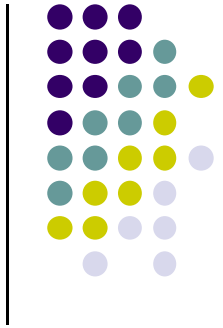
Kerberos协议

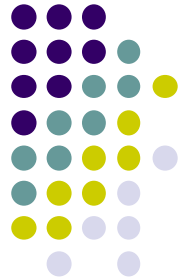


- 问题提出
- 协议

问题

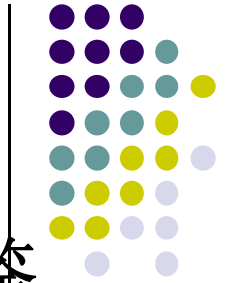
- 在一个开放的分布式网络环境中，用户通过工作站访问服务器上提供的服务。
  - 服务器应能够限制非授权用户的访问并能够认证对服务的请求。
  - 工作站不能够被网络服务所信任其能够正确地认定用户，即工作站存在三种威胁。
    - 一个工作站上一个用户可能冒充另一个用户操作；
    - 一个用户可能改变一个工作站的网络地址，从而冒充另一台工作站工作；
    - 一个用户可能窃听他人的信息交换，并用回放攻击获得对一个服务器的访问权或中断服务器的运行。

Kerberos要解决的问题



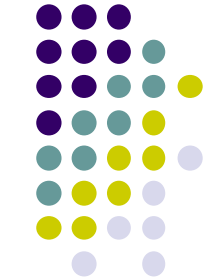
- 所有上述问题可以归结为一个非授权用户能够获得其无权访问的服务或数据。
- 不是为每一个服务器构造一个身份认证协议，**Kerberos**提供一个中心认证服务器，提供用户到服务器和服务器到用户的认证服务。
- **Kerberos**采用传统加密算法（无公钥体制）。

Kerberos



- 是美国麻省理工学院（MIT）开发的一种身份鉴别服务。 **http://web.mit.edu/kerberos/**
- **“Kerberos”**的本意是希腊神话中守护地狱之门的守护者。
- **Kerberos**提供了一个集中式的认证服务器结构，认证服务器的功能是实现用户与其访问的服务器间的相互鉴别。
- **Kerberos**建立的是一个实现身份认证的框架结构。
- 其实现采用的是对称密钥加密技术，而未采用公开密钥加密。
- 公开发布的**Kerberos**版本包括版本**4**和版本**5 (RFC1510)**。

信息系统资源保护的动机



- 单用户单机系统。用户资源和文件受到物理上的安全保护；
- 多用户分时系统。操作系统提供基于用户标识的访问控制策略，并用**logon**过程来标识用户。
- **Client/Server**网络结构。由一组工作站和一组分布式或中心式服务器组成。

C/S环境下三种可能的安全方案

- 相信每一个单独的客户工作站可以保证对其用户的识别，并依赖于每一个服务器强制实施一个基于用户标识的安全策略。
- 要求客户端系统将它们自己向服务器作身份认证，但相信客户端系统负责对其用户的识别。
- 要求每一个用户对每一个服务证明其标识身份，同样要求服务器向客户端证明其标识身份。

Kerberos的解决方案

- Kerberos支持以上三种策略。
- 在一个分布式的client/server体系机构中采用一个或多个Kerberos服务器提供一个认证服务。
- 总体方案是提供一个可信第三方的认证服务。

Kerberos系统应满足的要求

- 安全。网络窃听者不能获得必要信息以假冒其它用户；**Kerberos**应足够强壮以至于潜在的敌人无法找到它的弱点连接。
- 可靠。**Kerberos**应高度可靠，并且应借助于一个分布式服务器体系结构，使得一个系统能够备份另一个系统。
- 透明。理想情况下，用户除了要求输入口令以外应感觉不到认证的发生。
- 可伸缩。系统应能够支持大数量的客户和服务器。

Kerberos设计思路

- 基本思路：
  - 使用一个（或一组）独立的认证服务器（AS — Authentication Server），来为网络中的用户（C）提供身份认证服务；
  - 认证服务器 (AS)，用户口令由 AS 保存在数据库中；
  - AS 与每个服务器（V）共享一个惟一保密密钥（K_v）（已被安全分发）。
- 会话过程：

(1) $C \rightarrow AS: \quad ID_C \| P_C \| ID_v$

(2) $AS \rightarrow C: \quad Ticket$

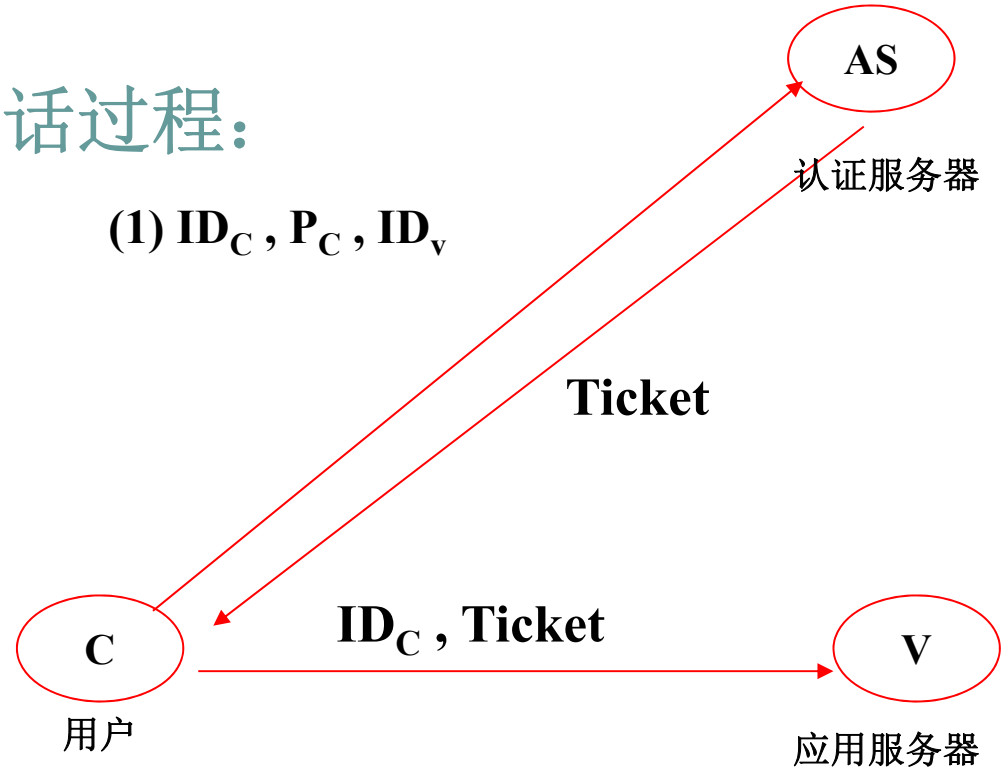
(3) $C \rightarrow V: \quad ID_C \| Ticket$

