东南大学网络空间安全学院
密码学与安全协议

第七讲 密钥管理技术

黄　杰
信息安全研究中心

知识点：

1、密钥管理的重要性

2、基于对称密码算法的密钥分配方法

3、基于公钥密码算法的密钥分配方法

4、DH算法的工作原理

密钥管理技术概述

- 密钥管理的重要性
  - ✓ 所有的密码技术都依赖于密钥。

现代密码体制要求加密算法是可以公开评估的，整个密码系统的安全性并不取决于对密码算法的保密或者是对加密设备等的保护。

决定整个密码体制安全性的因素将是密钥的保密性（“一切秘密予于密钥之中！”）：

密码算法可以公开，密码设备可以丢失，但它们都不危及密码体制的安全性；但一旦密钥丢失，非法用户将会有可能窃取信息。

✓ 在考虑密码系统的应用设计时，特别是在商用系统的设计时，需要解决的核心问题之一是密钥管理问题，而不是密码算法问题。

例如：商用系统可以使用公开的、经过大量评估分析认为抗攻击能力比较强的算法。

✓ 密钥的管理本身是一个很复杂的课题，而且是保证安全性的关键点之一。

- **密钥管理的概念**

密钥管理是一门综合性的技术，涉及密钥的产生、检验、分发、传递、保管、使用、销毁的全部过程，还与密钥的行政管理制度以及人员的素质密切相关。

- **密钥管理的目的**

维持系统中各实体之间的密钥关系，以抗击各种可能的威胁：

- 密钥的泄露
- 秘密密钥或公开密钥的身份的真实性丧失
- 未经授权使用

- **密钥管理系统的要求**

- 密钥难以被非法窃取；
- 在一定条件下获取了以前的密钥用处也很小；
- 密钥的分配和更换过程对用户是透明的。

密钥管理的原则

- 全程安全原则

必须在密钥的产生、检验、分发、传递、保管、使用、销毁全过程中对密钥采取妥善的安全管理

- 最小权利原则

应当只分发给用户进行某一事务处理所需的最小的密钥集合

- 责任分离原则

一个密钥应当专职一种功能，不要让一个密钥兼任几种功能

- 密钥分级原则

可减少受保护的密钥数量，又可简化密钥的管理工作。一般可将密钥划分为三级：主密钥，二级密钥，初级密钥

- 密钥更新原则

密钥必须按时更新。否则，即使是采用很强的密码算法，使用时间越长，敌手截获的密文越多，破译密码的可能性就越大

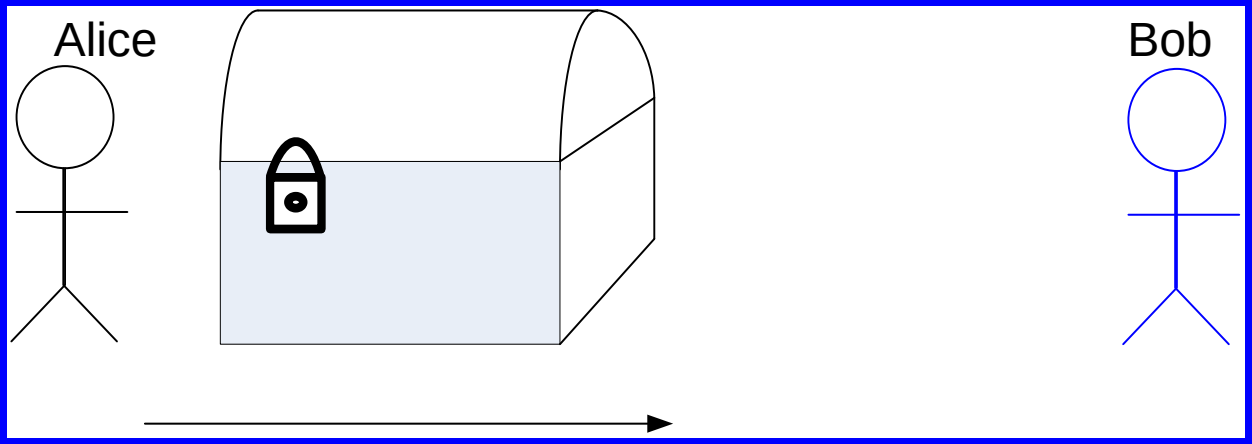
- 密钥应当有足够的长度

密码安全的一个必要条件是密钥有足够的长度。密钥越长，密钥空间就越大，攻击就越困难，因而也就越安全

- 密码体制不同，密钥管理也不相同

由于对称密码体制与公开密钥密码体制是性质不同的两种密码，因此它们在密钥管理方而有很大的不同。

引言1 双重加密方案



- **双重加密方案**
  1. Alice把消息放到箱子里，用自己的锁锁上发送给Bob
  2. Bob收到箱子后加上自己的锁，再把箱子返回给Alice
  3. Alice打开自己的锁，再把箱子寄给Bob
  4. Bob除掉自己的锁，读取消息

- **双重加密方案的数学描述：**

  1. Alice发送$E_A(P)$给Bob
  2. Bob发送$E_B(E_A(P))$给Alice
  3. Alice发送$D_A(E_B(E_A(P)))=D_A(E_A(E_B(P)))=E_B(P)$给Bob
  4. Bob解密$D_B(E_B(P)=P$

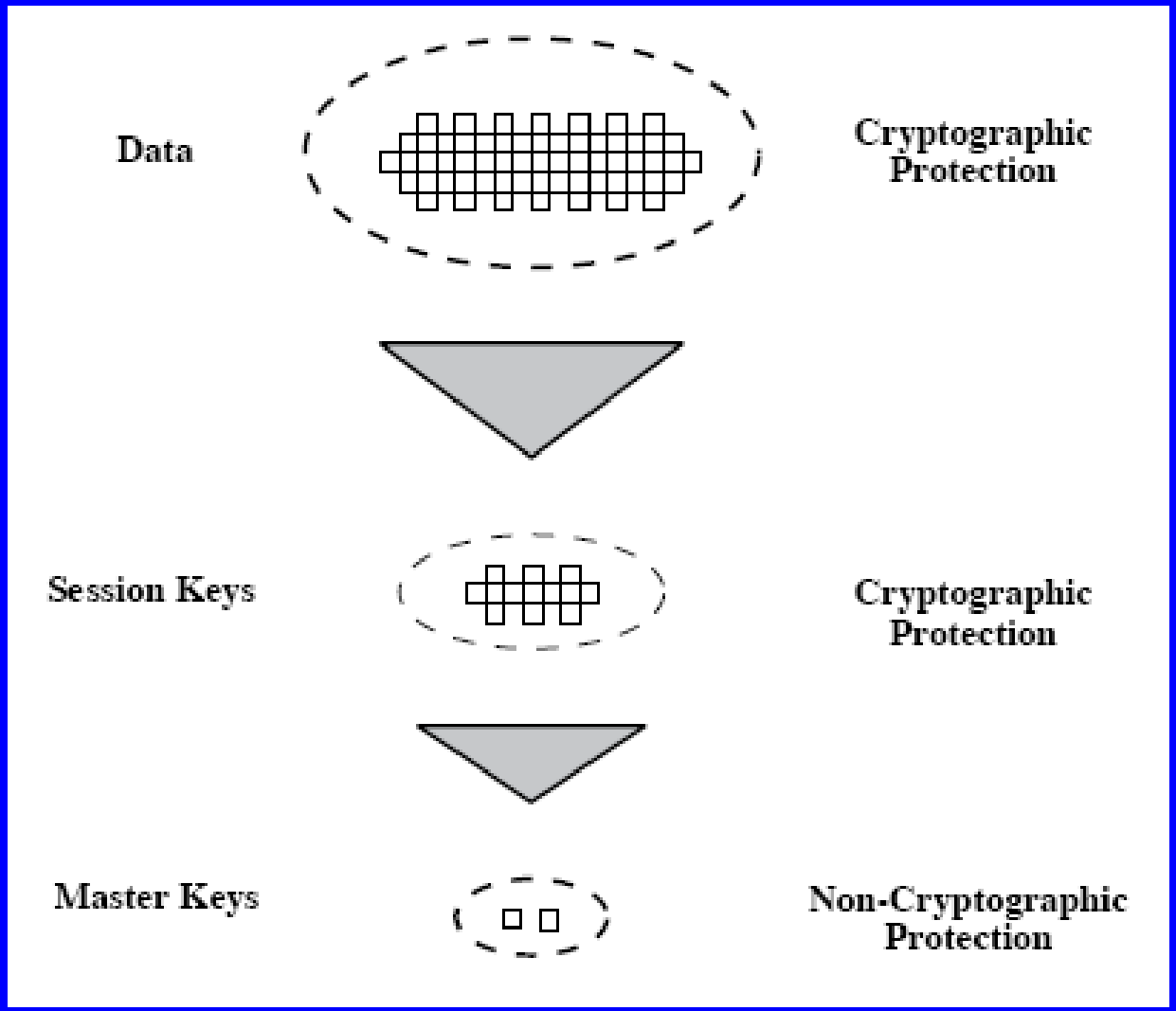
- **双重加密方案的分析**
  - 要求构造一个函数，满足：
    - $E_B(E_A(P))=E_A(E_B(P))$

