§6.2 密钥分配协议

- 密钥分配是密码系统中密钥的传送和分发中的问题。
- 密钥分配的基本方法：
  - 利用安全信道实现。如传统的方法或隐蔽方法。
    传统的方法是通过邮递或信使护送密钥，隐蔽的方法如将密钥分拆成几部分分别递送。
  - 利用数学上求逆的困难性建立的安全信道来实现。如双钥密码体制采用的密钥分配协议。

一、公开密钥分发

二、秘密密钥分发

三、**Differ-Hellman**密钥分发

§6.2 密钥分配协议

一、公开密钥分发

- 公开密钥分发，就是公钥系统中的加密公开密钥是公开的，它通过各种公开的手段和方式，或由公开权威机构实现公开密钥分发和传送。
- 公开密钥分发不需要机密性然而完整性是必需的。
- 公开密钥分发方式：广播式公开密钥分发、公开可用的目录、公开密钥机构分发、公开密钥证书分发。

（1）广播式公开密钥分发

一种自由公布方式，通过**BBS、USENET**或**Email**等。

说明：易受假冒用户攻击。

§6.2 密钥分配协议

（2）公开可用的目录

需要可信任的中央授权机构。

- 授权机构维护着动态｛用户名，公开密钥｝目录。
- 用户在授权机构注册其公钥。注册可以人工或以某种安全认证的通信方式。
- 用户可以替换其公钥。
- 授权机构定期发布或更新整个目录。
- 用户可在网络上直接访问公共目录。这时需要在用户和中央授权机构之间强制使用安全、认证的通信方式。