- 其中：

$Ticket = E_{Kv}[ID_C \| AD_C \| ID_v]$

Kerberos设计思路（续）

- 会话过程：



搜索数据库看用户是否合法

如果合法，验证用户口令是否正确

如果口令正确，检查是否有权限访问服务器V

用与AS共享密钥解密票据

检查票据中的用户标识与网络地址是否与用户发送的标识及其地址相同

如果相同，票据有效，认证通过

$Ticket = E_{Kv}[ID_C , AD_C , ID_v]$

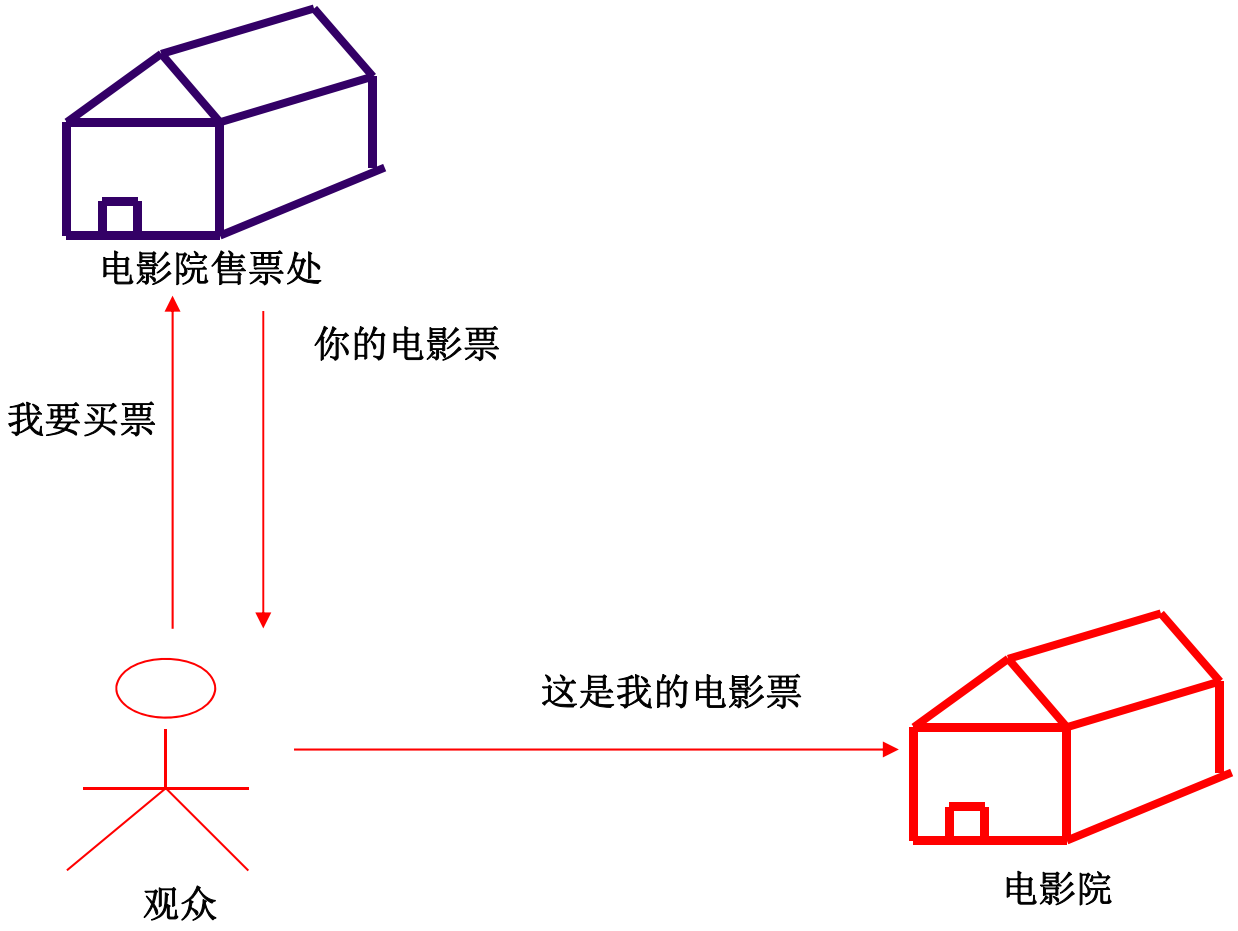
ID_C：用户C的标识

P_C：用户口令

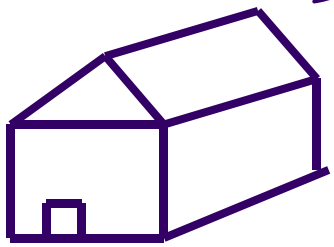
ID_v：服务器标识

AD_C：用户网络地址

Kerberos设计思路（续）



Kerberos设计思路（续）



电影院售票处

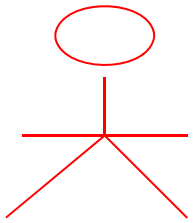
问题：如何买票

答案：出示信用卡卡号和密码

问题之一：信用卡问题

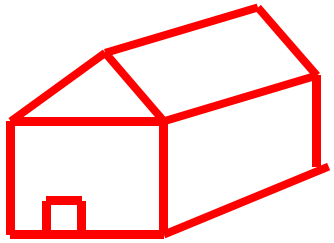
你的电影票

我要买票，
这是我的信
用卡密码



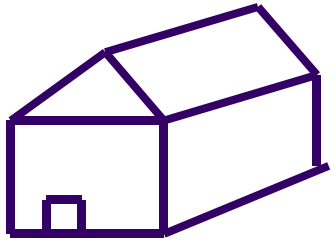
观众

这是我的电影票



电影院