对称密钥管理技术

密钥分配

- **需求：**
  - 通信双方共享密钥，且不为他人所知
  - 为保证通信安全，密钥必须经常变动
- **密钥分配方法：**
  - 密钥由A选择，并亲自交给B　　人工
  - 第三方选择密钥后亲自交给A和B　　人工
  - A或B使用最近使用过的密钥加密新密钥再发给另一方
  - A和B与第三方C均有秘密通道，则C可将密钥分别秘密发给A和B
- **N个用户需要N(N-1)/2个密钥，如果有10,000个用户，则需要近5000万个密钥！**

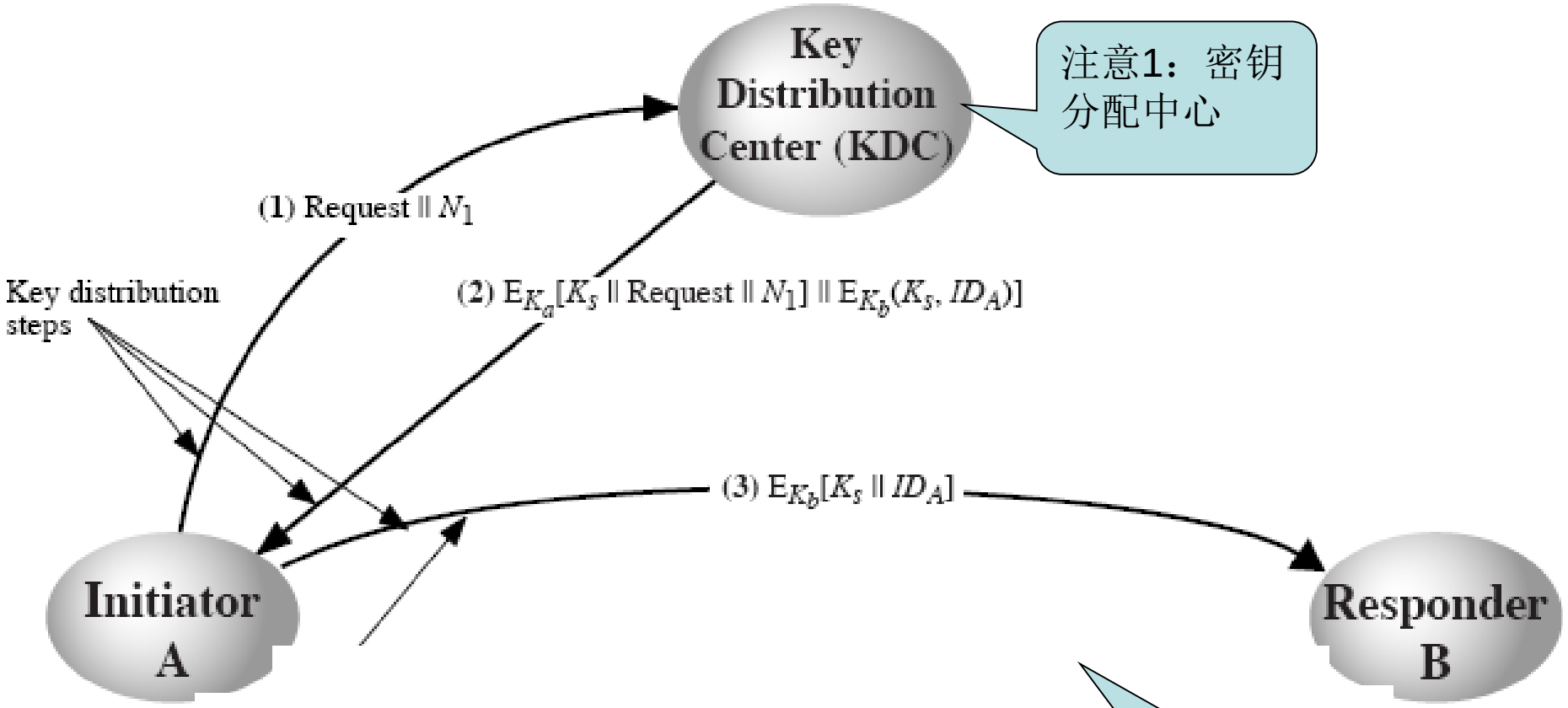
层次式密钥


Data
Cryptographic Protection
Session Keys
Cryptographic Protection
Master Keys
Non-Cryptographic Protection

典型的对称密钥分配模式

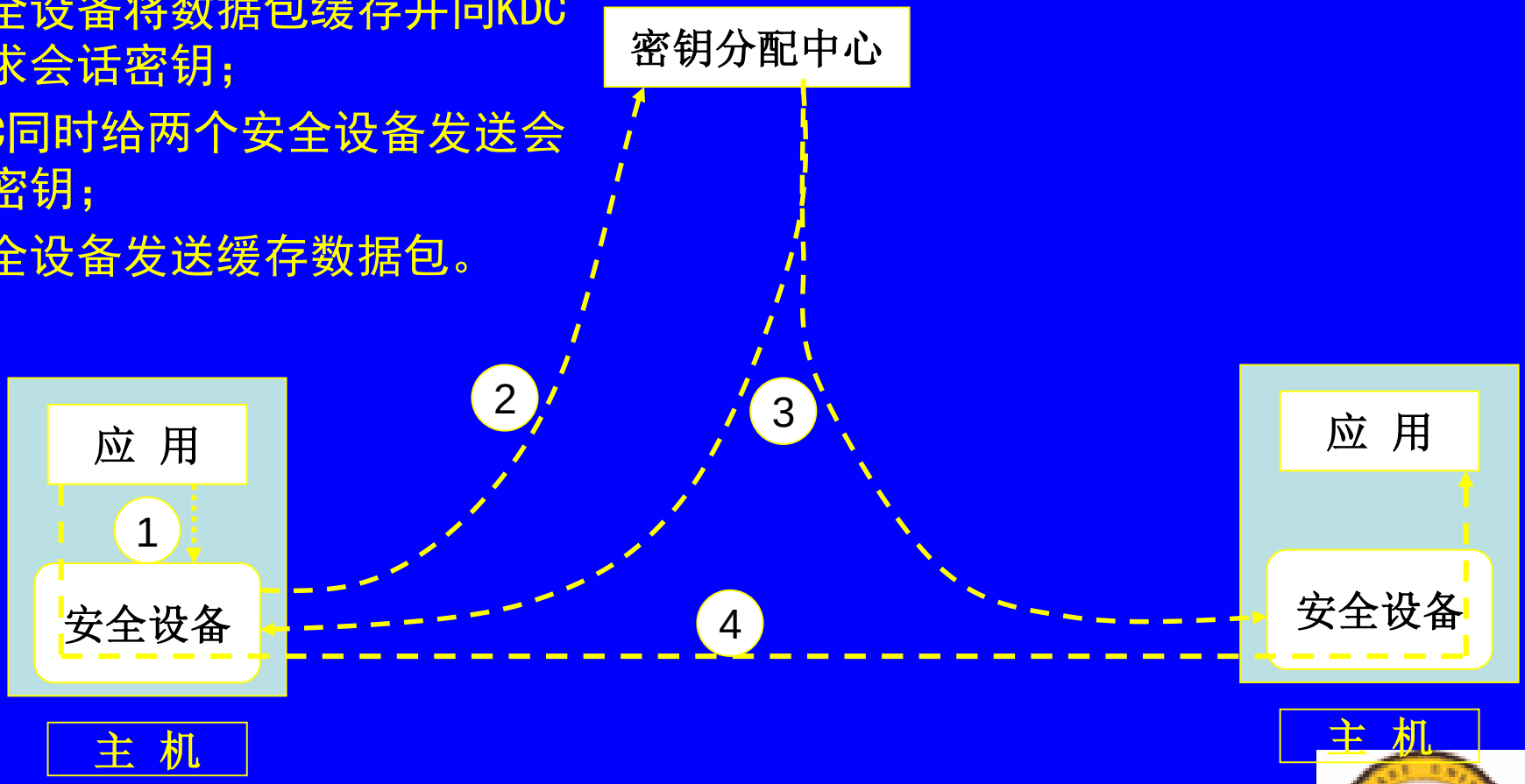
- 密钥分配中心(KDC)：负责分发密钥给需要的用户。
- A与B要建立逻辑连接，需要用一个一次性的会话密钥来保护数据的传输
- 假定：
  - A拥有一个主密钥Ka，与KDC共享
  - B拥有一个主密钥Kb，与KDC共享