说明：若成功修改了公共目录，则攻击者可假冒用户。

§6.2 密钥分配协议

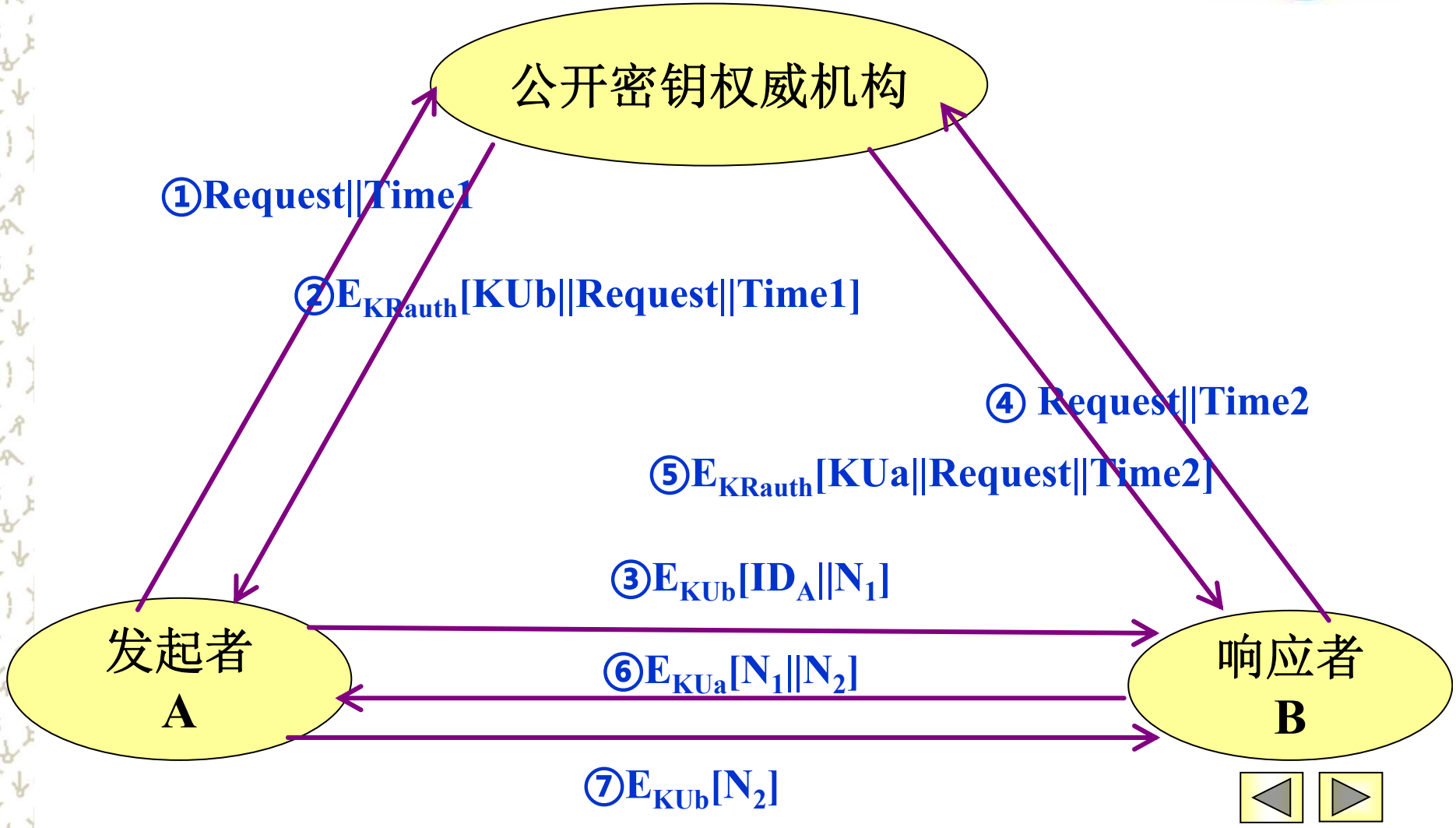
（3）公开密钥机构分发

需要可信任的中央授权机构，每个用户知道授权机构的公钥。

说明：

- 若成功修改了公共目录，则攻击者可假冒用户。
- 授权中心易成为性能及安全瓶颈。

§6.2 密钥分配协议

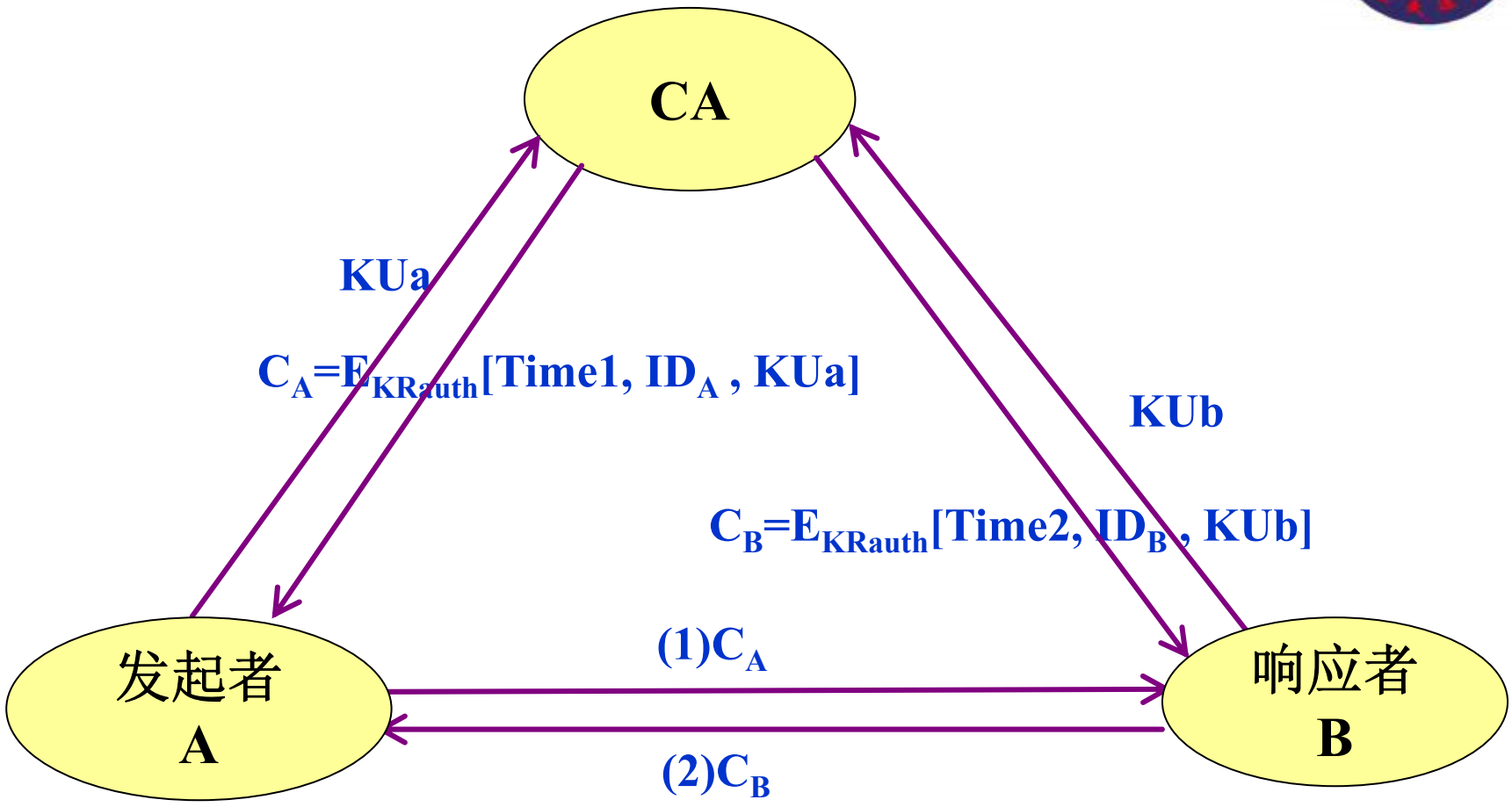

公开密钥权威机构
①Request||Time1
②E_{KRauth}[KUb||Request||Time1]
④ Request||Time2
⑤E_{KRauth}[KUa||Request||Time2]
③E_{KUb}[ID_A||N_1]
发起者
A
⑥E_{KUa}[N_1||N_2]
⑦E_{KUb}[N_2]
响应者
B

§6.2 密钥分配协议

（4）公开密钥证书分发

- 任何参与者可以通过读取证书来确定被证实的主人的名字和公钥；