Kerberos设计思路（续）

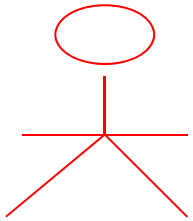


电影院售票处

问题之二：票的有效期问题

我要买票，
这是我的信
用卡密码

你的电影票

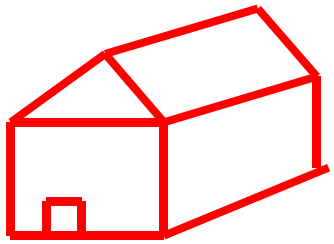


观众

这是我的电影票

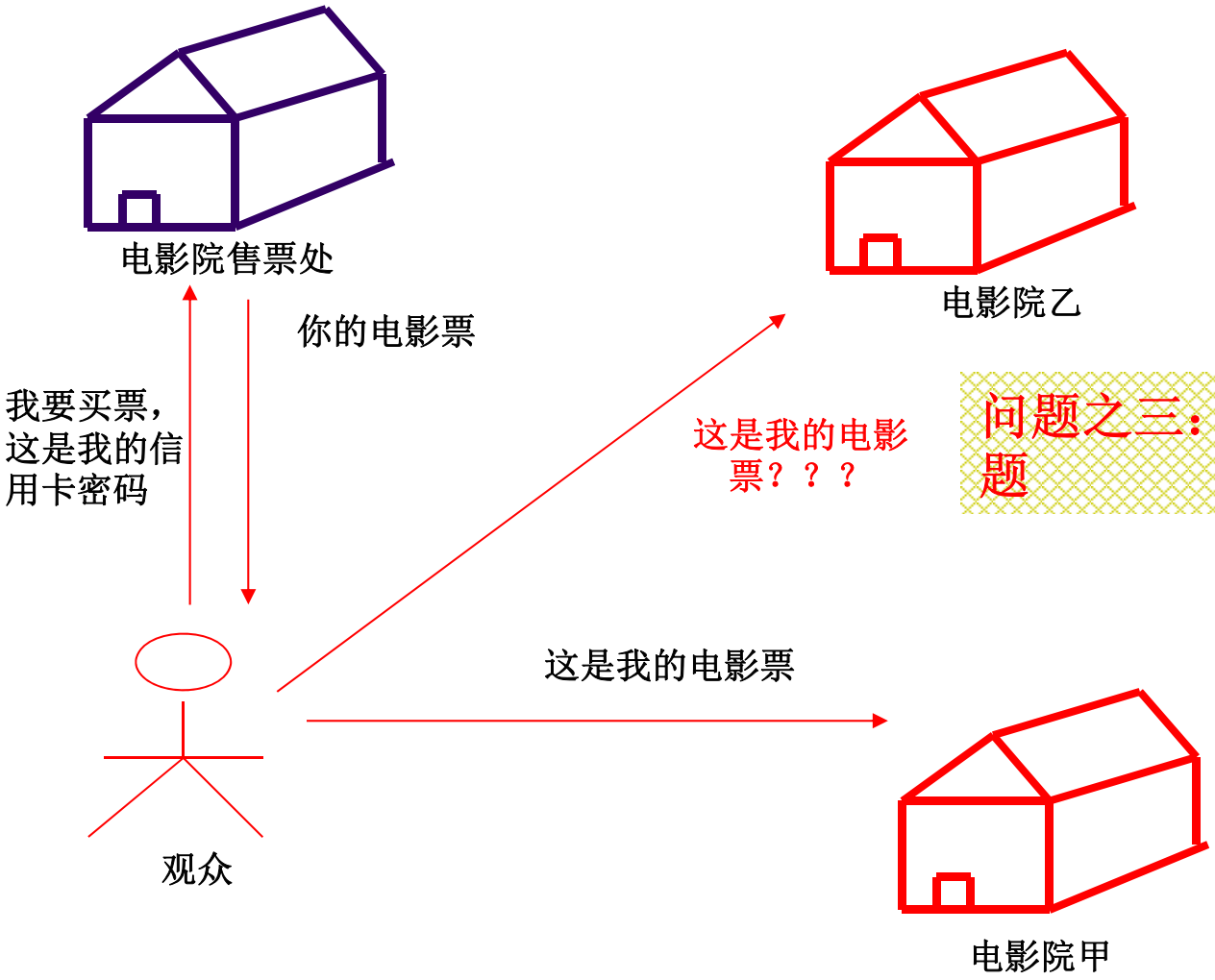
这是我的电影票

这是我的电影票



电影院

Kerberos设计思路（续）



Kerberos设计思路（续）

上述协议问题？

上述协议的问题：

（1）口令明文传送

（2）票据的有效性（多次使用）

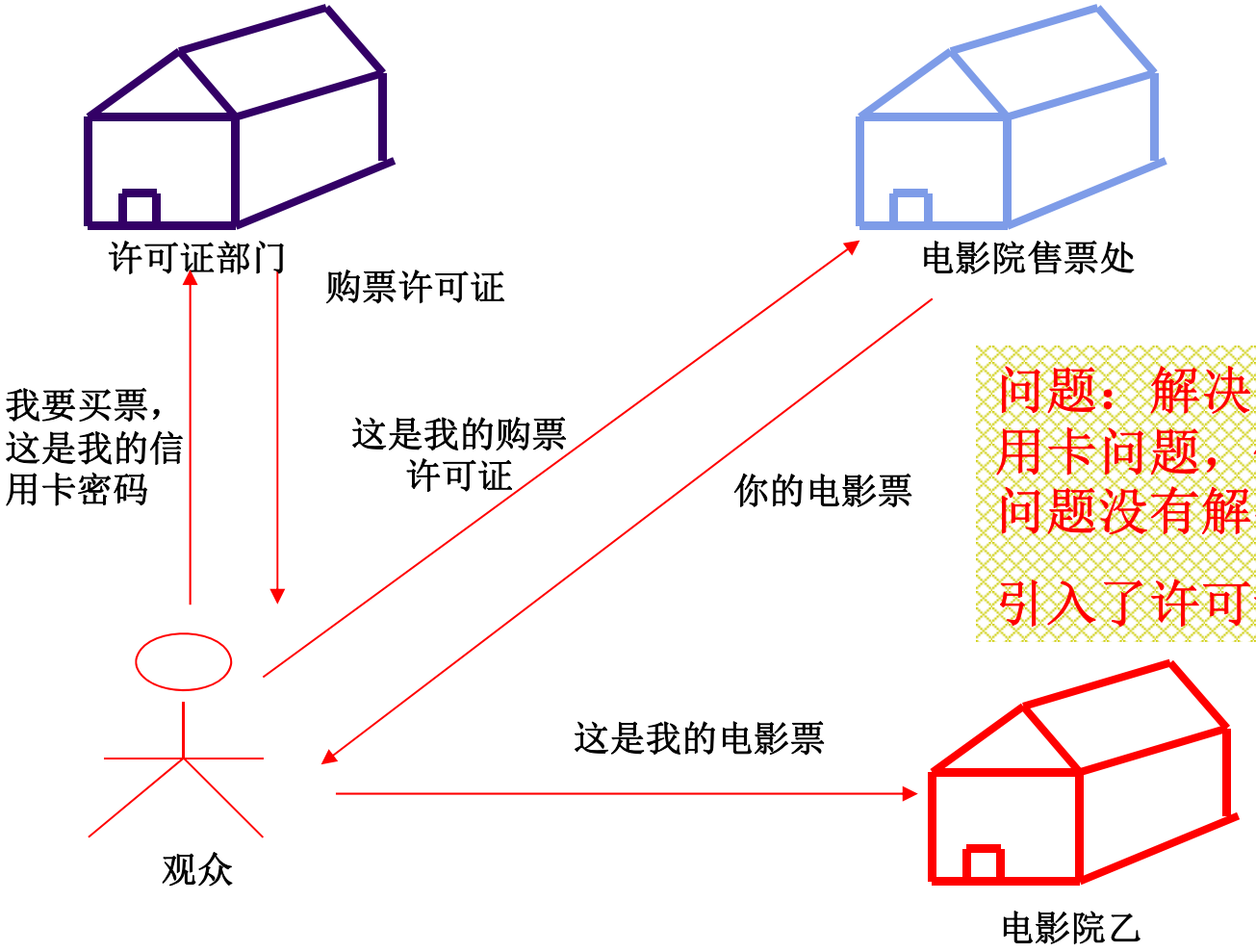
（3）访问多个服务器则需多次申请票据（即口令多次使用）

如何解决

Kerberos设计思路（续）

- 问题：
  - 用户希望输入口令的次数最少。
  - 口令以明文传送会被窃听。
- 解决办法
  - 票据重用（**ticket reusable**）。
  - 引入票据许可服务器（**TGS - ticket-granting server**）
    - 用于向用户分发服务器的访问票据；
    - 认证服务器 **AS** 并不直接向客户发放访问应用服务器的票据，而是由 **TGS** 服务器来向客户发放。