典型的密钥分配过程


Key Distribution Center (KDC)
注意1：密钥分配中心
(1) Request || N_1
Key distribution steps
(2) $E_{K_a}[K_s \| \text{Request} \| N_1] \| E_{K_b}(K_s, ID_A)]$
(3) $E_{K_b}[K_s \| ID_A]$
Initiator A
Responder B
注意2：如何确定会话密钥的生命期。面向连接和面向非连接协议各不同。

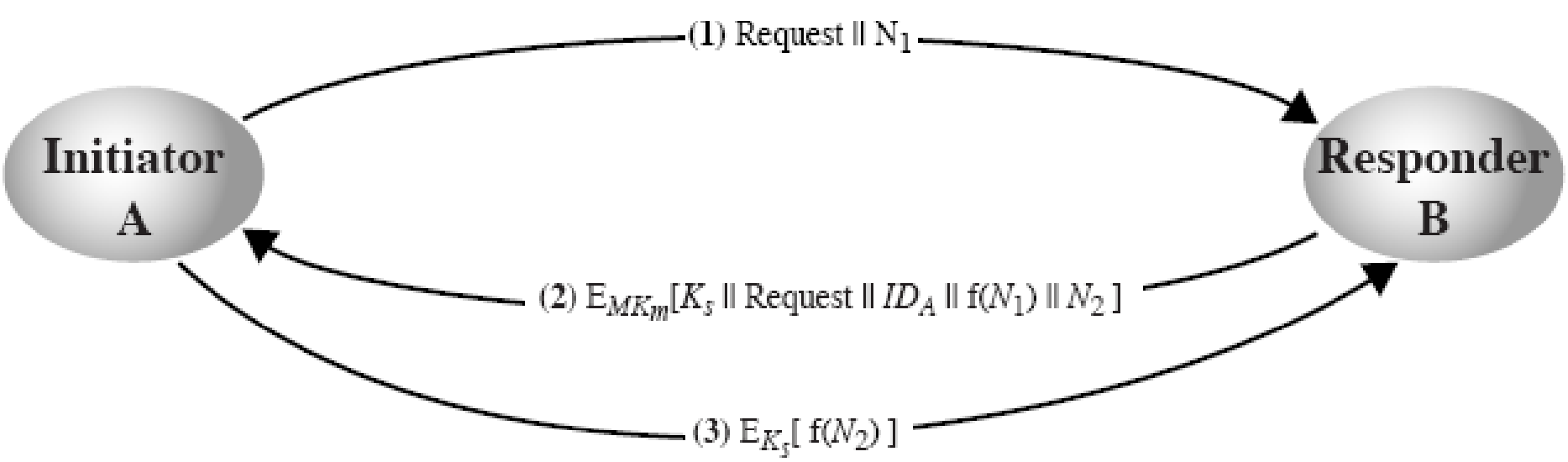
透明的密钥控制方案

- 主机发送请求连接的数据包；
- 安全设备将数据包缓存并向KDC请求会话密钥；
- KDC同时给两个安全设备发送会话密钥；
- 安全设备发送缓存数据包。



分散式密钥控制

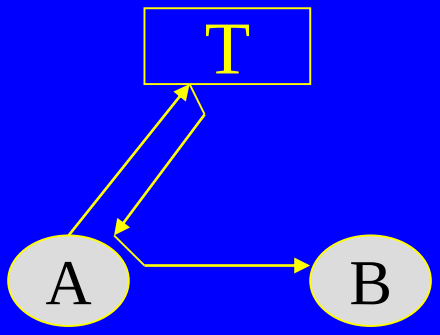
- **KDC：有中心的密钥分配**
  - 主密钥个数减少
  - 必须保证KDC是可信的，且不会受到破坏
- **分散式密钥控制：**
  - 不需要KDC
  - 每个节点需要保存(n-1)个主密钥



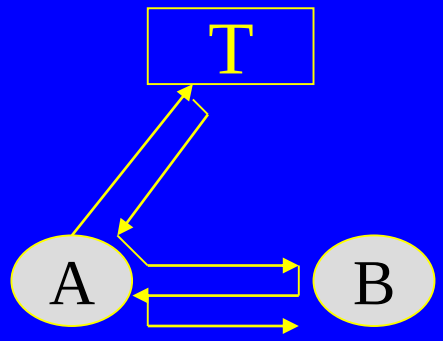
密钥的使用方法

- **定义若干种类型的会话密钥**
  - 数据加密密钥
  - PIN加密密钥
  - 文件加密密钥
- **定义、标识不同的密钥**
  - 如：DES密钥 其中的8位效验位

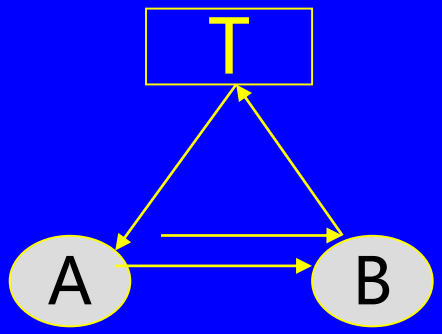
其他密钥分配方案



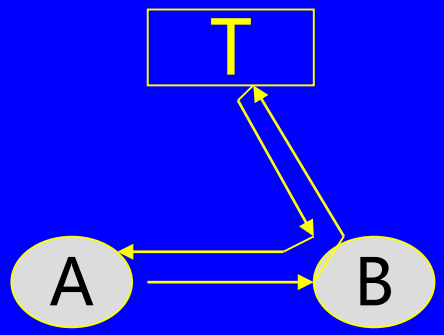
Wide-Mouth Frog协议



Needham-Schroeder协议



Yahalom协议



Otway-Rees协议

随机数的应用

- 密钥分配时的临时交互号nonce，防止重放攻击
- 会话密钥产生
- RSA公钥加密中密钥的产生
- 密文分组链接、密文反馈链接和输出反馈连接使用的IV。

随机数的产生

- **评价标准**
  - 分布一致性：序列中的随机数的分布应是一致的，即出现频率大约相等。
  - 独立性：序列中任何数不能由其他数推导出。
- **真随机数与伪随机数**
  - 真随机数：各个数之间的统计独立性而使序列不可预测，例如物理噪声发生器。
  - 伪随机数：序列可以经受住随机性检测，但并非统计随机的。

