秘钥K_A加密

$m = K_B(K_A(m))$：利用秘钥K_B解密

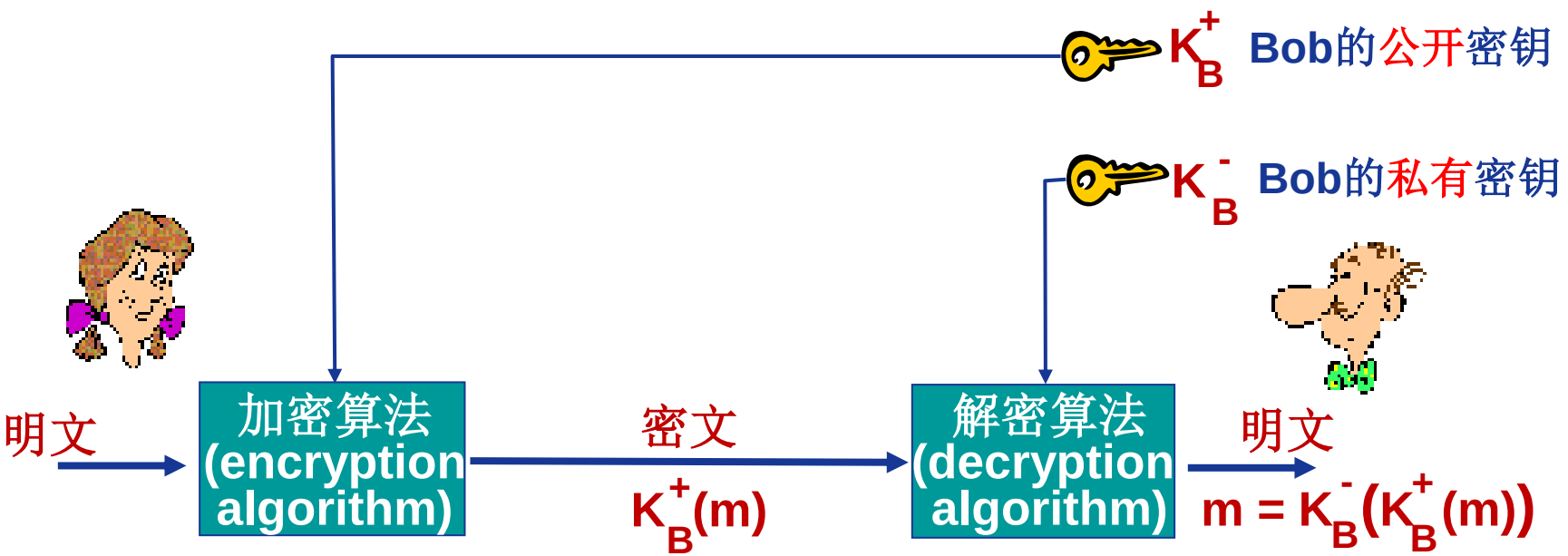
对称密钥加密



对称密钥加密: Bob和Alice共享相同(对称)密钥: K_S

Q: Bob和Alice如何确认密钥值（密钥分发）？

公开密钥加密


K_B^+ Bob的公开密钥
K_B^- Bob的私有密钥
明文
加密算法 (encryption algorithm)
密文
$K_B^+(m)$
解密算法 (decryption algorithm)
明文
$m = K_B^-(K_B^+(m))$

破解加密方法

- 唯密文攻击(cipher-text only attack): 入侵者(如Trudy)只截获到密文，基于对密文的分析进行破解
- 两条途径:
  - 暴力破解(brute force): 尝试所有可能的密钥
  - 统计分析
- 已知明文攻击(known-plaintext attack): 入侵者已知(部分)明文以及与之匹配的密文
  - e.g., 在单码替代密码(monoalphabetic cipher)中，入侵者已确认字母a,l,i,c,e,b,o的替换关系
- 选择明文攻击(chosen-plaintext attack): 入侵者可以获取针对选择的明文的密文

本讲主题

密码学基础（2）

传统加密方法

替代密码(substitution cipher): 利用一种东西替代另一种东西

- 凯撒密码(Casesar cipher)：一个字母替代另一个字母
  - 将一个字母利用字母表中该字母后面的第*k*个字母替代
  - 如*k=3*，“bob. i love you. alice”→“ere, l oryh brx. dolfh”
- 单码(字母)替代密码(monoalphabetic cipher)

明文:abcdefghijklmnopqrstuvwxyz

密文:mnbvcxzasdfghjklpoiuytrewq

e.g.: 明文: bob. i love you. alice

密文: nkn. s gktc wky. mgsbc

加密秘钥: 26个字母集合向26个字母集合的映射

传统加密方法

替代密码(substitution cipher): 利用一种东西替代另一种东西

- 多码(字母)替代加密(polyalphabetic encryption)：使用多个单码替代密码，明文中不同位置的字母使用不同的单码替代密码
- 例如，使用采用两个凯撒密码的多码替代加密：

Plaintext letter:	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
$C_1(k = 5)$:	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e
$C_2(k = 19)$:	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s

加密秘钥: **(C_1=5, C_2=19); C_1, C_2, C_2, C_1, C_2;……**

明文：bob. i love you.

密文：ghu. n etox dhz.

本讲主题

密码学基础（3）

现代加密技术

- 现代加密技术的基本操作包括经典的替代和置换
  - 不再针对一个个字母，而是针对二进制位操作
- 现代加密技术主要分为：
  - 对称密钥加密
  - 非对称密钥加密（公开密钥加密）

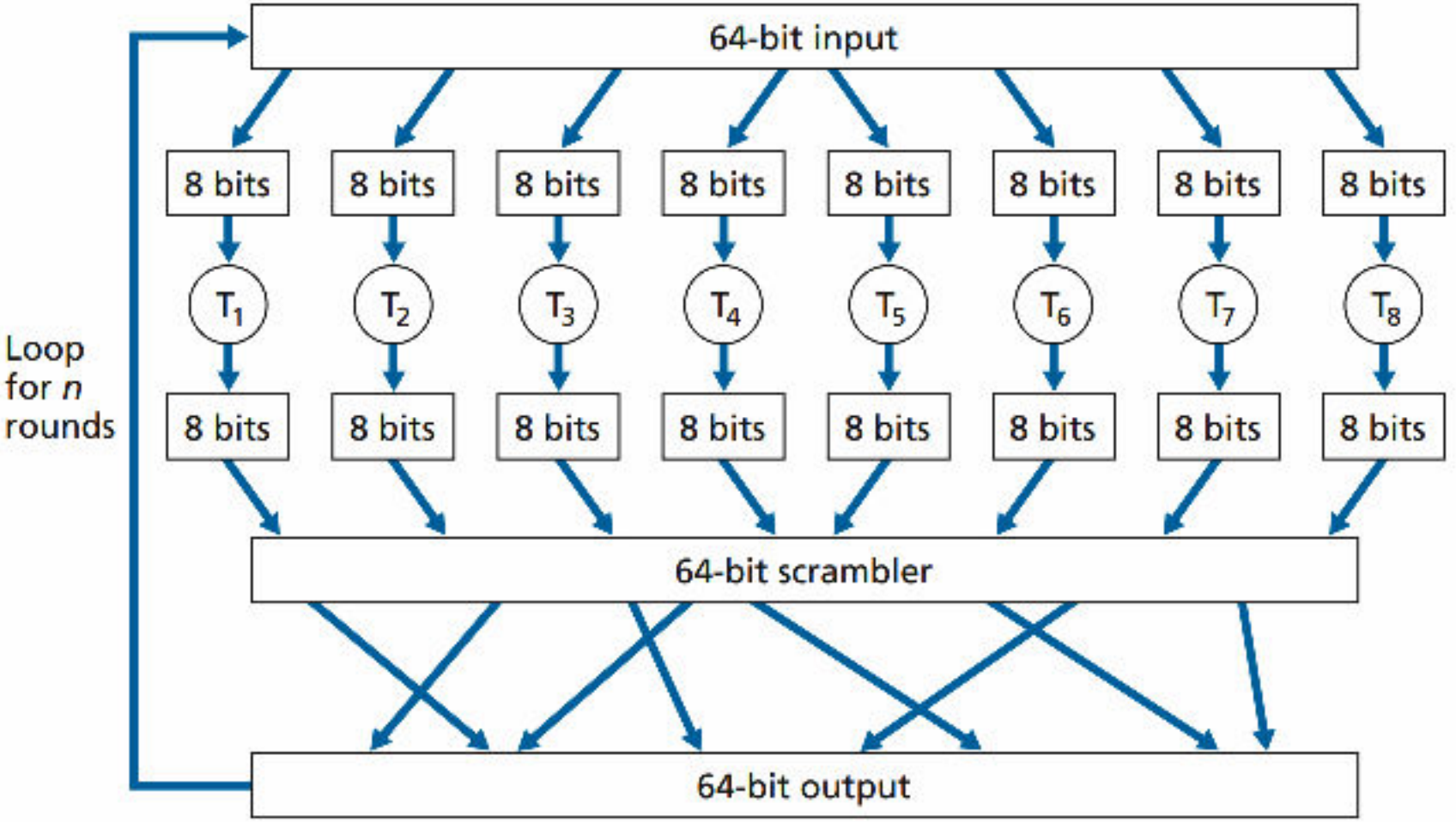


Figure 8.5 ♦ An example of a block cipher

公钥密码学

对称密钥加密：

- 需要发送方与接收方知道共享的秘密密钥
- Q: 最初如何商定密钥（尤其“素未谋面”）？

公开密钥加密

- 完全不同的方法 **[Diffie-Hellman76, RSA78]**
- 发送方与接收方无需共享秘密密钥
- 公开密钥（公钥）完全公开
- 私有密钥（私钥）只有接收方知道