Kerberos设计思路（续）



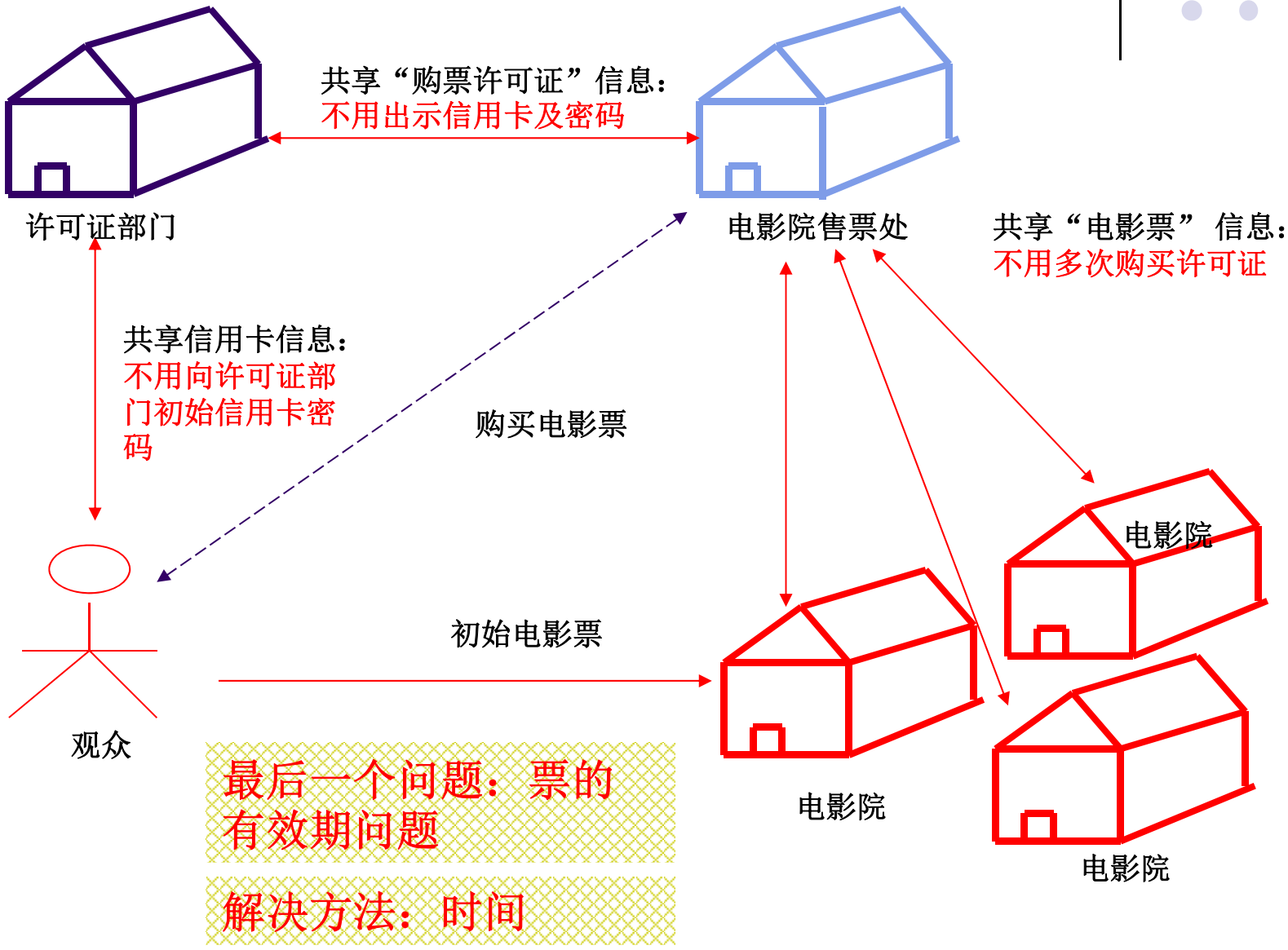
问题：解决了重复使用信用卡问题，但是其他两个问题没有解决

引入了许可证可信问题

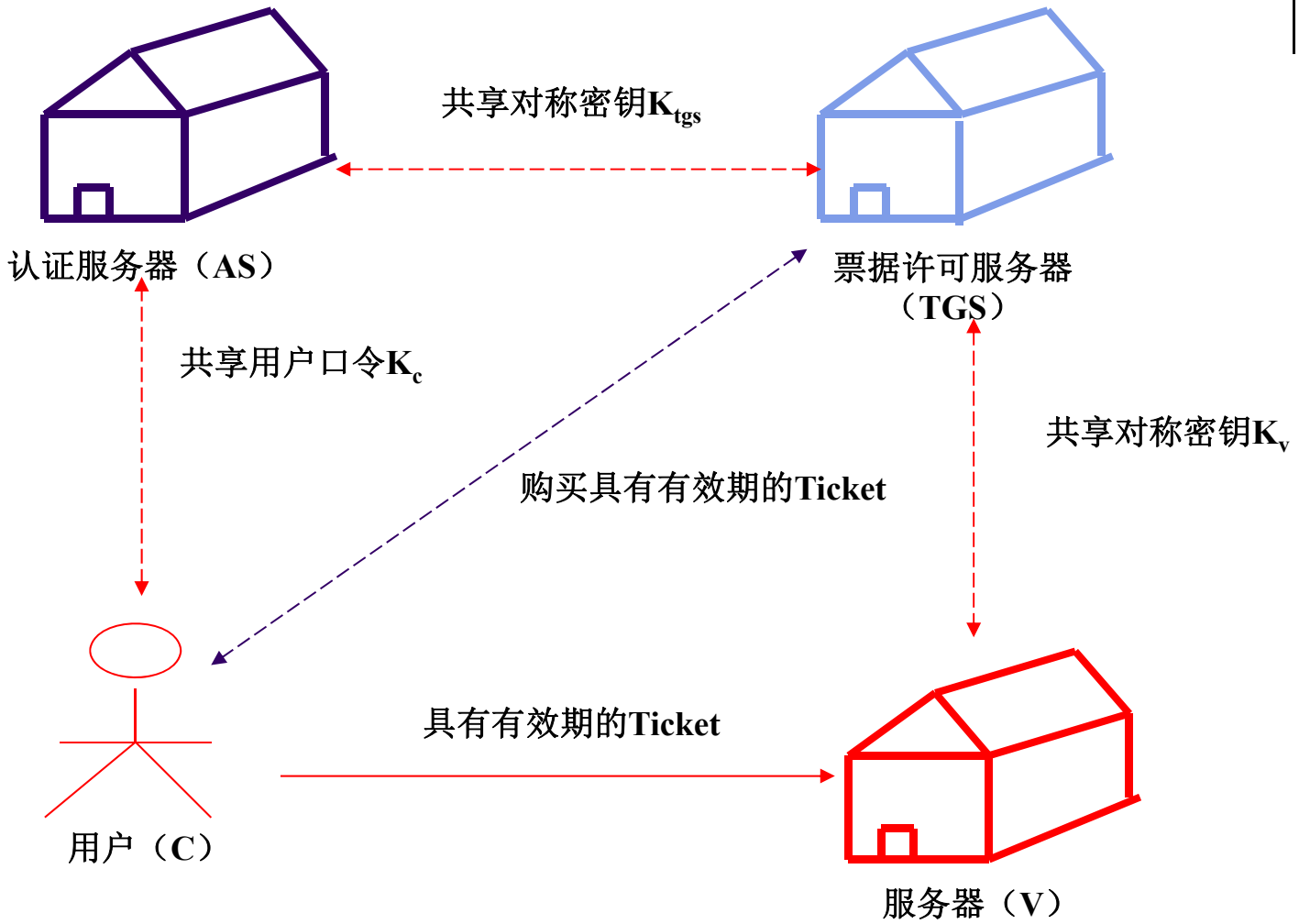
Kerberos的票据

- 两种票据
  - 票据许可票据（Ticket granting ticket）
    - 客户访问 TGS 服务器需要提供的票据，目的是为了申请某一个应用服务器的 “服务许可票据”；
    - 票据许可票据由 AS 发放；
    - 用 $\mathbf{Ticket_{tgs}}$ 表示访问 TGS 服务器的票据；
    - $\mathbf{Ticket_{tgs}}$ 在用户登录时向 AS 申请一次，可多次重复使用；
  - 服务许可票据（Service granting ticket）
    - 是客户时需要提供的票据；
    - 用 TicketV 表示访问应用服务器 V 的票据。