伪随机数的产生

- **线性拟合法(Lehmer 算法)**
  - 计算公式

$$X_{n+1}=(aX_n+c)\bmod m$$

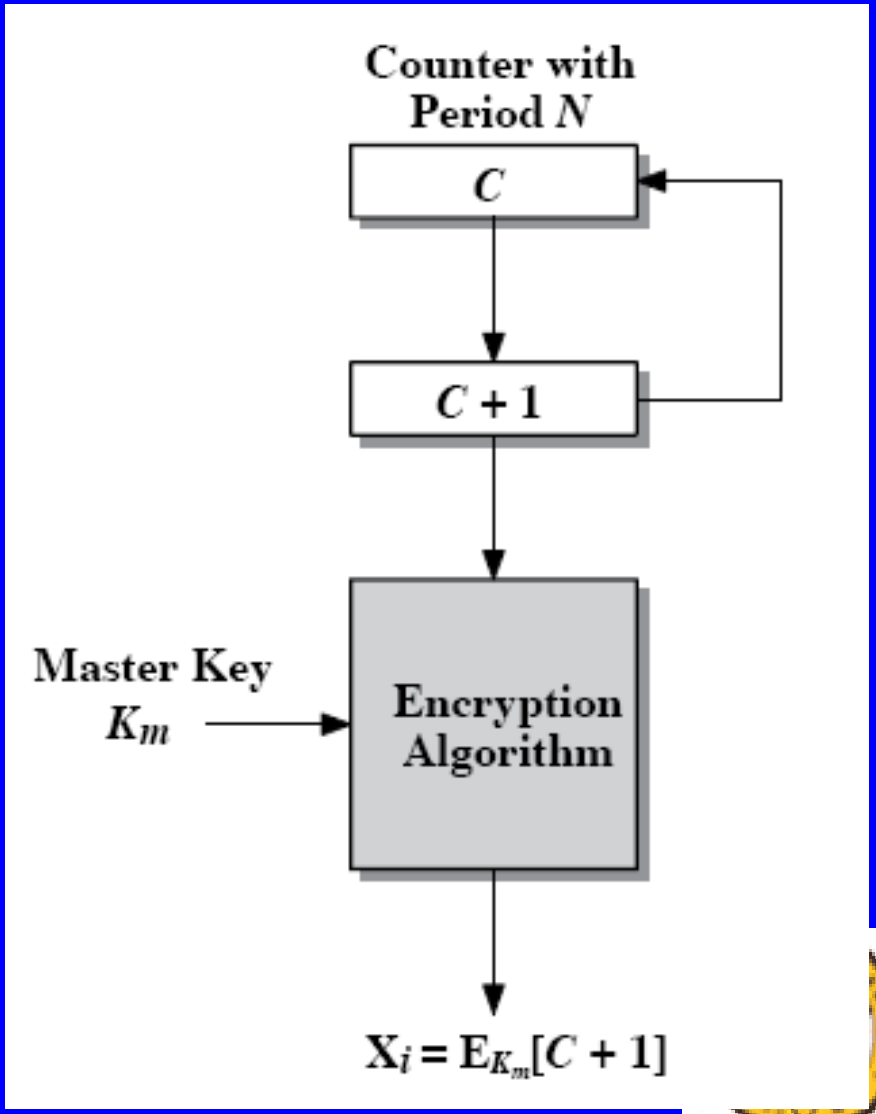
  - 参数选择（a,c和m的选择）
    - 例如：a=7,c=0,m=32且X_0 =1,序列{7,17,23,1,7,...}
    - m的选择标准一般与给定计算机标识的最大非负整数接近。
  - 弱点
    - 知道算法和参数后，只要知道一个随机数，就能获得后续的所有序列。
    - 知道采用线性拟合算法，根据随机数序列中的一小部分可以解得参数。

伪随机数的产生

- **评价随机数发生器的标准**
  - 生成函数应该是全周期的。
  - 产生的序列应看上去随机。
  - 生成函数可以用32位运算器方便地实现。

伪随机数的产生

- **密码编码学方法产生**
  - 循环加密
  - DES输出反馈模式



伪随机数的产生

– ANSI X9.17 伪随机数发生器

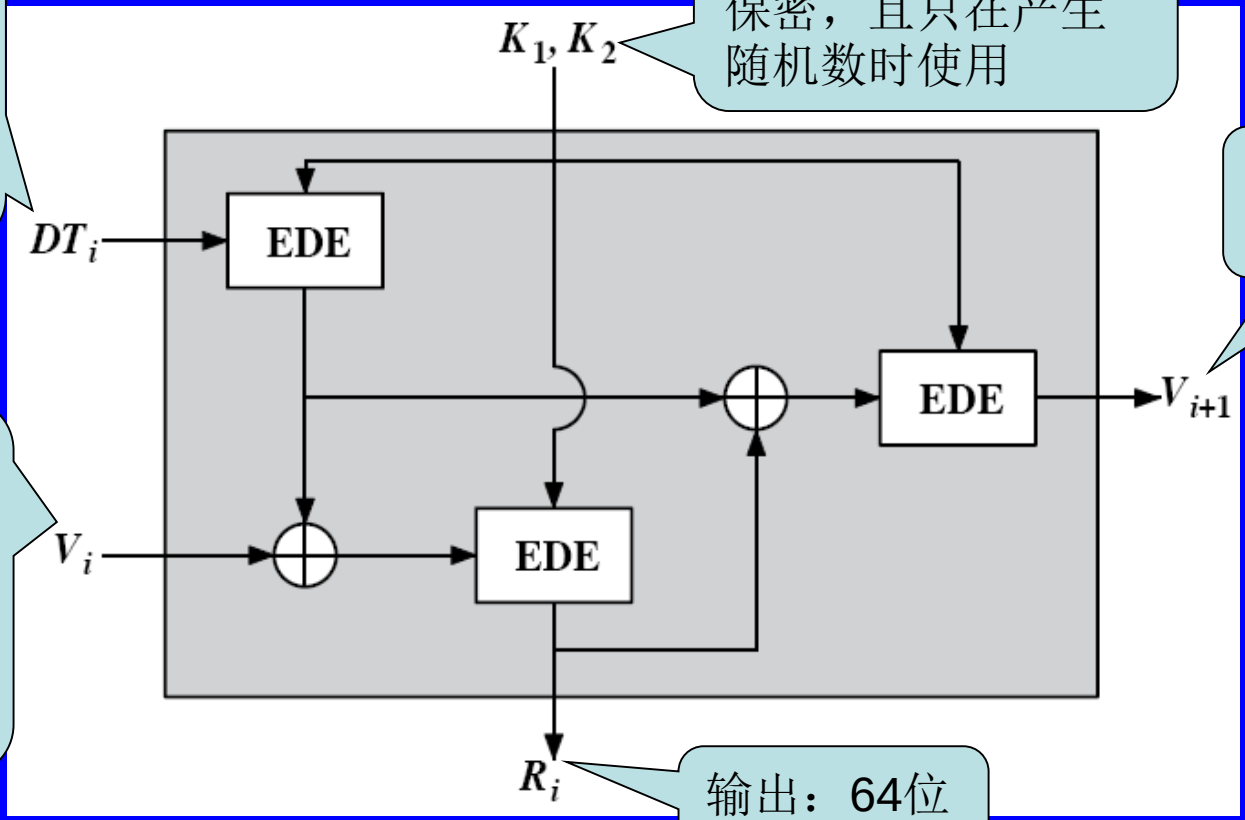
输入：64位，代表时间和日期，每次均改变。

密钥：56位密钥对，保密，且只在产生随机数时使用

输出：64位种子

输入：64位种子，可为任意值，在生成随机数的过程中改变

输出：64位伪随机数



BBS发生器

- BBS发生器是现在产生安全伪随机数的普遍方法。
- 算法描述
  - 选择两个大素数p,q
  - 确定n=p*q，选择随机数s，s与n互素
  - 递推产生随机数序列