公钥加密算法

需求:

① 公钥加密 $K_B^+(-)$ 和私钥解密 $K_B^-(-)$ 需要满足:

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

② 给定公钥 K_B^+，不可能计算得到私钥 K_B^-

RSA: Rivest, Shamir, Adelson algorithm

前提条件：模运算

- x mod n = x除以n的余数
- 事实上：

  $[(a \bmod n) + (b \bmod n)] \bmod n = (a+b) \bmod n$

  $[(a \bmod n) - (b \bmod n)] \bmod n = (a-b) \bmod n$

  $[(a \bmod n) * (b \bmod n)] \bmod n = (a*b) \bmod n$
- 因此：

  $(a \bmod n)^d \bmod n = a^d \bmod n$
- 例如：x=14, n=10, d=2，则

  $(x \bmod n)^d \bmod n = 4^2 \bmod 10 = 6$

  $x^d = 14^2 = 196 \quad x^d \bmod 10 = 6$

RSA: 预备知识

- 报文/信息(message): 仅仅是一个比特模式(bit pattern)
- 每个比特模式可以表示为一个唯一的整数
- 因此，加密一个报文就等价于加密一个数

例如:

- m= 10010001，可以唯一地表示为十进制数145
- 为了加密m，我们可以加密对应的数(145)，得到一个新的数（即密文）

RSA: 生成公钥/私钥对

1. 选择2个大质数*p*和*q*。(e.g., 1024bits的大质数)

2. 计算$n = pq$，$z = (p-1)(q-1)$

3. 选择*e* (满足$e<n$)，使e与z 之间没有公因子，
   即*e*, *z*互质(relatively prime)

4. 选择*d*使得*ed*-1刚好可以被*z*整除，
   (即: $ed \bmod z = 1$).

5. 公钥：$\underbrace{(n,e)}_{K_B^+}$；私钥：$\underbrace{(n,d)}_{K_B^-}$.

RSA: 加密、解密

0. 给定公钥 (n,e)和私钥(n,d)

1. 加密报文m $(m<n)$时，计算

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

2. 解密密文c时，计算

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

不可思议
事情发生!

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

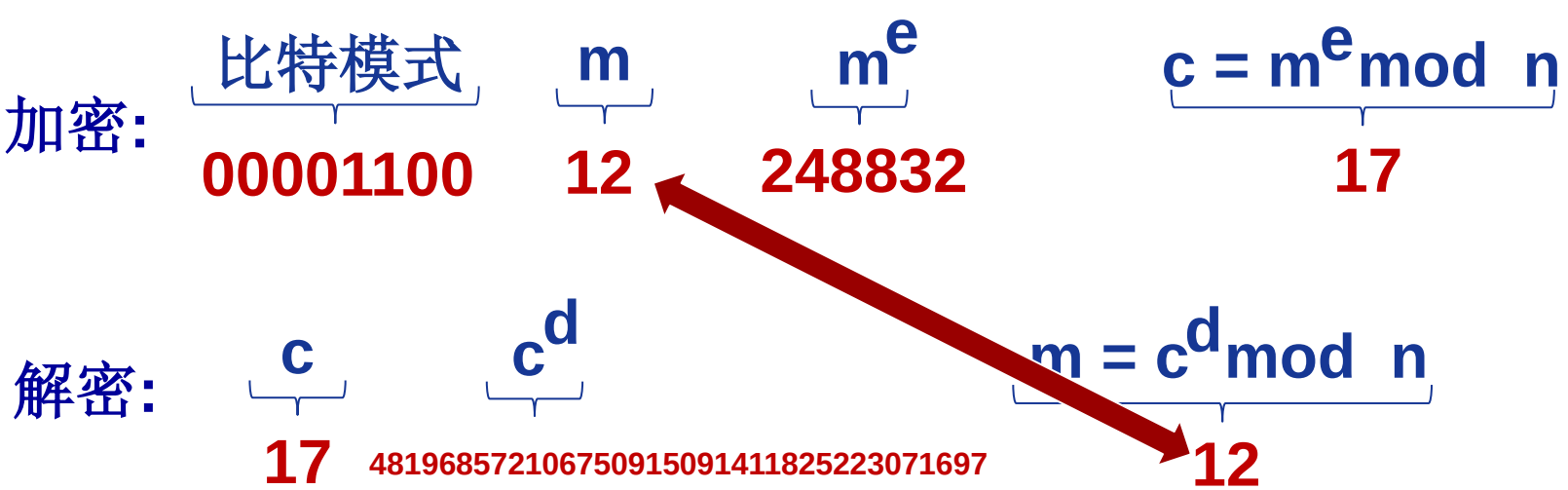
RSA举例

Bob选择p=5, q=7. 于是n=35, z=24.

e=5 (e, z互质).

d=29 (ed-1刚好被z整除).

加密8-bit报文（e.g. 1个字符）。



本讲主题

密码学基础（8）

RSA的理论依据?

- 必须满足：$c^d \bmod n = m$，其中$c = m^e \bmod n$
- 可以证明：对于任意x和y，有$x^y \bmod n = x^{(y \bmod z)} \bmod n$
  - 其中$n = pq$，$z = (p-1)(q-1)$
- 因此：

$$
\begin{aligned}
c^d \bmod n &= (m^e \bmod n)^d \bmod n \\
&= m^{ed} \bmod n \\
&= m^{(ed \bmod z)} \bmod n \\
&= m^1 \bmod n \\
&= m
\end{aligned}
$$

RSA: 另一个重要性质

下列性质将非常重要:

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

利用公钥加密，可以利用私钥解密

利用私钥加密，可以利用公钥解密

结果相同!

为什么？

$$(m^e \bmod n)^d \bmod n = m^{ed} \bmod n$$

$$= m^{de} \bmod n$$

$$= (m^d \bmod n)^e \bmod n$$

RSA为什么安全?

- RSA的安全性建立在“大数分解和素性检测”这个数论难题的基础上
  - 既将两个大素数相乘在计算上容易实现，而将该乘积分解的计算量相当大
- 假设已知Bob的公钥(n,e)，那么有多大难度确定d，即私钥(n,d)?
- 本质上需要在不知道两个因子p和q的前提下，找出n的因子
  - 分解一个大数是很困难的!

RSA的实际应用

- RSA的幂运算强度很大
- DES至少比RSA快100倍
- 实际应用中：
  - 利用公钥加密建立安全连接，然后建立第二个密钥-对称会话密钥，用于加密数据

会话密钥(session key, K_S)

- Bob与Alice利用RSA交换对称会话密钥K_S
- 一旦双方确认K_S，则利用会话密钥加密/解密会话数据

本讲主题

身份认证

身份认证(Authentication)

目标: Bob希望Alice“证明”她的身份

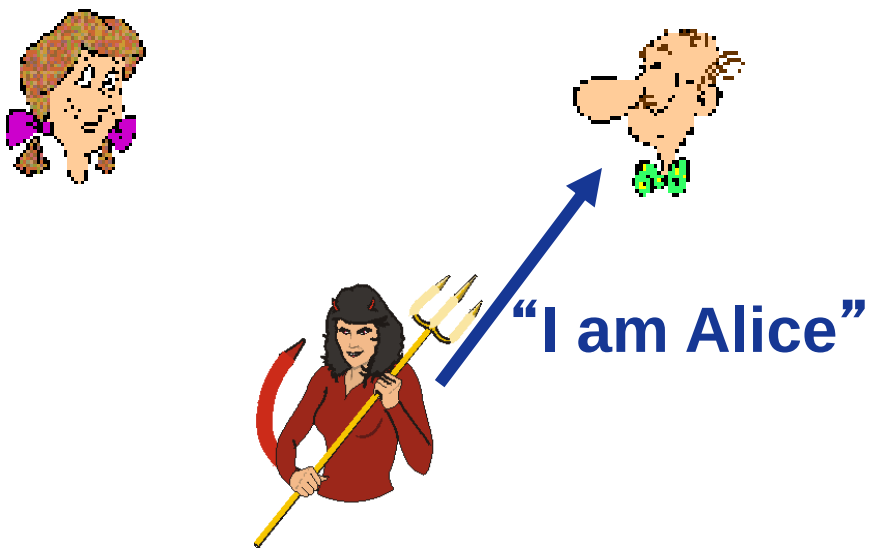
协议ap1.0: Alice声明“I am Alice”

失效场景??