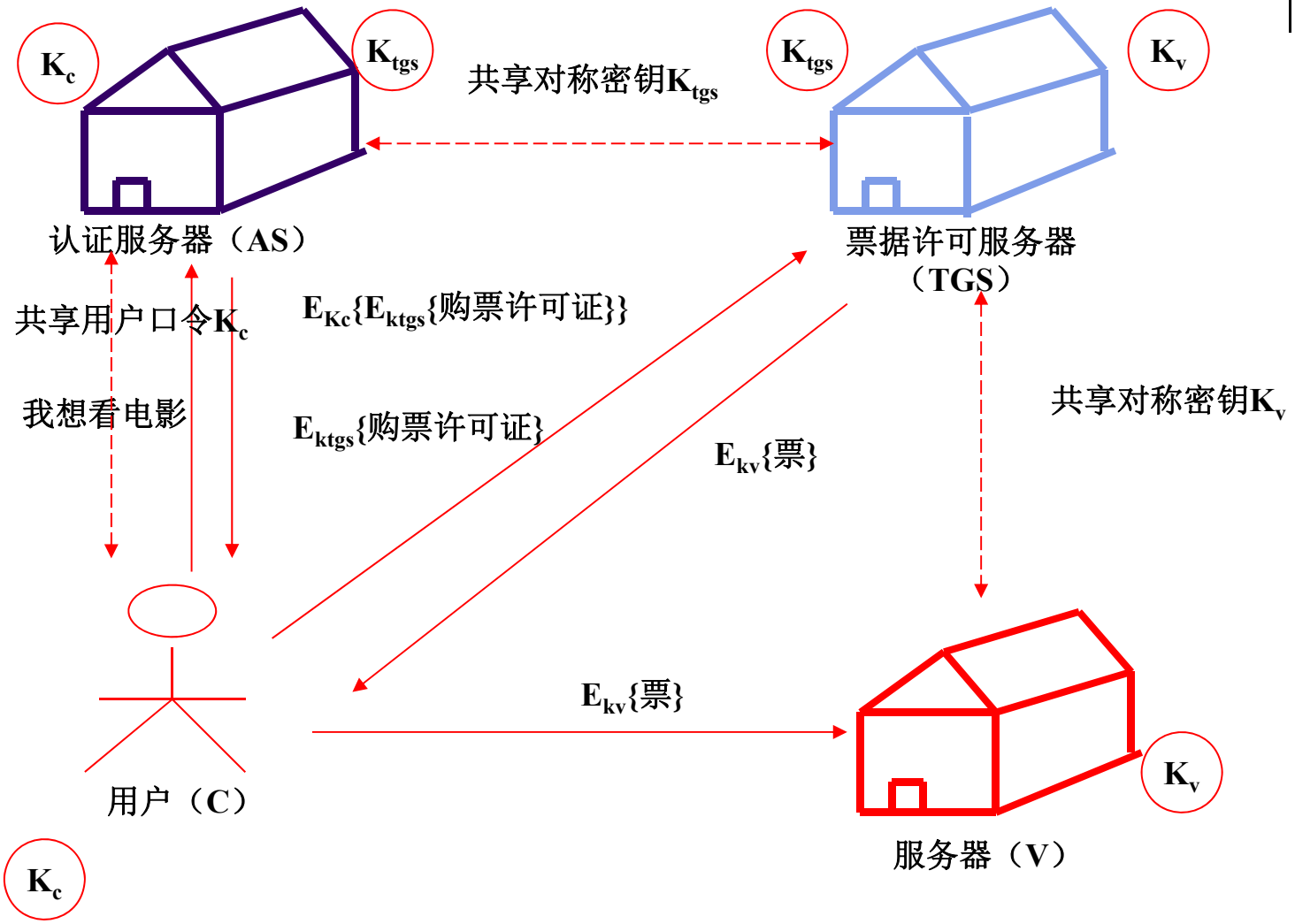
Kerberos设计思路（续）



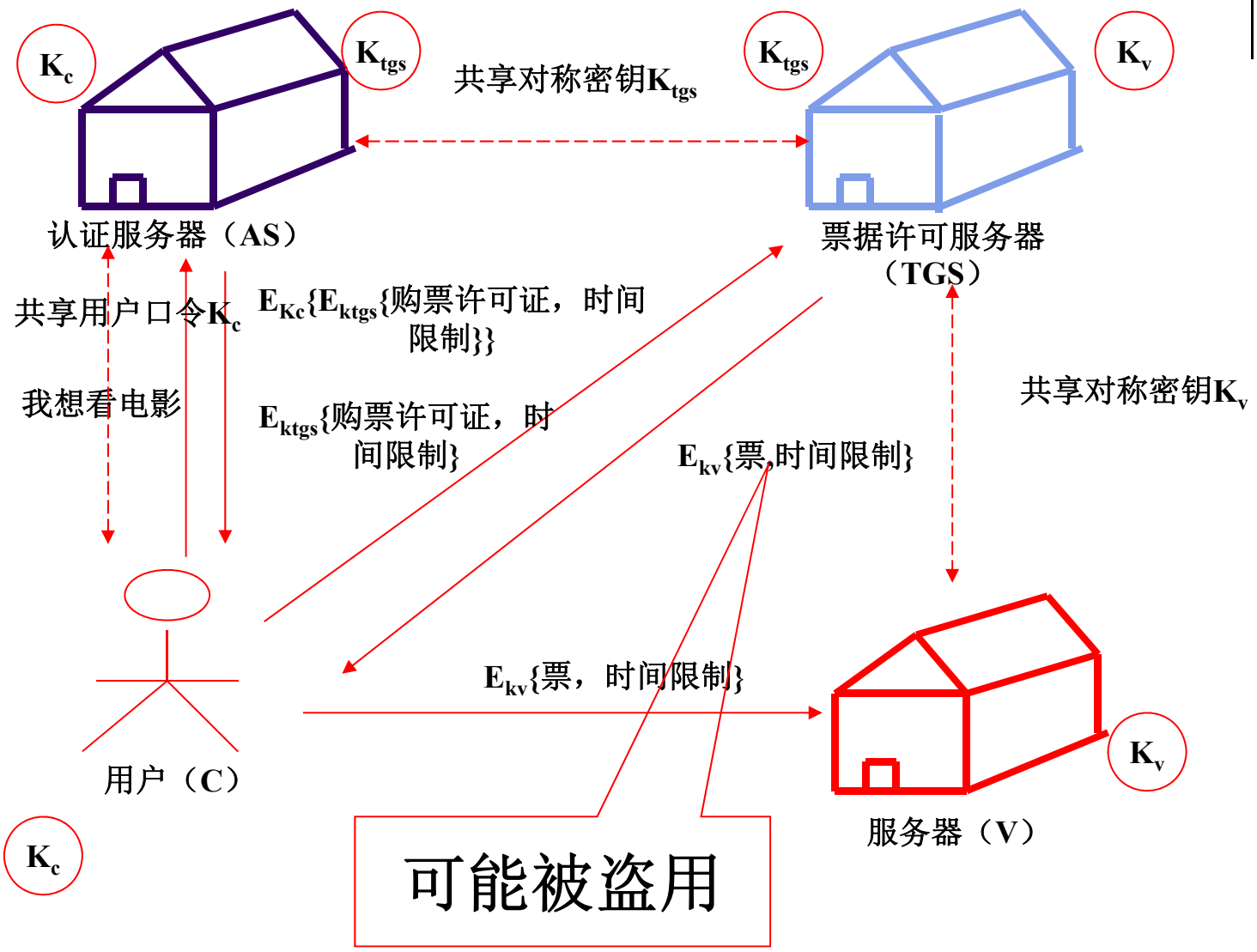
Kerberos设计思路（续）



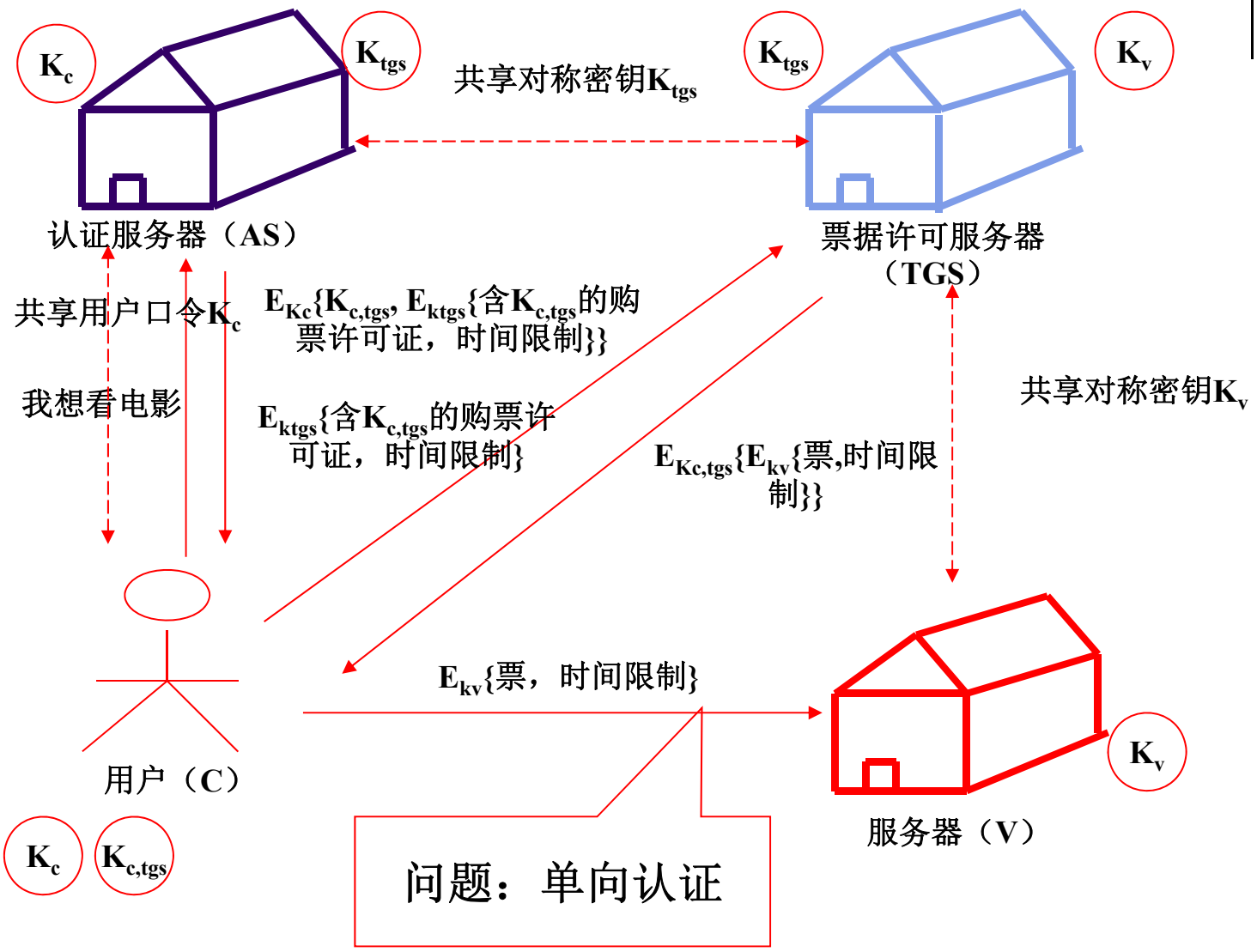
Kerberos设计思路（续）



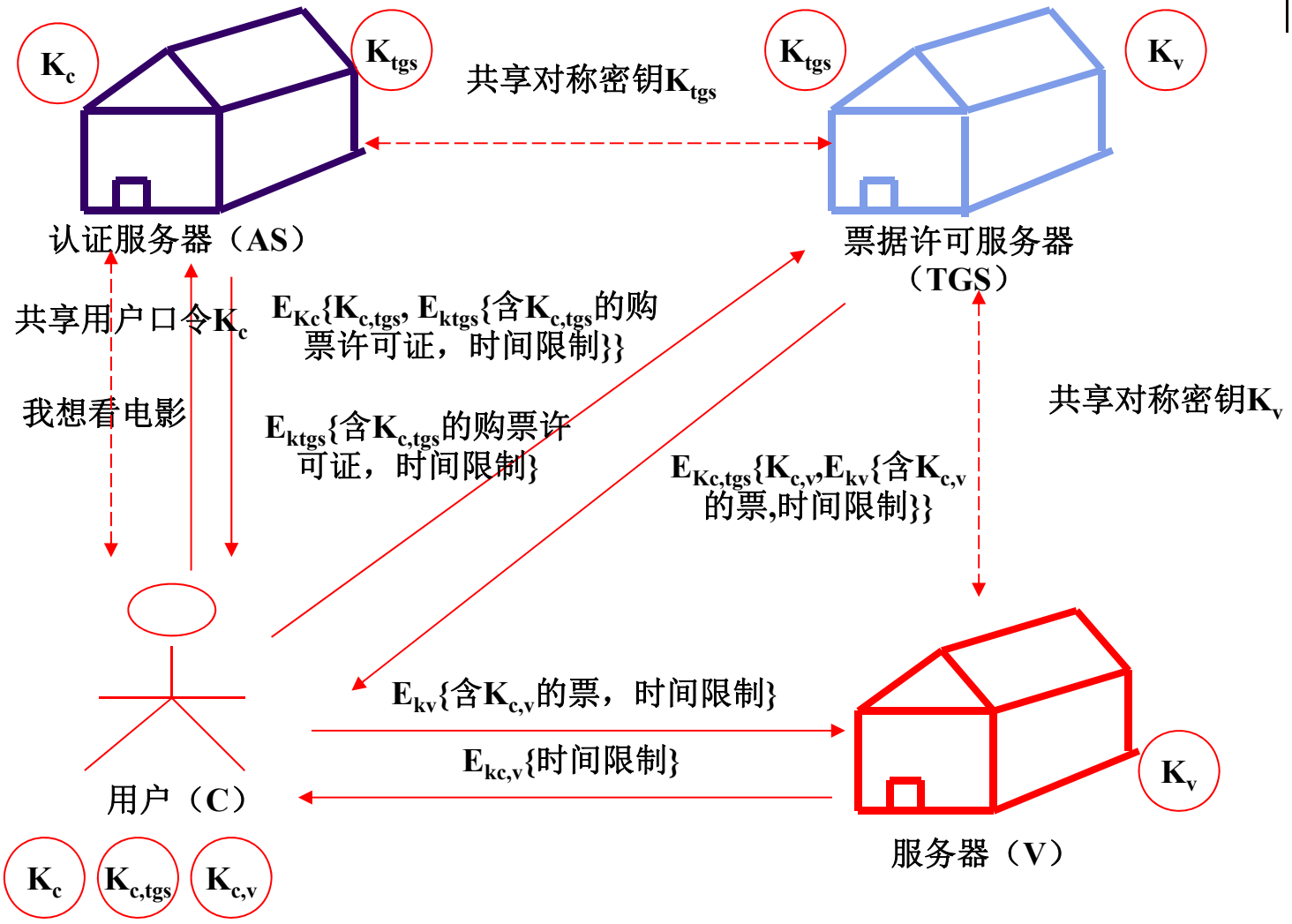
Kerberos设计思路（续）



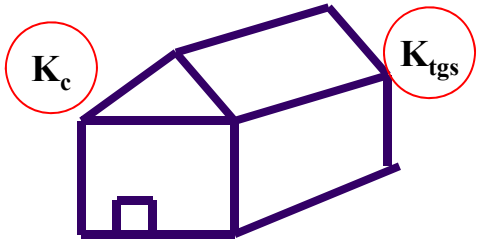
Kerberos设计思路（续）



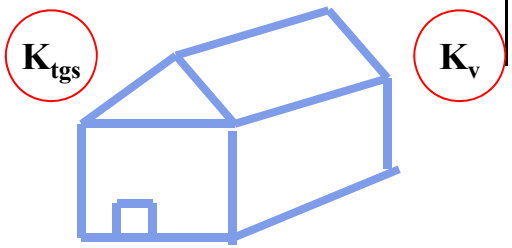
Kerberos设计思路（续）