$X_0 = s^2 \bmod n$

for i = 0 to ∞

$X_i = (X_{i-1})^2 \bmod n$

$B_i = X_i \bmod 2$ //每隔循环都取最低有效位

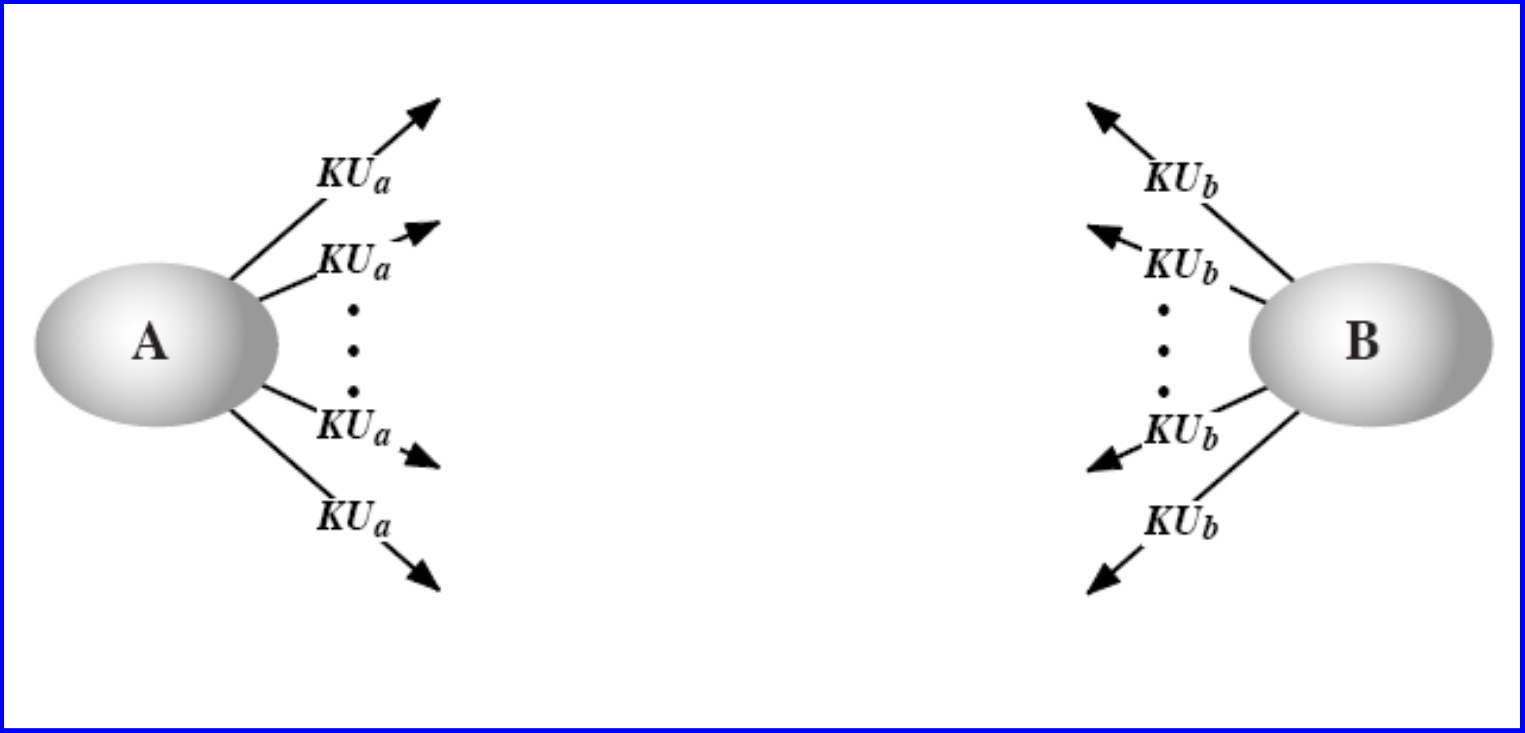
- 它的安全性是基于大数分解的困难。

非对称密钥的管理技术

密钥管理

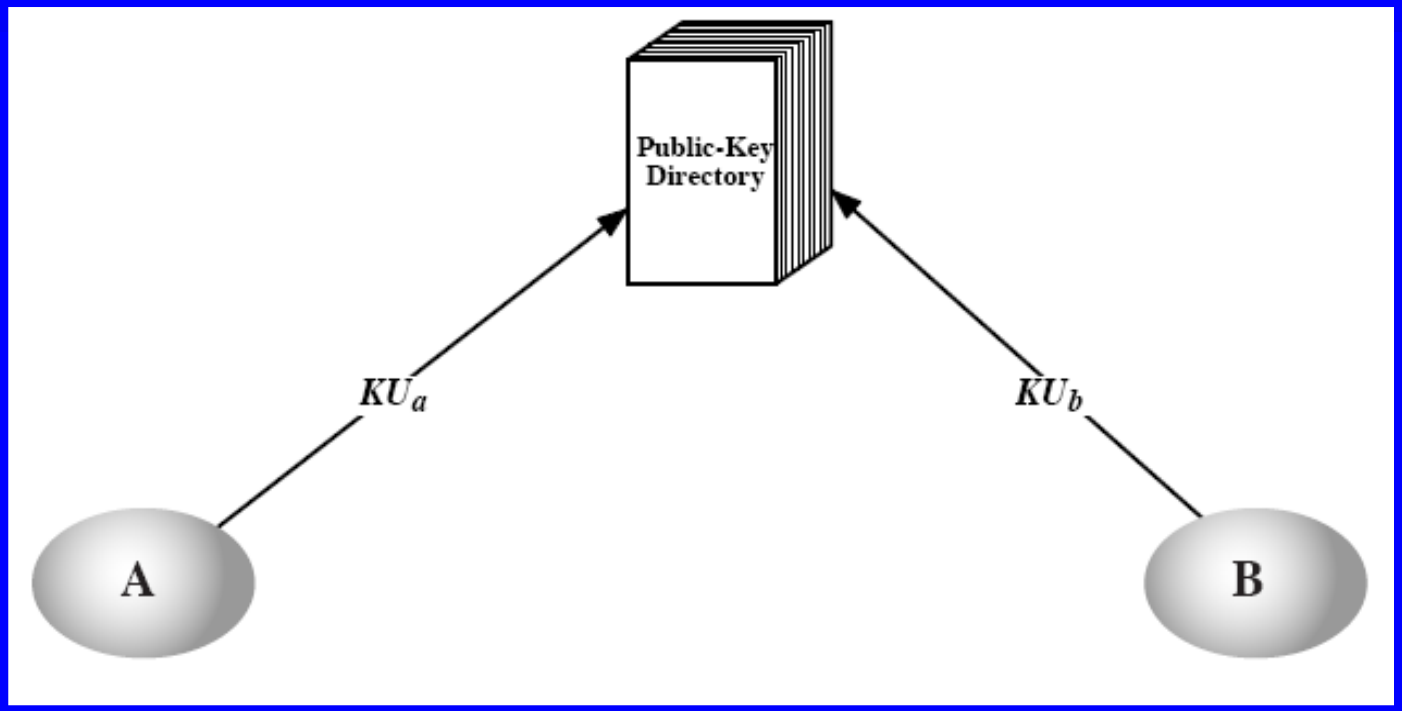
- **密钥分配**
  - 公钥分配
  - 公钥密码用于传统密码体制的密钥分配
- **公钥分配方法**
  - 公钥的公开发布
  - 公钥可访问目录
  - 公钥授权
  - 公钥证书

