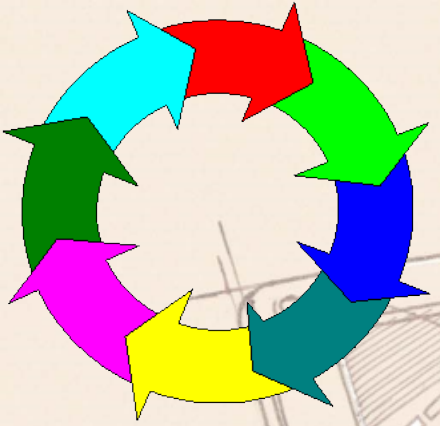

# 《信息平安技术》

## 补充：　身份认证

“身份认证”也称“身份鉴别”

——证实主体的真实身份与其声称的身份是否相符

- 以密码理论为根底
- 是访问控制和审计的前提
- 通过特定的协议和算法实现

# 补.1 身份认证根底

## 补.1.1 物理根底

1. 用户知道什么(What you know)——密码、口令、序列号、特定知识等
2. 用户拥有什么(What you have) ——身份证、护照、通行证、门禁卡等
3. 用户是谁 (Who are you) ——指纹、脸型、虹膜、声纹、DNA、笔迹等生物特征

# 补.1.2 数学根底

1. 基于知识的证明——P为了向V证明自己知道什么信息，说出或无意透露出得到该信息的某些知识
2. 零知识证明——P使V确信自己知道什么信息，但却不泄露任何与得到该信息相关的知识

例1：钱琪向雷蕾声称他知道程嘉的偶像是谁，雷蕾表示不信，钱琪有两种证明方式：

①钱琪不得已说出是章子怡，于是在让雷蕾相信的同时也让她知道了这一信息——基于知识的证明

②钱琪让程嘉作证，于是雷蕾相信了，但雷蕾仍不知程嘉的偶像是谁——零知识证明

例2：钱琪要向王雪梅证明自己拥有某个房间的钥匙，假设该房间只能用该钥匙翻开，而其他任何方法都打不开。钱琪有两种证明方式：

①钱琪把钥匙交给王雪梅，王雪梅用这把钥匙翻开该房间，从而证明钱琪拥有该房间的钥匙。——基于知识的证明

②王雪梅确定该房间内有某一物体，钱琪用自己拥有的钥匙翻开该房间的门，然后把物体拿出来出示给王雪梅，从而证明自己确实拥有该房间的钥匙，但王雪梅始终不能看到钥匙的样子，从而防止了钥匙的泄露。——零知识证明

例3：王雪梅拥有钱琪的公钥，但没有见过钱琪，而钱琪见过王雪梅的照片，偶然一天两人见面了，钱琪认出了王雪梅，但王雪梅不能确定面前的人是否是钱琪，这时钱琪要向王雪梅证明自己是钱琪，也有两种证明方法：

①钱琪把自己的私钥给王雪梅，王雪梅用这个私钥对某个数据加密，然后用钱琪的公钥解密，如果正确，那么王雪梅证实对方是钱琪，但钱琪不情愿地泄漏了自己的私钥。——基于知识的证明

②王雪梅给出一个随机值，钱琪用自己的私钥对其加密，然后把加密后的数据交给王雪梅，王雪梅用钱琪的公钥解密，如果能够得到原来的随机值，那么证明对方是钱琪，但王雪梅并未得到钱琪的私钥。——零知识证明

# 补.1.3 协议根底

## 1. 身份认证协议的类型

- 双向认证协议
- 单向认证协议

- 基于对称密码的认证协议
- 基于公钥密码的认证协议

- 双方直接认证协议
- 基于第三方的认证协议

2. 通过会话密钥〔用于下次认证通信的对称密钥〕提供认证协议的平安性
3. 重放攻击——C用截获得的A已用过的认证信息假冒A与B建立通信

典型例子：A与B建立认证会话时，C通过窃听获得A发给B的认证信息M；在A、B结束通信后，C用M冒充A与B建立认证会话，使B信以为真，然后与C进行通信。

4. 防重放的方法之一——时间戳(Time Stamp)
5. “时戳”——代表当前时刻的数
6. 根本思想——A接收一个消息当且仅当其包含一个对A而言足够接近当前时刻的时戳
7. 原理——重放的时戳将相对远离当前时刻
8. 时钟要求——通信各方的计算机时钟保持同步
9. 处理方式——设置大小适当的时间窗〔间隔〕，越大越能包容网络传输延时，越小越能防重放攻击
10. 适用性——用于非连接性的对话〔在连接情形下双方时钟假设偶然出现不同步，那么正确的信息可能会被误判为重放信息而丢弃，而错误的重放信息可能会当作最新信息而接收〕