身份认证

目标: Bob希望Alice“证明”她的身份

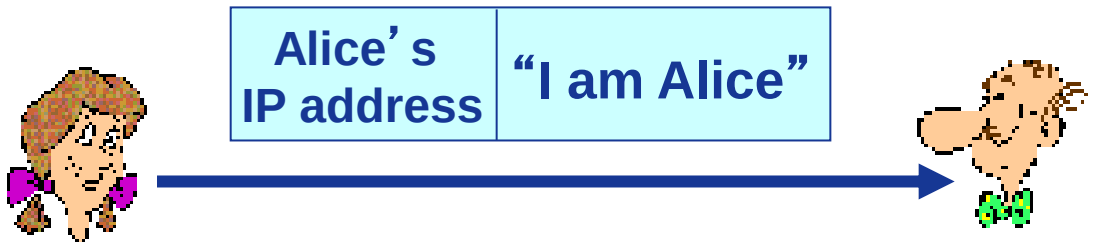
协议ap1.0: Alice声明“I am Alice”



在网络中,Bob“看”不到 Alice, 因此Trudy可以简单地声明她就是Alice！

身份认证

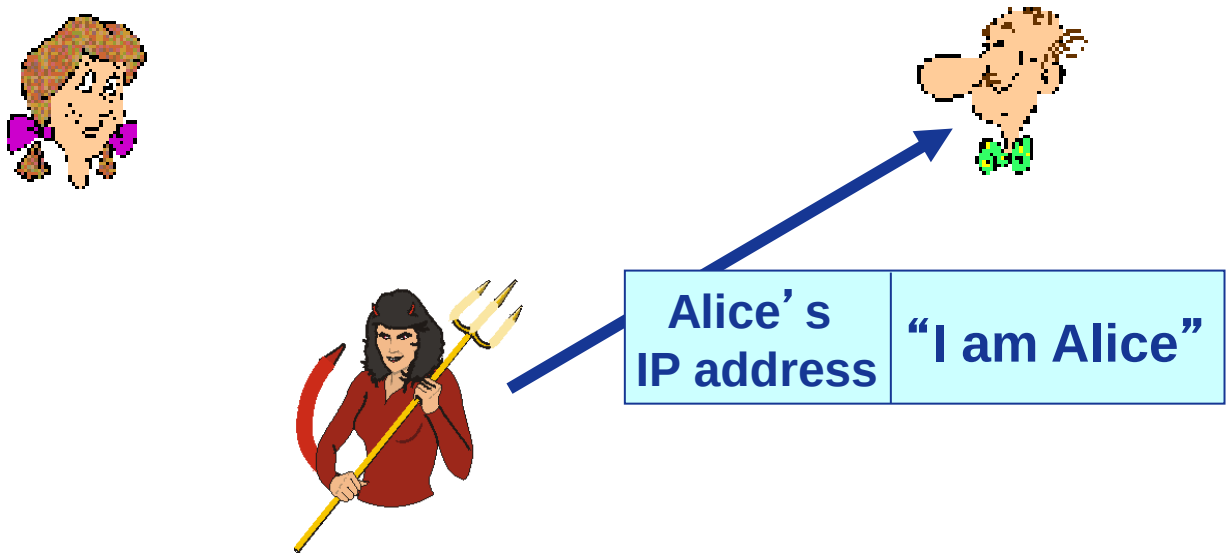
协议ap2.0: Alice在IP分组中声明“I am Alice”，IP分组包含Alice的源IP地址



失效场景??

身份认证

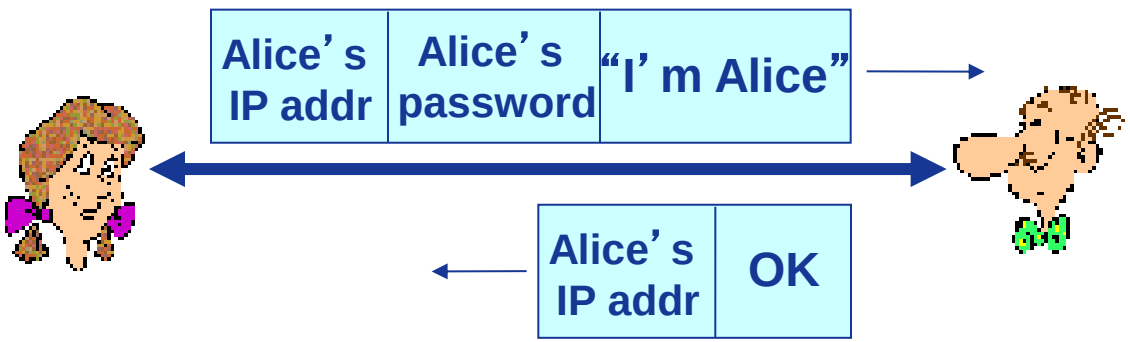
协议ap2.0: Alice在IP分组中声明“I am Alice”，IP分组包含Alice的源IP地址



Trudy可以构造一个分组，“欺骗”为Alice的IP地址

身份认证

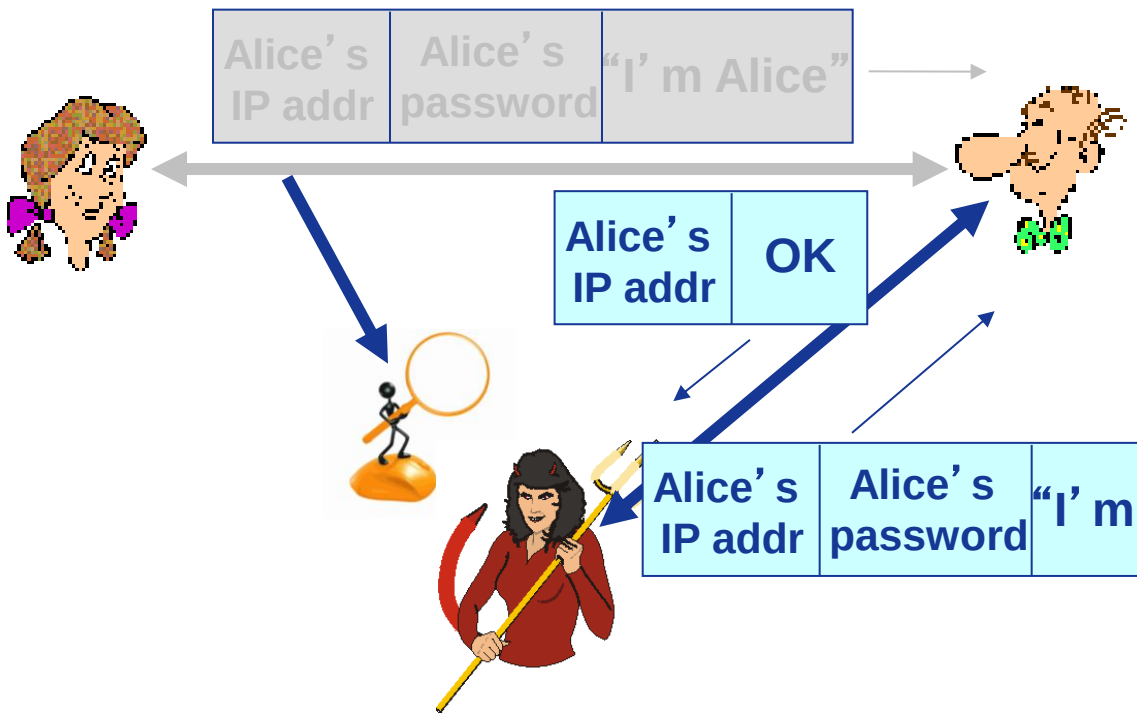
协议ap3.0: Alice声明“I am Alice”的同时，发送她的秘密口令进行“证明”.



失效场景??

身份认证

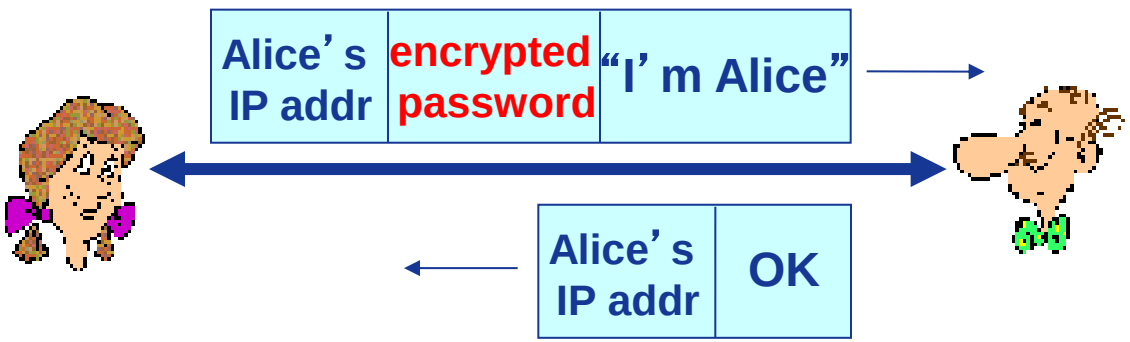
协议ap3.0: Alice声明“I am Alice”的同时，发送她的秘密口令进行“证明”.



嗅探(sniffing): Trudy嗅探Alice 的分组，提取口令

身份认证

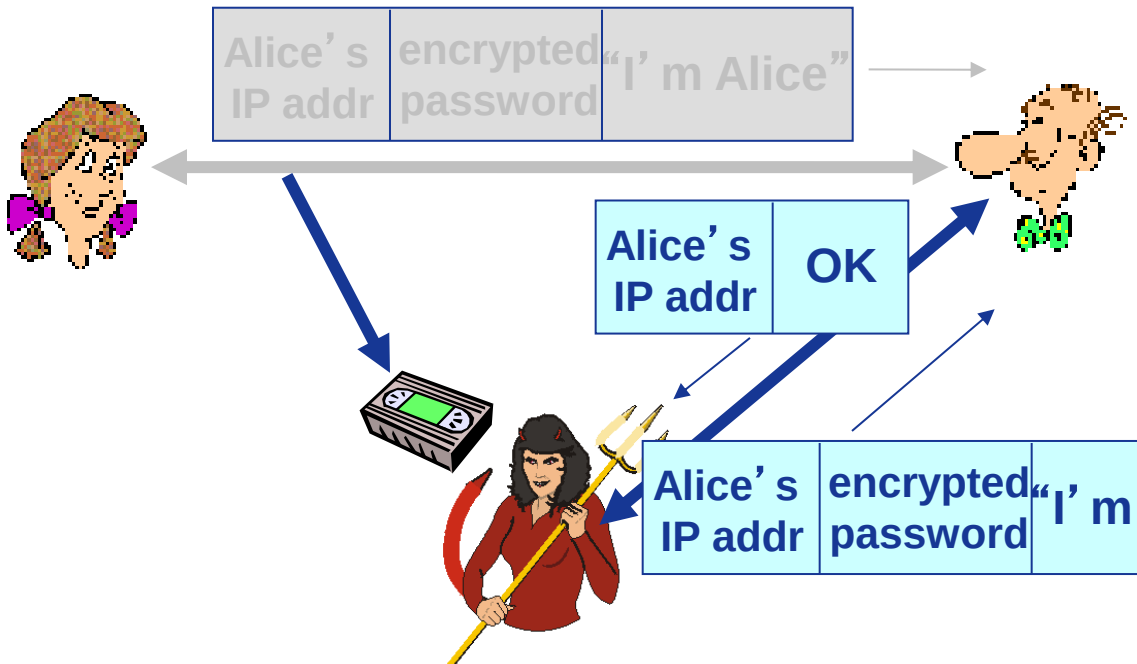
协议ap3.1: Alice声明“I am Alice”的同时，发送她的加密的秘密口令进行“证明”.