Kerberos V4协议描述：第一阶段



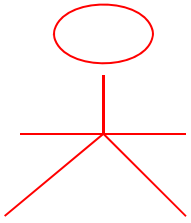
认证服务器（AS）



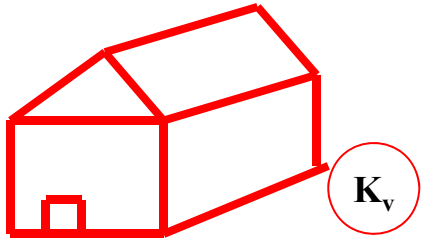
票据许可服务器（TGS）

$E_{Kc}\{K_{c,tgs}, ID_{tgs}, TS_2, LT_2, Ticket_{tgs}\}$

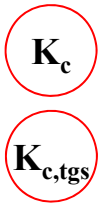
ID_C, ID_{tgs}, TS_1



用户（C）



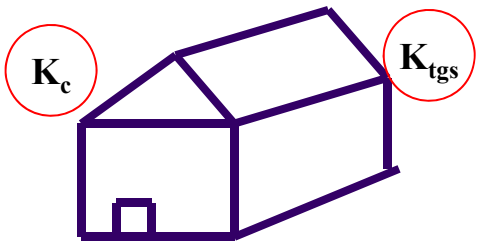
服务器（V）



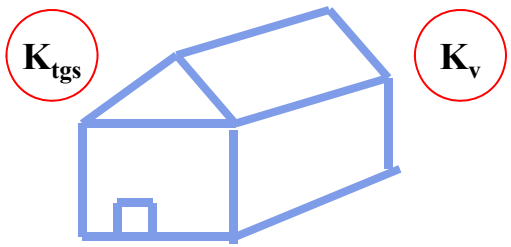