公开发布



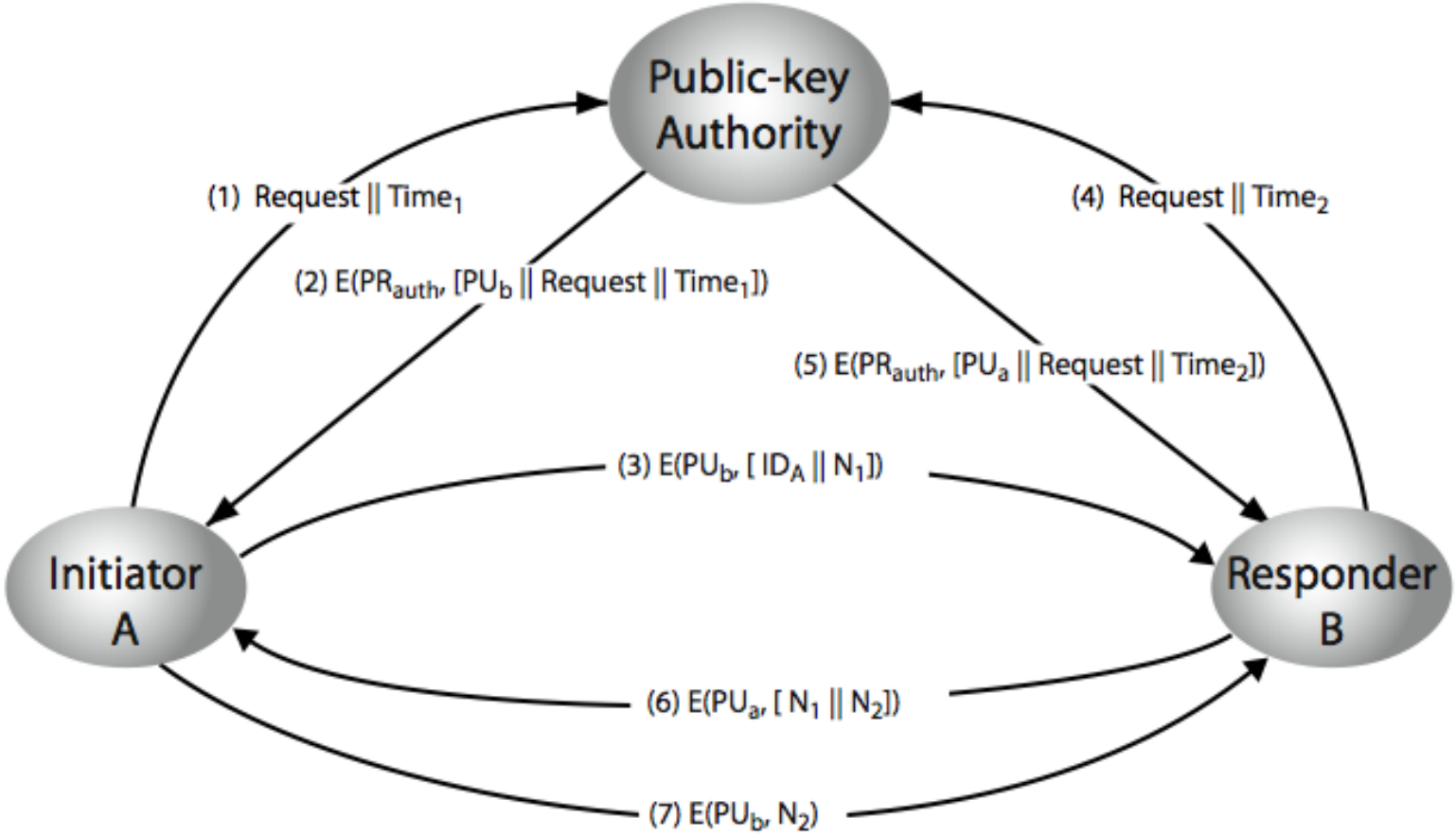
- 方法简便
- 缺乏安全性，任何人都可以伪造其他人的公钥

公开可访问目录

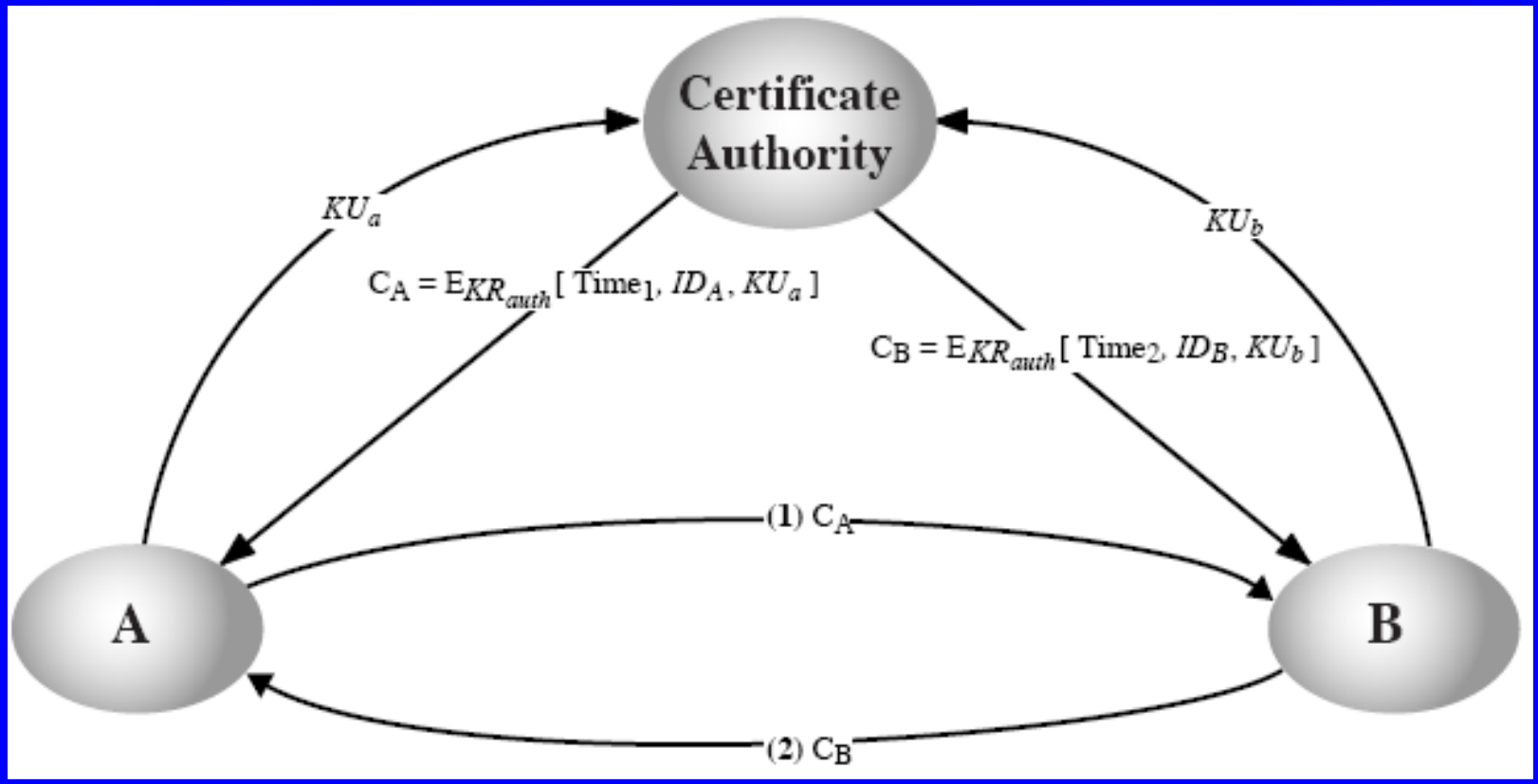


- 类似于电话号码簿
- 用户注册必须通过安全的认证通信
- 缺点：一旦攻击者获得管理员的私钥即可以传递伪造的公钥。

公 钥 授 权


Public-key Authority
(1) Request || Time1
(2) E(PRauth, [PUb || Request || Time1])
(4) Request || Time2
(5) E(PRauth, [PUa || Request || Time2])
(3) E(PUb, [IDA || N1])
Initiator A
Responder B
(6) E(PUa, [N1 || N2])
(7) E(PUb, N2)