失效场景??

身份认证

协议ap3.1: Alice声明“I am Alice”的同时，发送她的加密的秘密口令进行“证明”.



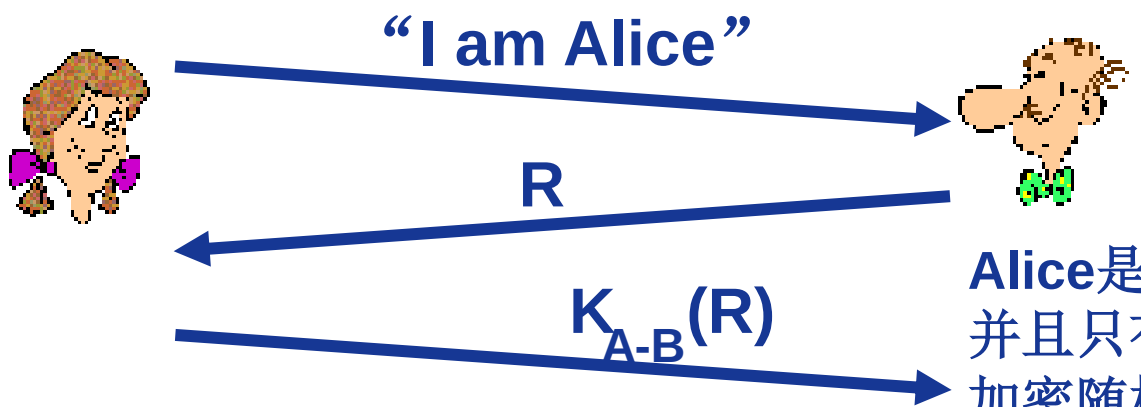
回放攻击(playback attack): Trudy记录Alice 的分组，稍后“回放”给Bob

身份认证

目标: 避免回放攻击

一次性随机数(nonce): 一个生命期内只用一次的数R

ap4.0: 为了证明是“真实的”Alice，Bob向Alice发送一个随机数R，Alice必须返回R，并利用共享密钥进行加密



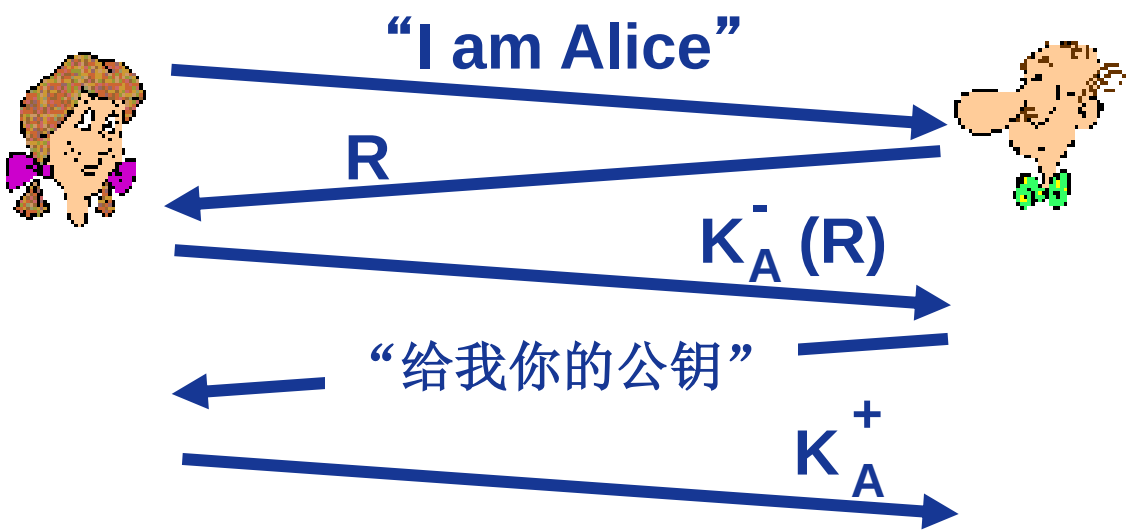
失效、缺点?

身份认证: ap5.0

ap4.0需要共享密钥!

- 是否可以利用公钥技术那?

ap5.0: 利用一次性随机数以及公钥加密技术



Bob计算:

$K_A^+(K_A^-(R)) = R$

并已知只有Alice拥有加密R的私钥,因此:

$K_A^+(K_A^-(R)) = R$

失效?

ap5.0: 安全漏洞

中间人攻击(man in the middle attack): Trudy向Bob假扮Alice，向Alice假扮Bob。

I am Alice

I am Alice

R

$K_T^-(R)$

给我你的公钥

K_T^+

R

$K_A^-(R)$

给我你的公钥

K_A^+

$K_T^+(m)$

Trudy截获明文

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

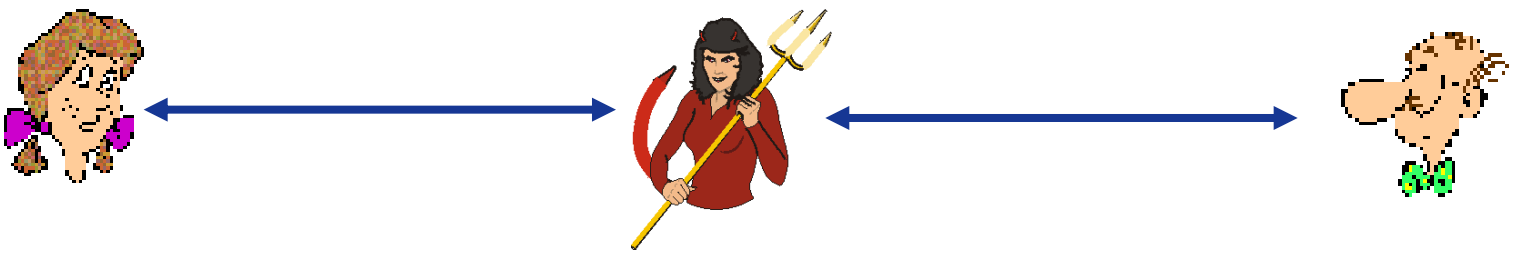
利用Alice的公钥加密明文m并发送给Alice

$K_A^+(m)$

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

ap5.0: 安全漏洞

中间人攻击(man in the middle attack): Trudy向Bob假扮Alice，向Alice假扮Bob。



很难检测:

- **Bob与Alice可以收到彼此发送的所有信息。**
- **问题是Trudy也收到了所有信息！**

本讲主题

报文完整性

报文完整性？

- 报文/消息完整性(message integrity)，也称为报文/消息认证（或报文鉴别），目标：
  - 证明报文确实来自声称的发送方
  - 验证报文在传输过程中没有被篡改
  - 预防报文的时间、顺序被篡改
  - 预防抵赖
    - 发送方否认
    - 接收方否认

密码散列函数

密码散列函数(Cryptographic Hash Function)：H(m)

- 散列算法公开
- H(m)能够快速计算
- 对任意长度报文进行多对一映射，均产生定长输出
- 对于任意报文无法预知其散列值
- 不同报文不能产生相同的散列值
- 单向性：无法根据散列值倒推出报文
  - 对于给定散列值h，无法计算找到满足h = H(m)的报文m
- 抗弱碰撞性(Weak Collision Resistence-WCR)
  - 对于给定报文x，计算上不可能找到y且y≠x，使得H(x)=H(y)
- 抗强碰撞性(Strong Collision Resistence-SCR)
  - 在计算上，不可能找到任意两个不同报文x和y(x≠y)，使得H(x)=H(y)

Internet校验和是优秀的密码散列函数吗？

Internet校验和(checksum)具备散列函数的某些属性：

- ✓ 多对一映射
- ✓ 对于任意报文，产生固定长度的散列值(16-bit校验和)

但是，对于给定的报文及其散列值，很容易找到另一个具有相同散列值的不同报文！

message	ASCII format
I O U 1	49 4F 55 31
0 0 . 9	30 30 2E 39
9 B 0 B	39 42 D2 42
	B2 C1 D2 AC

message	ASCII format
I O U 9	49 4F 55 39
0 0 . 1	30 30 2E 31
9 B 0 B	39 42 D2 42
	B2 C1 D2 AC

不同报文却
得到完全相同的
散列值!