5. 防止重放的方法之二——提问-应答
6. “现时”——与当前事件有关的一次性随机数N（互不重复即可）
7. 根本做法——期望从B获得消息的A事先发给B一个现时N，并要求B应答的消息中包含N或f(N)，f是A、B预先约定的简单函数
8. 原理——A通过B回复的N或f(N)与自己发出是否一致来判定本次消息是不是重放的
9. 时钟要求——无
10. 适用性——用于连接性的对话

# 补.2 身份认证协议

## 补.2.1 双向认证协议

1. 基于对称密码和可信第三方的双向认证协议
2. N-S协议（由Roger Needham和Michael Schroeder创造）：
3. A→KDC : IDA || IDB || N1
4. KDC→A : EKa[Ks || IDB || N1 || EKb[Ks || IDA ]]
5. A→B : EKb[Ks || IDA ]
6. B→A : EKs [N2]
7. A→B : EKs[f(N2 )]

其中：Ka、Kb分别是KDC与A、B的共享密钥；Ks是A、B之间的会话密钥。$N_1$用于防C冒充KDC对A的重防攻击；$N_2$用于防C冒充A对B的重防攻击。

漏洞：假设攻击者C已经知道A、B之间以前用过的会话密钥K’，那么C可在第3)步用以前所截获的EKb[K’ || IDA ]冒充A，实施重放攻击：

C→B : EKb[K’ || IDA ]

只要B不记得K’是先前用过的会话密钥，便会以为是一次新的认证过程，从而做第4)步：

B→A : EK’ [N2]

C假设能成功地阻止A收到此握手信息，那么就可以用K’解密出N2，并用K’加密得EK’[f(N2 )]冒充A做第5)步：

C→B : EK’[f(N2 )]

从而假冒A与B建立通信

**Denning协议〔使用时戳T〕：**

**A→KDC : IDA || IDB**

**KDC→A : EKa[Ks || IDB || T || EKb[Ks || IDA || T]]**

**A→B : EKb[Ks || IDA || T]**

**B→A : EKs [N1]**

**A→B : EKs[f(N1 )]**

**假设|Clock-T|<Δt1+Δt2，那么不是重放。其中Clock是当前时刻、Δt1是KDC与A或B时钟之差估计值、Δt2是网络延迟估计值**

**注：此法要求A、B的时钟同步**

## 2. 基于公钥密码和可信第三方的双向认证协议

**WOO92b协议：**

1) $A \rightarrow KDC : ID_A \| ID_B$
2) $KDC \rightarrow A : E_{KR}[KU_b \| ID_B]$
3) $A \rightarrow B : E_{KUb}[ID_A \| N_a]$
4) $B \rightarrow KDC : ID_B \| ID_A \| E_{KU}[N_a]$
5) $KDC \rightarrow B :$

$E_{KR}[KU_a \| ID_A] \| E_{KUb}[E_{KR}[Ks \| ID_A \| ID_B \| N_a]]$

6) $B \rightarrow A : E_{KUa}[E_{KR}[Ks \| ID_A \| ID_B \| N_a] \| N_b]$
7) $A \rightarrow B : E_{Ks}[N_b]$

其中KU、KR分别是KDC的公钥、私钥; $KU_a$和$KR_a$、$KU_b$和$KR_b$分别是A、B的公钥和私钥

A、B都向KDC索要对方的公钥，并用对方能否解密用其公钥加密的消息来证明对方的身份

# 补.2.2 单向认证协议

1. 基于对称密码的单向〔A向B证明〕认证协议
2. A→B : IDA || N1
3. B→A : EKab[Ks || IDB || f(N1) || N2]
4. A→B : EKs[f(N2 )]

其中Kab是A、B的共享密钥

缺点：双方都必须同时在线并等待对方的答复