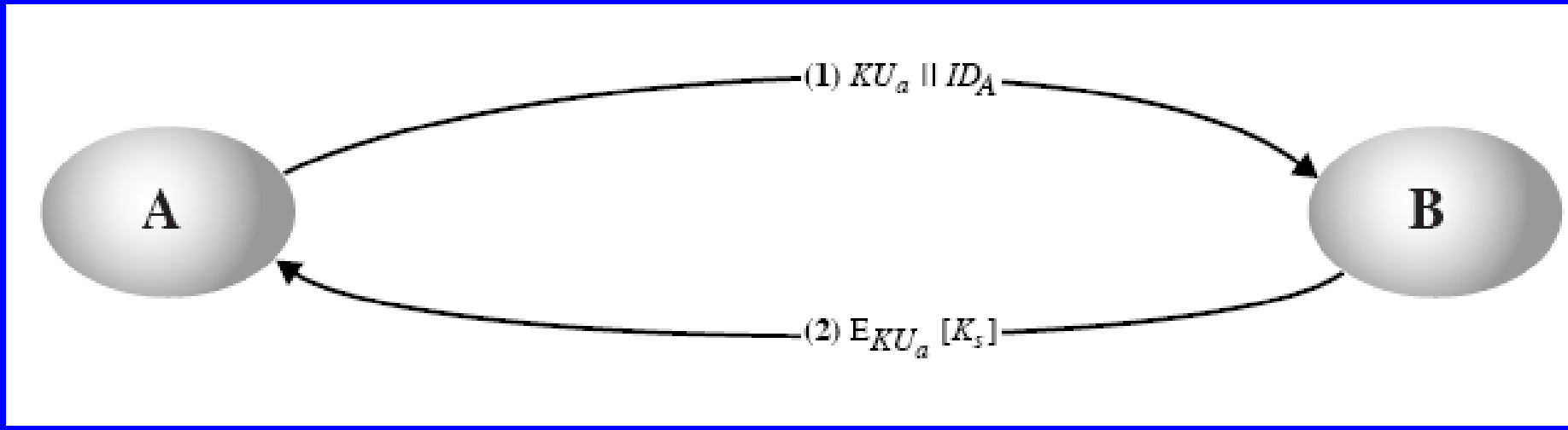
公钥证书



- 通信双方使用证书来交换密钥，而不是通过公钥管理员。
- 证书包含用户编号和公钥，由证书管理员产生。
- $C_A = E_{KRauth}[T, ID_A, K_{Ua}]$

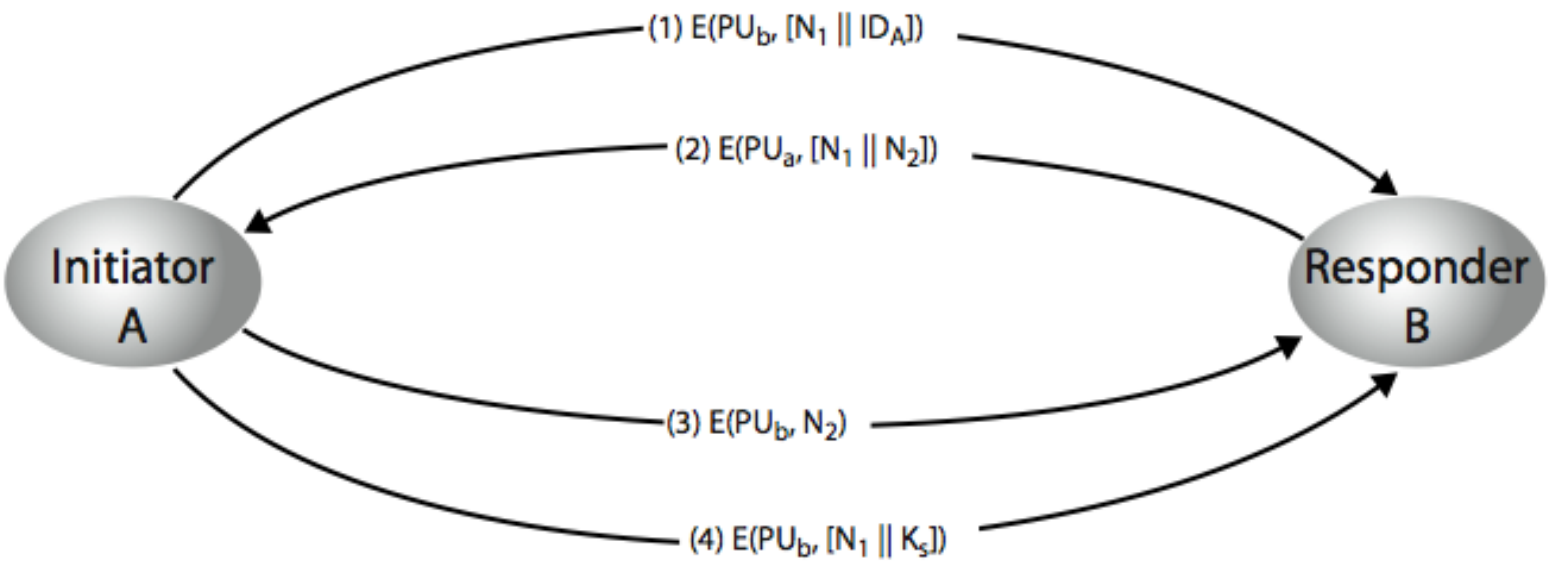
- 利用公钥密码分配传统密码体制的密钥
  - 简单的密钥分配
  - 具有保密性和真实性的密钥分配
  - 混合方法
    - 利用公钥密码分配主密钥
    - 利用主密钥实现秘密的会话密钥分配
    - KDC与每一个用户分别共享一个秘密的主密钥

简单的密钥分配



- 协议简单
- 可以抗窃听攻击
- 容易受中间人攻击
  - 重放截获的消息
  - 对消息进行替换

具有保密性和真实性的密钥分配方法



Diffie-Hellman密钥交换

- Diffie-Hellman密钥交换是第一个公钥方案，由Diffie & Hellman 于 1976年提出。
- 该算法仅能完成密钥交换，使得两个用户可以安全地交换密钥，用于后续的通讯过程。
- 算法的安全性建立在：
  - 求关于素数的模素数幂运算相对容易
  - 计算离散对数非常困难
  - 对于大素数，求离散对数被认为不可行
- 容易受到中间人攻击