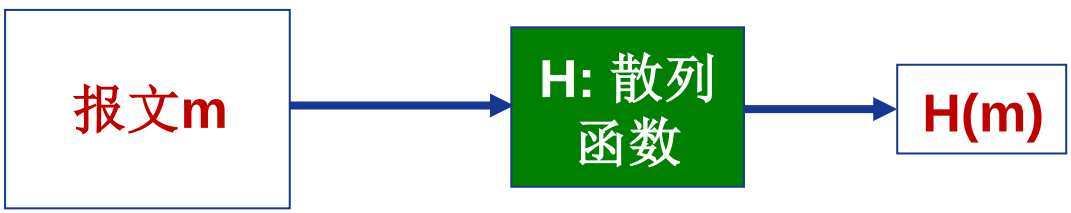
散列函数算法

- MD5：被广泛应用的散列函数(RFC 1321)
  - 通过4个步骤，对任意长度的报文输入，计算输出128位的散列值
  - MD5不是足够安全
    - 1996年，Dobbertin找到了两个不同的512-bit块，在MD5计算下产生了相同的散列值
- SHA-1(Secure Hash Algorithm)：另一个正在使用的散列算法
  - US标准 [NIST, FIPS PUB 180-1]
  - SHA-1要求输入消息长度<2^{64}
  - SHA-1的散列值为160位
  - 速度慢于MD5，安全性优于MD5

报文摘要(Message digests)

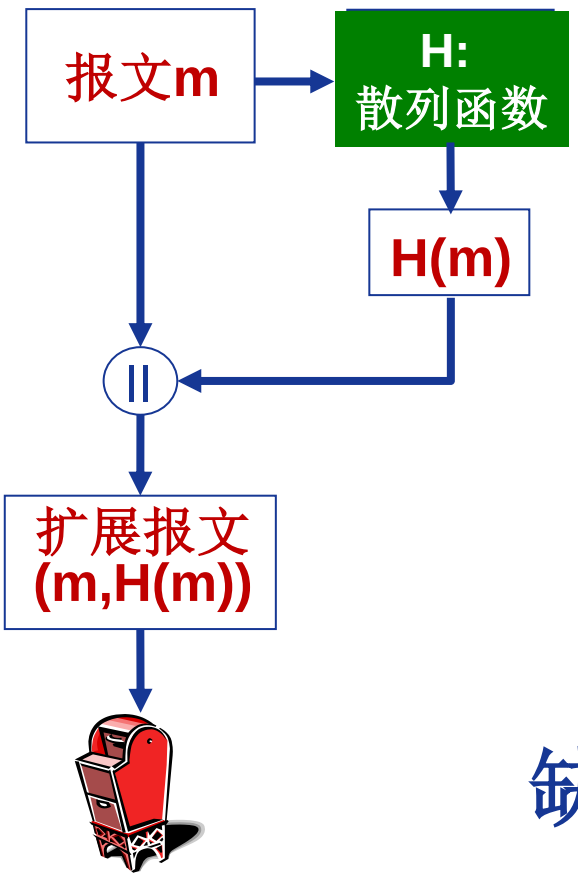
对报文m应用散列函数H，得到一个固定长度的散列码，称为报文摘要(message digest)，记为H(m)

✓ 可以作为报文m的数字指纹(fingerprint)。

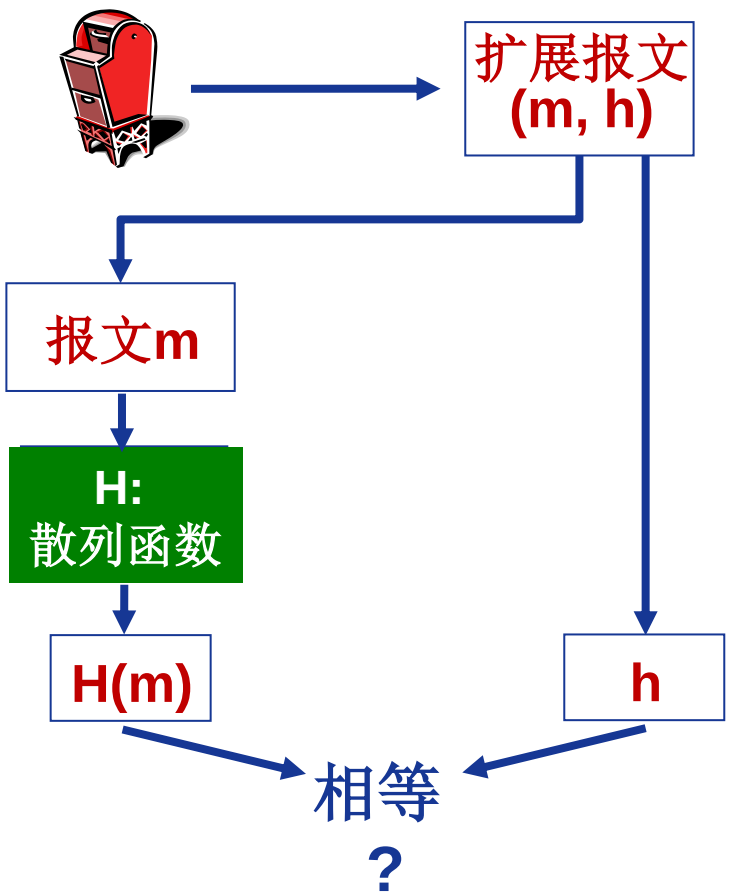


报文认证

简单方案：报文+报文摘要→扩展报文(m, H(m))



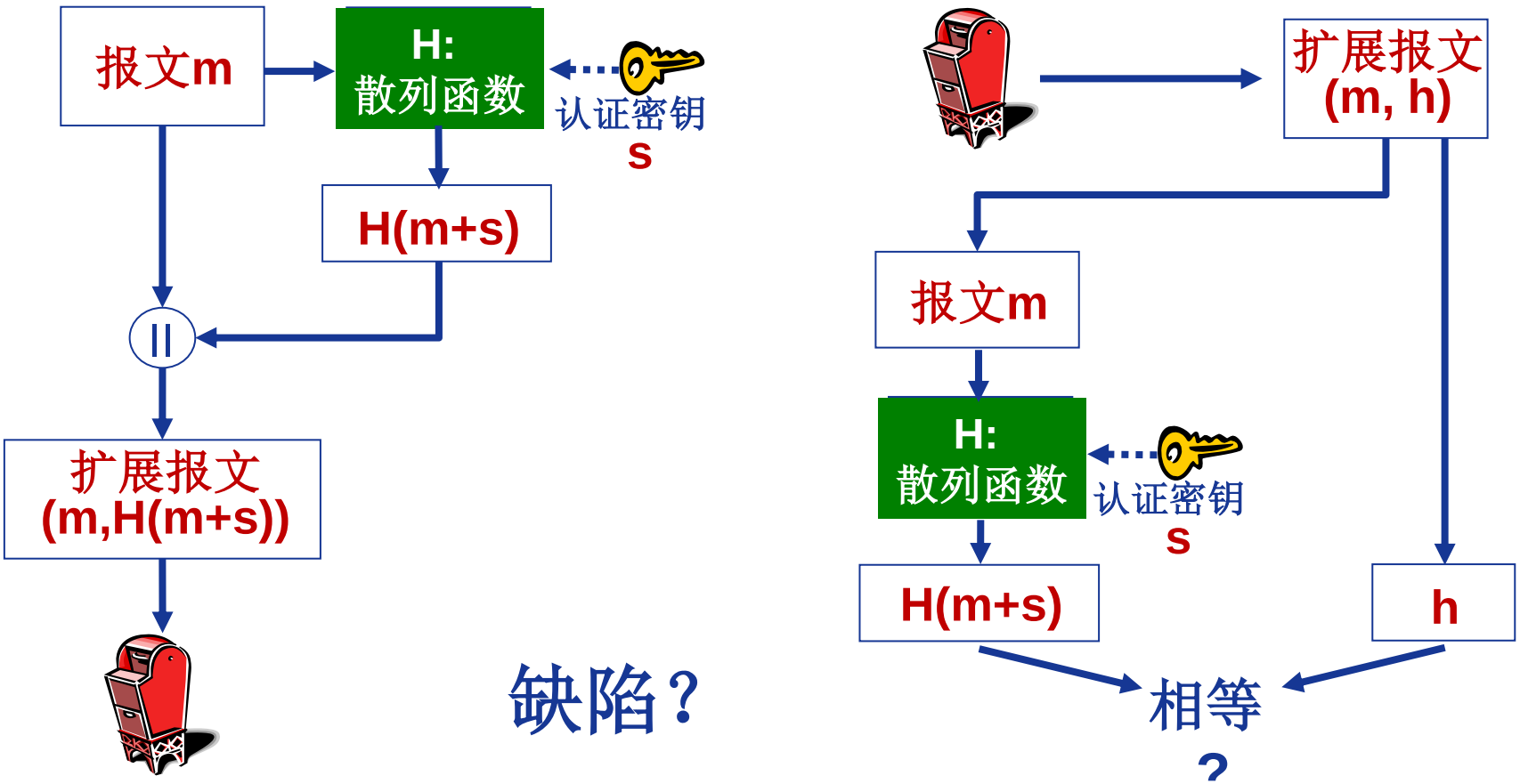
缺陷？



报文认证

报文认证码MAC(Message Authentication Code)：

报文m+认证密钥s+密码散列函数H→扩展报文(m, H(m+s))



缺陷？

本讲主题

数字签名

数字签名

Q:如何解决下列与报文完整性相关的问题？

- 否认：发送方不承认自己发送过某一报文
- 伪造：接收方自己伪造一份报文，并声称来自发送方
- 冒充：某个用户冒充另一个用户接收或发送报文
- 篡改：接收方对收到的信息进行篡改