- 任何参与者可以验证证书是来源于CA且非伪造的。
- 只有该CA才能生成和更新证书；
- 任何参与者可以验证证书的有效流通时间
- 说明：泄漏私钥等价于丢失证书。
  证书的时间作为有效期。

§6.2 密钥分配协议

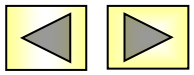


验证：$D_{KUauth}[C_A]= D_{KUauth}[E_{KRauth}[T, ID_A, KUa]]=(T, ID_A, KUa)$

§6.2 密钥分配协议

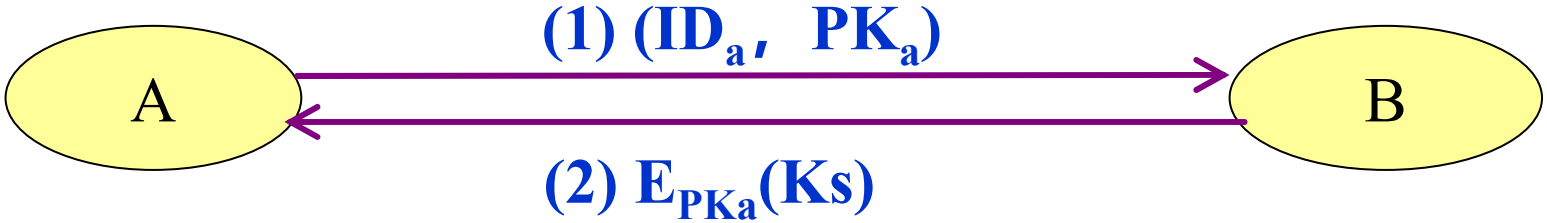
二、秘密密钥分发

- 基于公开密钥体制的秘密密钥分配
- 方案
  - 简单的秘密密钥分配Simple secret key distribution
  - 具有保密和鉴别能力的秘密密钥分配Secret key distribution with confidentiality and authentication
  - 混合方案Hybrid scheme