公钥和私钥的产生

Global Public Elements

q	prime number
α	$\alpha < q$ and α a primitive root of q

User A Key Generation

Select private X_A	$X_A < q$
Calculate public Y_A	$Y_A = \alpha^{X_A} \bmod q$

User B Key Generation

Select private X_B	$X_B < q$
Calculate public Y_B	$Y_B = \alpha^{X_B} \bmod q$

密钥的产生

Generation of Secret Key by User A

$$K = (Y_B)^{X_A} \bmod q$$

Generation of Secret Key by User B

$$K = (Y_A)^{X_B} \bmod q$$

举例说明

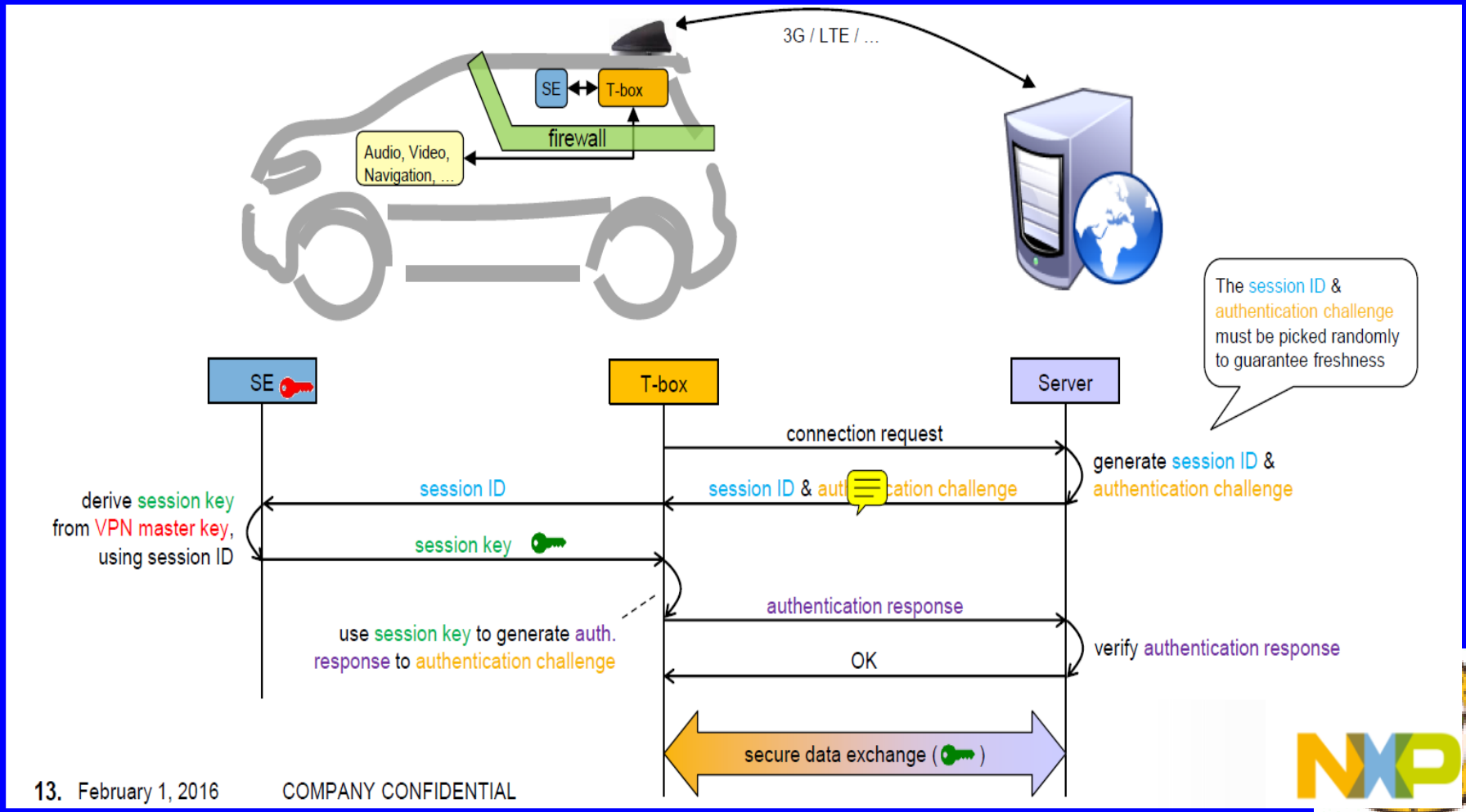
- 用户Alice和 Bob想交换密钥:
- 约定素数 P=353和 a=3
- 随机选择密钥:
  - A chooses x_A=97， B chooses x_B=233
- 计算公钥:
  - $y_A=3^{97} \bmod 353 = 40$ (Alice)
  - $y_B=3^{233} \bmod 353 = 248$ (Bob)
- 计算共享的会话密钥:

$K_{AB}= y_B^{x_A} \bmod 353 = 248^{97} = 160$ (Alice)

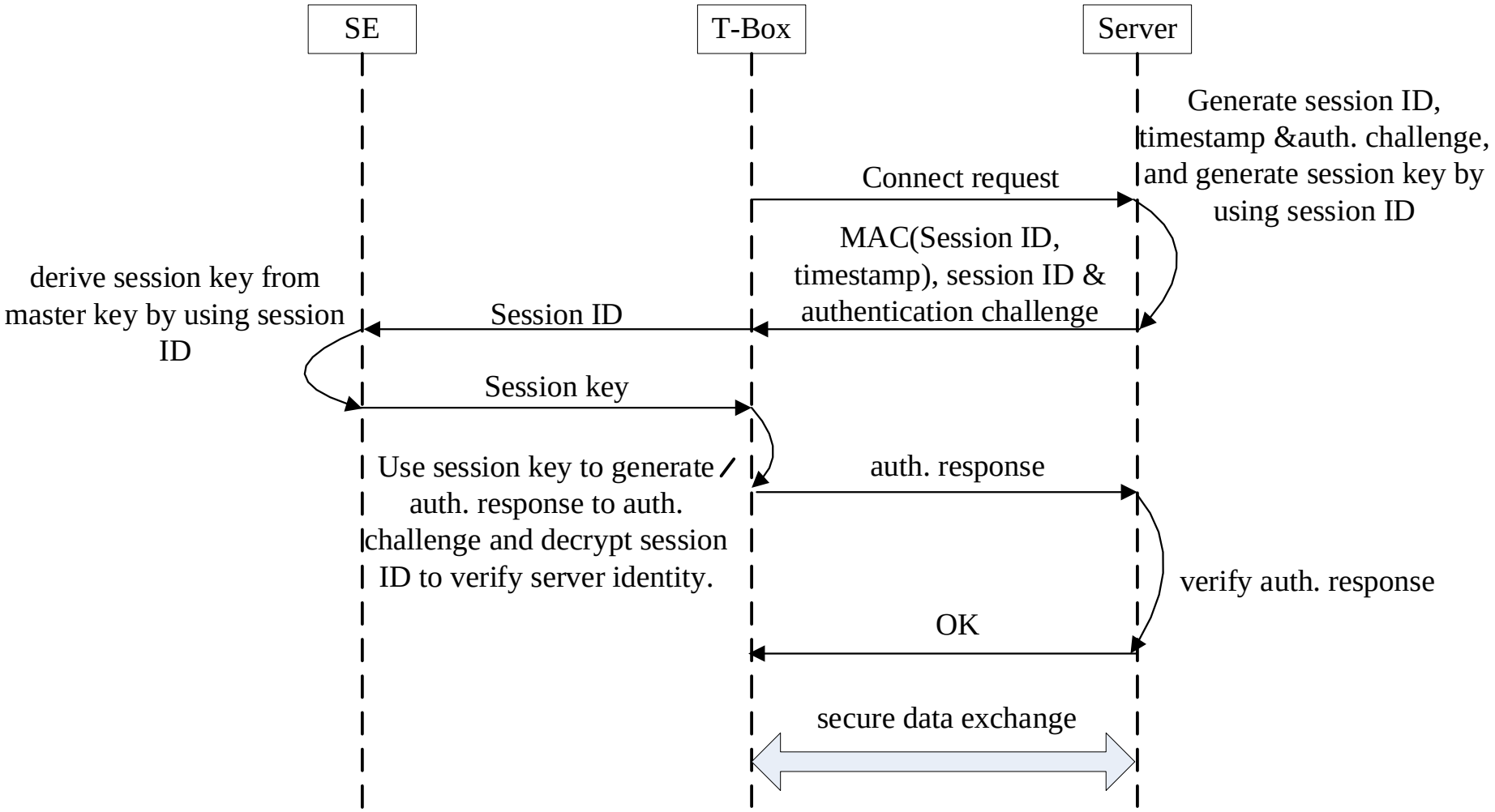
$K_{AB}= y_A^{x_B} \bmod 353 = 40^{233} = 160$ (Bob)

车联网安全方案分析

- 身份认证，密钥交换，传输数据的机密性



车内网络安全性分析


SE
T-Box
Server
Generate session ID, timestamp &auth. challenge, and generate session key by using session ID
Connect request
MAC(Session ID, timestamp), session ID & authentication challenge
derive session key from master key by using session ID
Session ID
Session key
Use session key to generate auth. response to auth. challenge and decrypt session ID to verify server identity.
auth. response
verify auth. response
OK
secure data exchange

问题：

1、会话密钥的生存周期如何确定？

2、证书在公钥分配中的作用是什么？

3、3个通信方如何利用DH算法进行密钥交换？

东南大学信息安全学院研究生学位课

密码学与安全协议

谢谢！