$Ticket_{tgs} = E_{Ktgs}\{K_{C,tgs}, ID_C, AD_C, ID_{tgs}, TS_2, LT_2\}$

AD_C防止何种攻击？

Kerberos V4协议描述：第二阶段



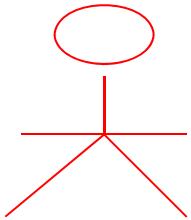
认证服务器（AS）



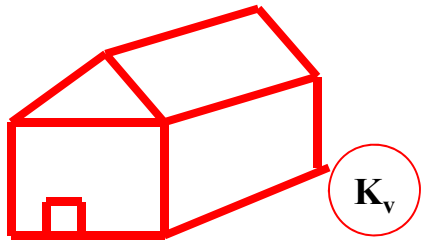
票据许可服务器（TGS）

$ID_V, Ticket_{tgs}, AU_C$

$E_{KC,tgs}\{K_{C,V}, ID_V, TS_4, Ticket_V\}$



用户（C）



服务器（V）

K_c

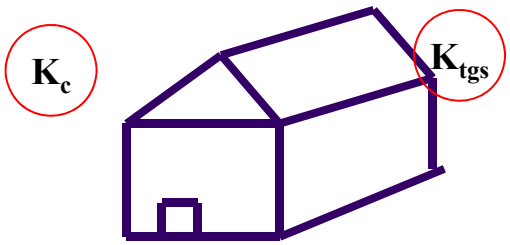