§6.2 密钥分配协议

简单的秘密密钥分配：



(1) A生成{PK_a, SK_a}，A → B：(ID_a，PK_a)

(2) B生成随机密钥Ks，B → A：$E_{PKa}(Ks)$

(3) A解密$E_{PKa}(Ks)$得到Ks：$D_{SKa}(E_{PKa}(Ks))$

(4) A丢弃{PK_a, SK_a}，B丢弃PK_a

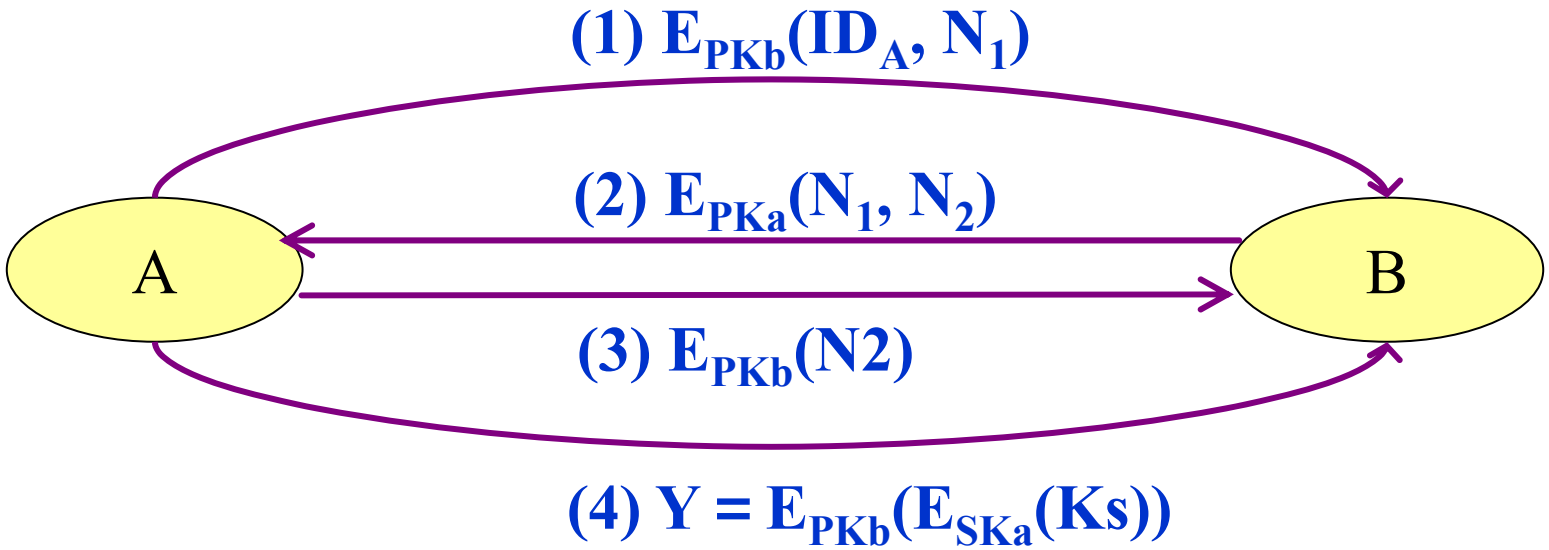
说明：通讯前不需存在密钥，通讯后也不存在密钥

能抵抗偷听，不能抵抗主动攻击(中间人攻击)