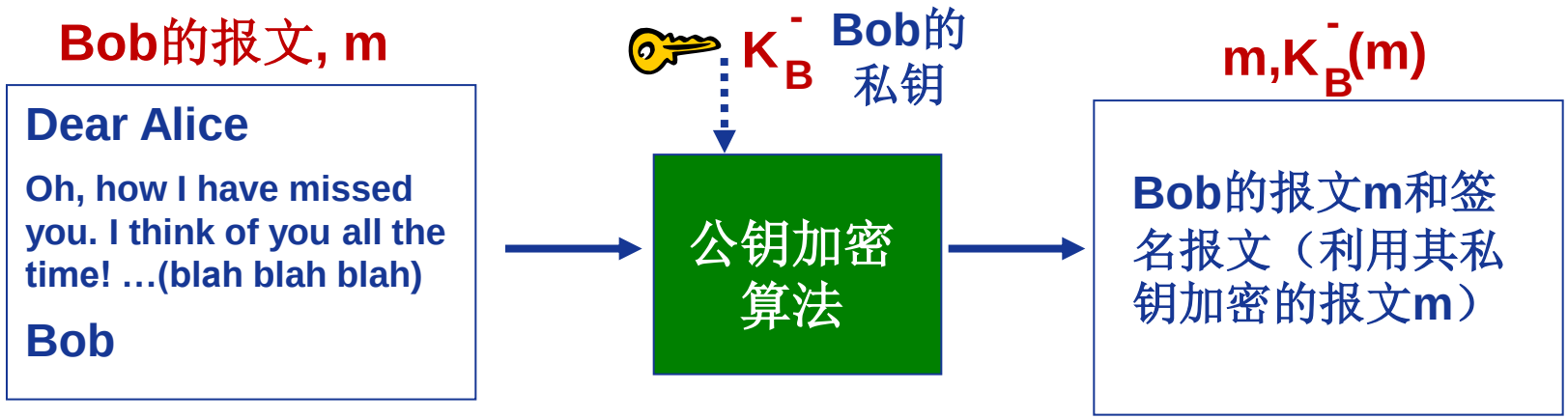
A:数字签名(Digital signatures)！

- 数字签名技术是实现安全电子交易的核心技术之一
- 可验证性(verifiable)
- 不可伪造性(unforgeable)
- 不可抵赖性(non-repudiation)

数字签名

对报文m的简单数字签名:

- 报文加密技术是数字签名的基础
- Bob通过利用其私钥 K_B^- 对m进行加密，创建签名报文，$K_B^-(m)$



数字签名

- 假设Alice收到报文m以及签名$K_B^-(m)$
- Alice利用Bob的公钥K_B^+ 解密$K_B^-(m)$，并检验$K_B^+(K_B^-(m)) = m$ 来证实报文m是Bob签名的。
- 如果$K_B^+(K_B^-(m)) = m$ 成立，则签名m的一定是Bob的私钥
- 于是：

Alice可以证实:

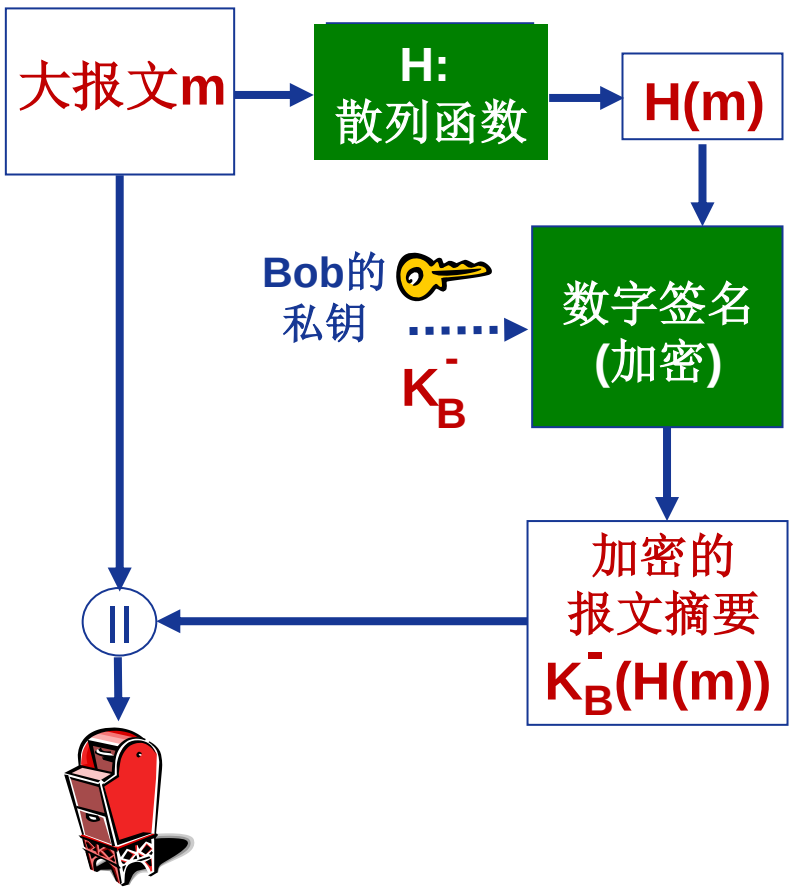
- ✓ Bob签名了m
- ✓ 没有其他人签名m的可能
- ✓ Bob签名的是m而不是其他报文m'

不可抵赖(non-repudiation):

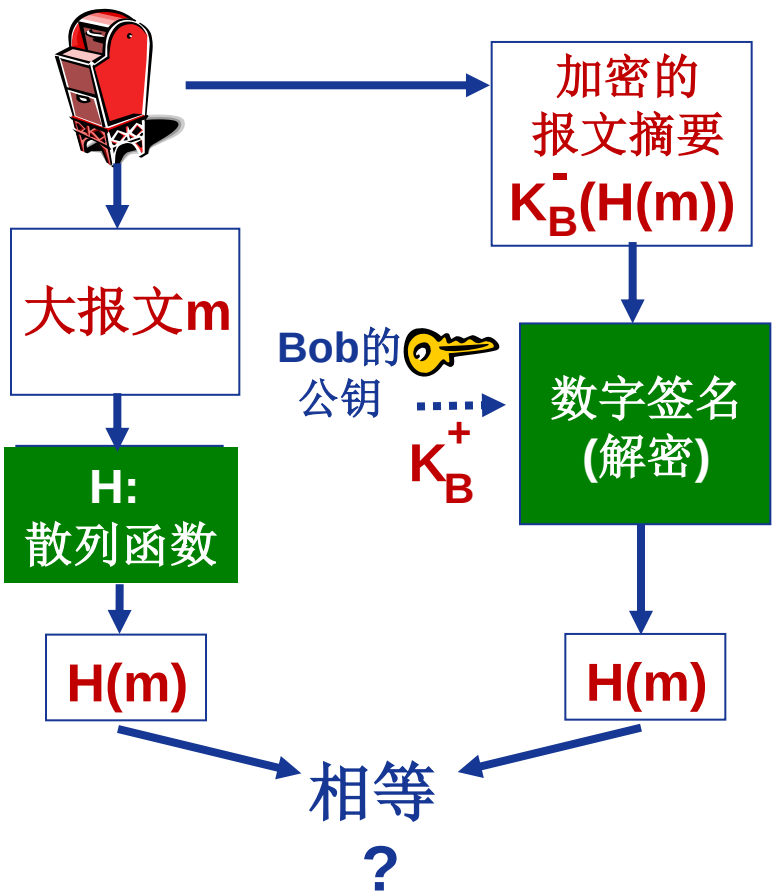
- ✓ Alice可以持有m和签名$K_B^-(m)$，必要时可以提交给法院证明是Bob签名的m

签名报文摘要

Bob发送数字签名的报文:



Alice核实签名以及数字签名报文的完整性:



本讲主题

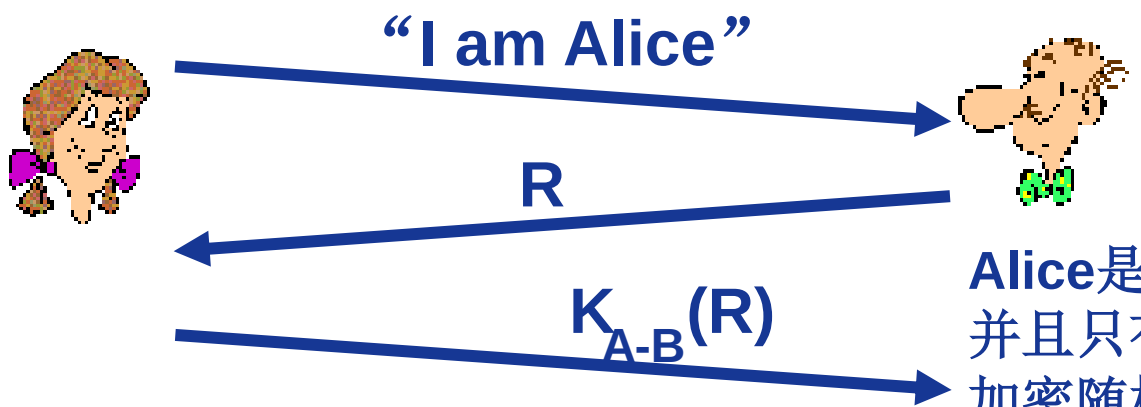
密钥分发中心(KDC)

回顾身份认证协议：ap4.0

目标：避免回放攻击

一次性随机数(nonce)：一个生命期内只用一次的数R

ap4.0：为了证明是“真实的”Alice，Bob向Alice发送一个随机数R，Alice必须返回R，并利用共享密钥进行加密



如何共享对称密钥？

对称密钥问题？

对称密钥问题:

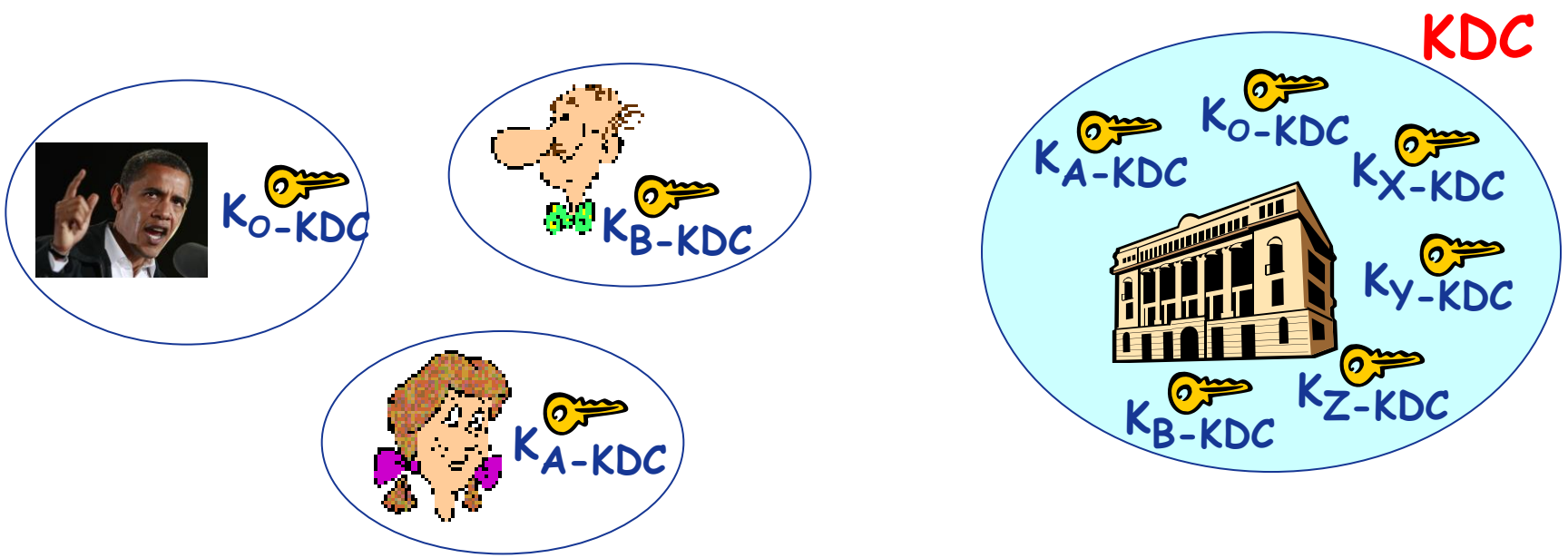
- 两个实体在网上如何建立共享秘密密钥？

解决方案:

- 可信任的密钥分发中心(Key Distribution Center-KDC)作为实体间的中介(intermediary)

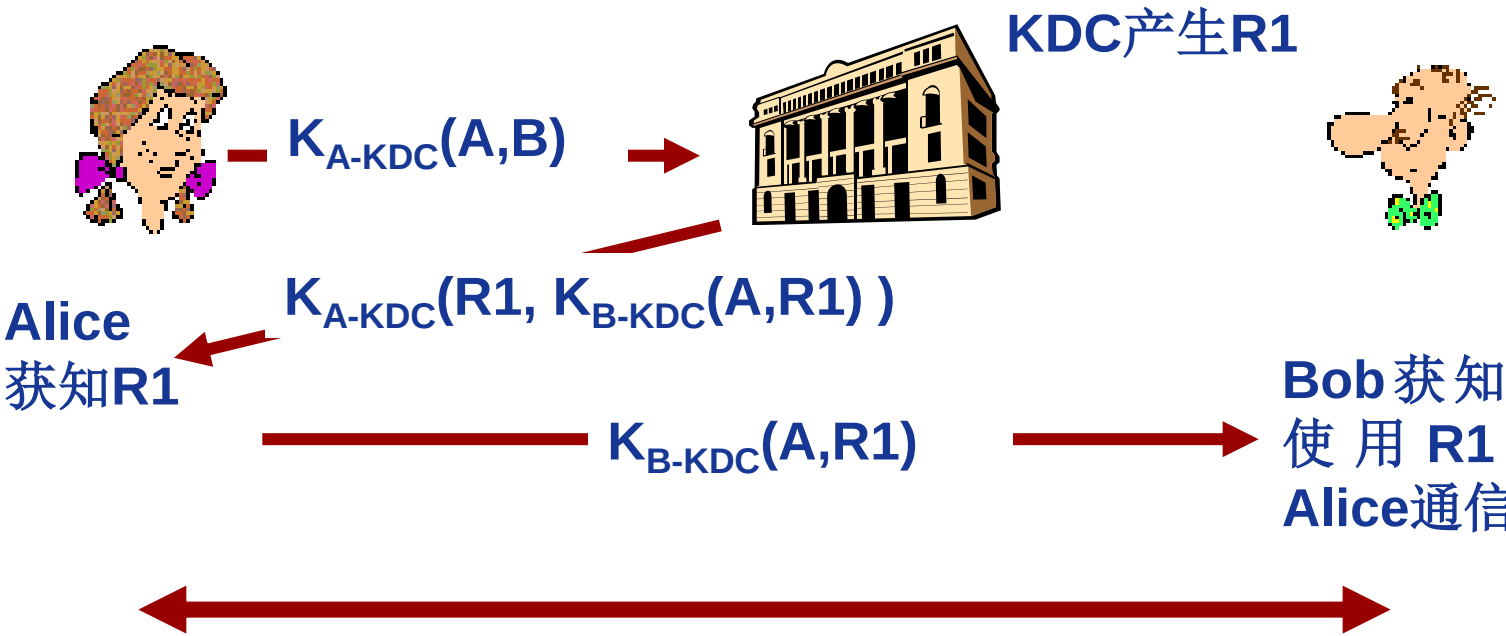
密钥分发中心(KDC)

- Alice与Bob需要共享对称密钥.
- KDC：一个服务器
  - 每个注册用户(很多用户)共享其与KDC的秘密密钥
- Alice和Bob只知道自己与KDC之间的对称密钥，用于分别与KDC进行秘密通信.



密钥分发中心(KDC)

Q: KDC如何支持Bob和Alice确定用于彼此通信的共享对称密钥呢？

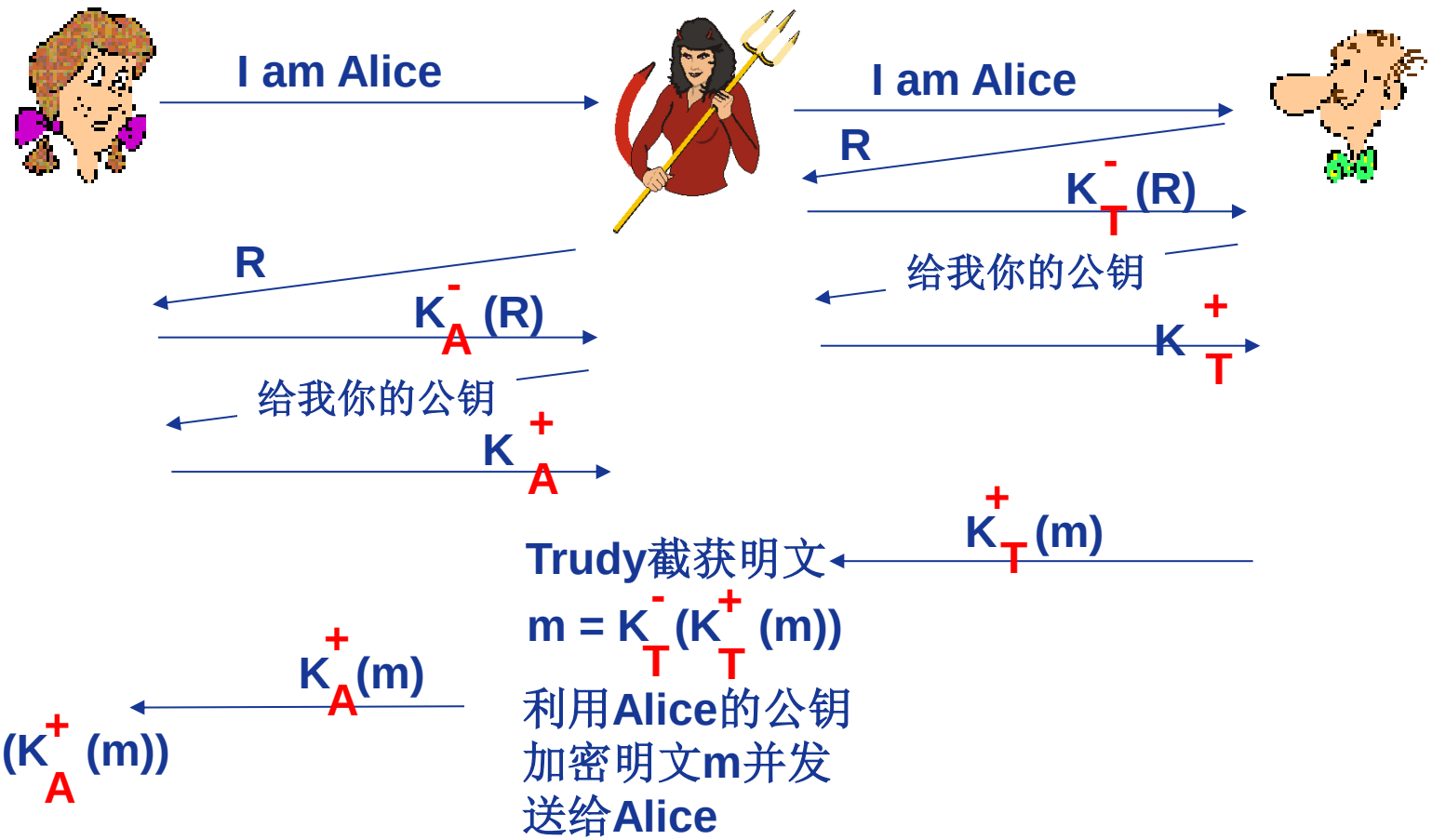


本讲主题

认证中心(CA)