§6.2 密钥分配协议

具有保密和鉴别能力的分配：

假定A和B已经获得了双方的公钥



B解密Y获得会话密钥$Ks=D_{PKa}(D_{SKb}(Y))$

§6.2 密钥分配协议



三、Diffie-Hellman密钥交换：

- 典型的密钥协商协议，于1976年提出，用于网络环境下用户双方安全的交换密钥。
- 其安全性主要基于有限域上离散对数问题的难解性
- 协议描述：

设p是大素数，g是Z_p的生成元，p和g公开。

1) 用户A选择大随机数r_A，$0< r_A< p-1$

2) 用户A计算$S_A = g^{r_A} \bmod p$，将结果传送给用户B

3) 用户B选择大随机数r_B，$0< r_B< p-1$

4) 用户B计算$S_B = g^{r_B} \bmod p$，将结果传送给用户A

5) 用户A计算$K = S_B^{r_A} \bmod p$，用户B计算$K = S_A^{r_B} \bmod p$

§6.2 密钥分配协议

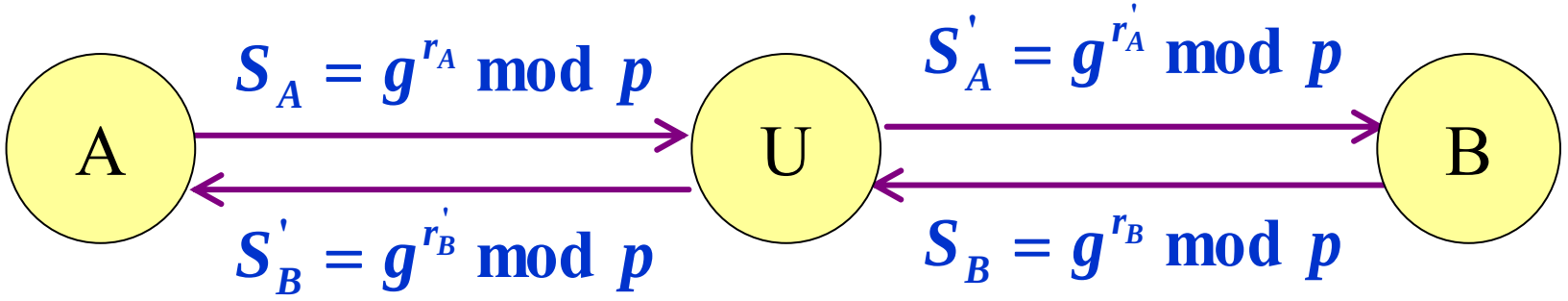
因为 $K = S_B^{\ r_A} \bmod p = (g^{r_B} \bmod p)^{r_A} \bmod p$

$= (g^{r_B r_A}) \bmod p = (g^{r_A} \bmod p)^{r_B} \bmod p$

$= S_A^{\ r_B} \bmod p$

于是A和B共享会话密钥K，使用对称密码体制进行保密通信。

- 安全缺陷：易受中间人攻击。



于是A和U共享密钥 $K_{AU} = (g^{r_B' r_A}) \bmod p$，

B和U共享密钥 $K_{BU} = (g^{r_A' r_B}) \bmod p$ 。

§6.2 密钥分配协议

- 改进方案：可信第三方TA，每个用户A持有TA签发的证书

  $C(A)=(ID(A)，Ver_A，Sig_A(ID(A)，Ver_A))$

- 协议描述：

  设p是大素数，g是Z_p的生成元，p和g公开。

  1) 用户A选择大随机数r_A，$0<r_A<p-1$

  2) 用户A计算$S_A = g^{r_A} \bmod p$，将S_A传送给用户B

  3) 用户B选择大随机数r_B，$0<r_B<p-1$

  4) 用户B计算$S_B = g^{r_B} \bmod p$，然后计算

  $$K = S_A^{\,r_B} \bmod p \ ， \quad y_B = Sig_B(S_A, S_B)$$

  5) 用户B将$(C(B)，S_B，y_B)$传送给用户A

§6.2 密钥分配协议

6) 用户A计算 $K = S_B^{r_A} \bmod p$ ， 并使用Ver_B验证y_B，
使用Ver_{TA}验证C(B)

7) 用户A计算 $y_A = Sig_A(S_A, S_B)$，将(C(A)，y_A)传送给B

8) 用户B使用Ver_A验证y_A，使用Ver_{TA}验证C(A)

- 改进方案可有效抵制中间人攻击。因为攻击者以 $S_A^{'} = g^{r_A^{'}} \bmod p$ 替换S_A后无法冒充B进行签名 $Sig_B(S_A^{'}, S_B)$