回顾身份认证协议：ap5.0

中间人攻击(man in the middle attack): Trudy向Bob假扮Alice，向Alice假扮Bob。



比萨恶作剧

- Trudy针对Bob实施“比萨恶作剧”
  - Trudy创建邮件订单:
    Dear Pizza Store, Please deliver to me four pepperoni pizzas. Thank you, Bob
  - Trudy利用她的私钥签名订单
  - Trudy向比萨店发送订单
  - Trudy向比萨店发送她的公钥，但她声称这是Bob的公钥
  - 比萨店核实签名；然后向Bob递送4个腊肠比萨
  - Bob根本就不喜欢腊肠

公钥问题？

公钥问题:

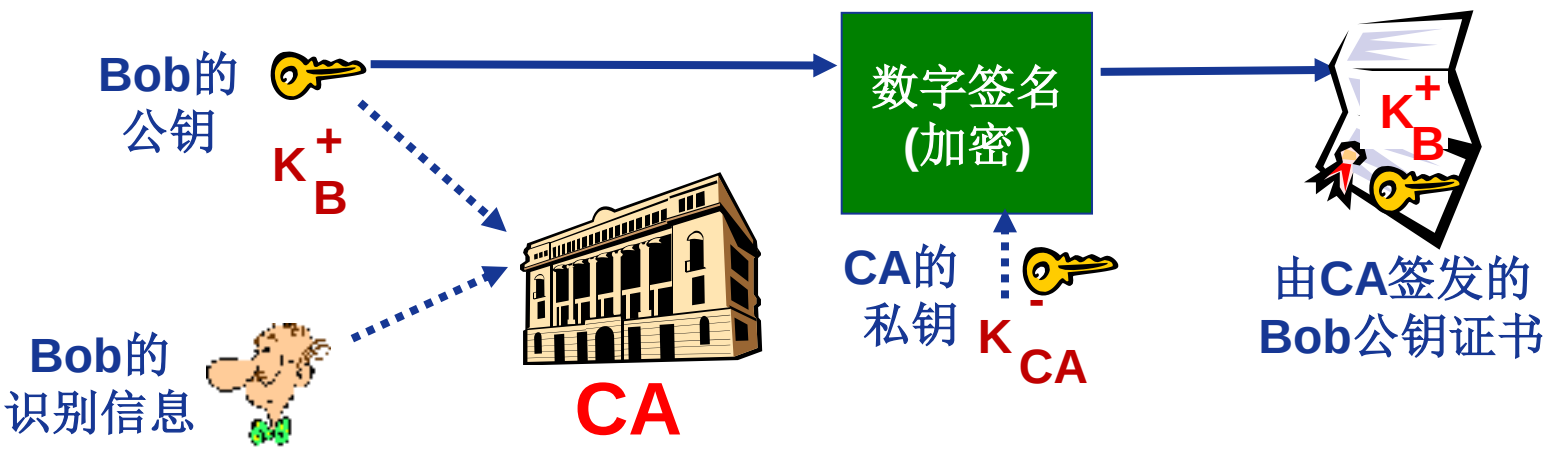
- 当Alice获得了Bob的公钥 (通过web网站、e-mail、磁盘等)，她怎么确认这真的是Bob的公钥而不是Trudy的?

解决方案:

- 可信任的认证中心(Certification Authority-CA)

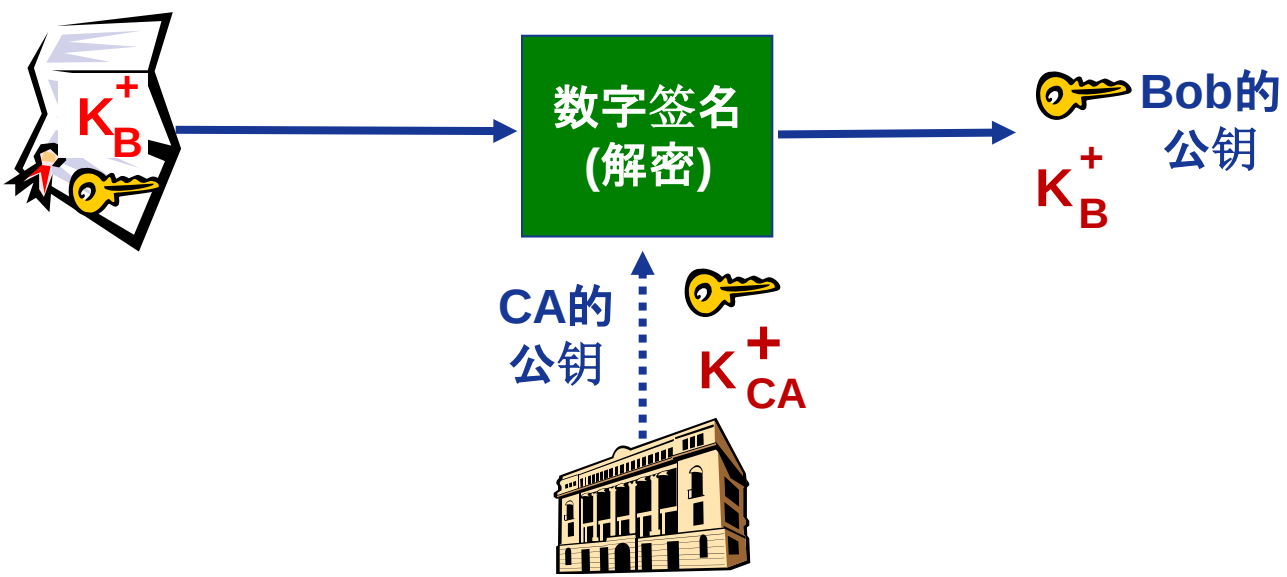
认证中心

- 认证中心(CA): 实现特定实体E与其公钥的绑定
- 每个E(如人、路由器等)在CA上注册其公钥.
  - E向CA提供“身份证明”.
  - CA创建绑定E及其公钥的证书(certificate).
  - 证书包含由CA签名的E的公钥 – CA声明：“这是E的公钥”



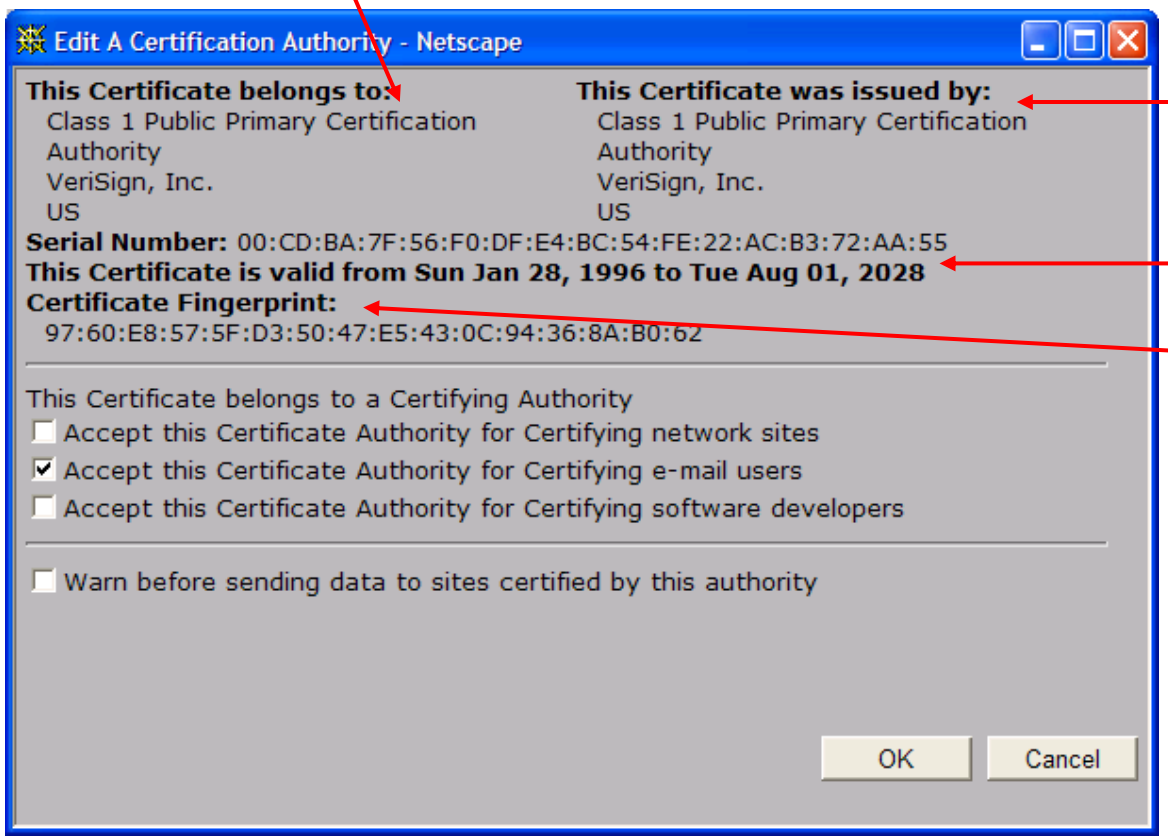
认证中心

- 当Alice想要Bob的公钥时:
  - 首先或取Bob的公钥证书(从Bob或者其他地方).
  - 应用CA的公钥，解密证书中签名的公钥，获得Bob公钥



公钥证书主要内容

- 序列号(唯一发行号)
- 证书持有者信息，包括算法和密钥值(未显示)



- 证书发行者信息
- 有效期
- 发行者数字签名