$Ticket_{tgs} = E_{Ktgs}\{K_{C,tgs}, ID_C, AD_C, ID_{tgs}, TS_2, LT_2\}$

$K_{c,tgs}$ $K_{c,v}$

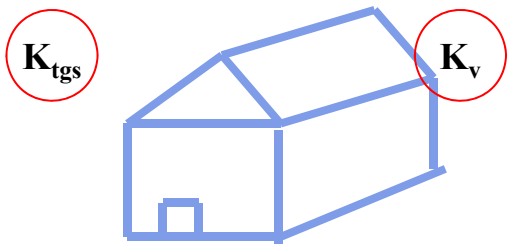
$AU_C = E_{KC,tgs}\{ID_C, AD_C, TS_3\}$

$Ticket_V = E_{KV}\{K_{C,V}, ID_C, AD_C, ID_V, TS_4, LT_4\}$

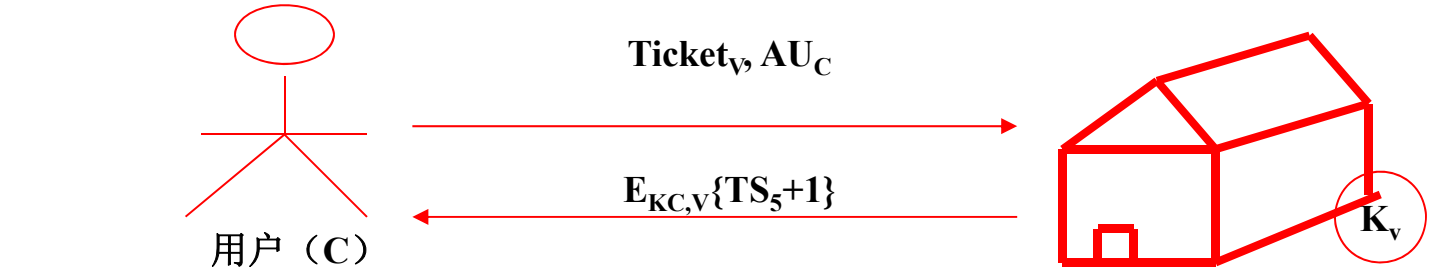
Kerberos V4协议描述：第三阶段



认证服务器（AS）



票据许可服务器（TGS）



K_c

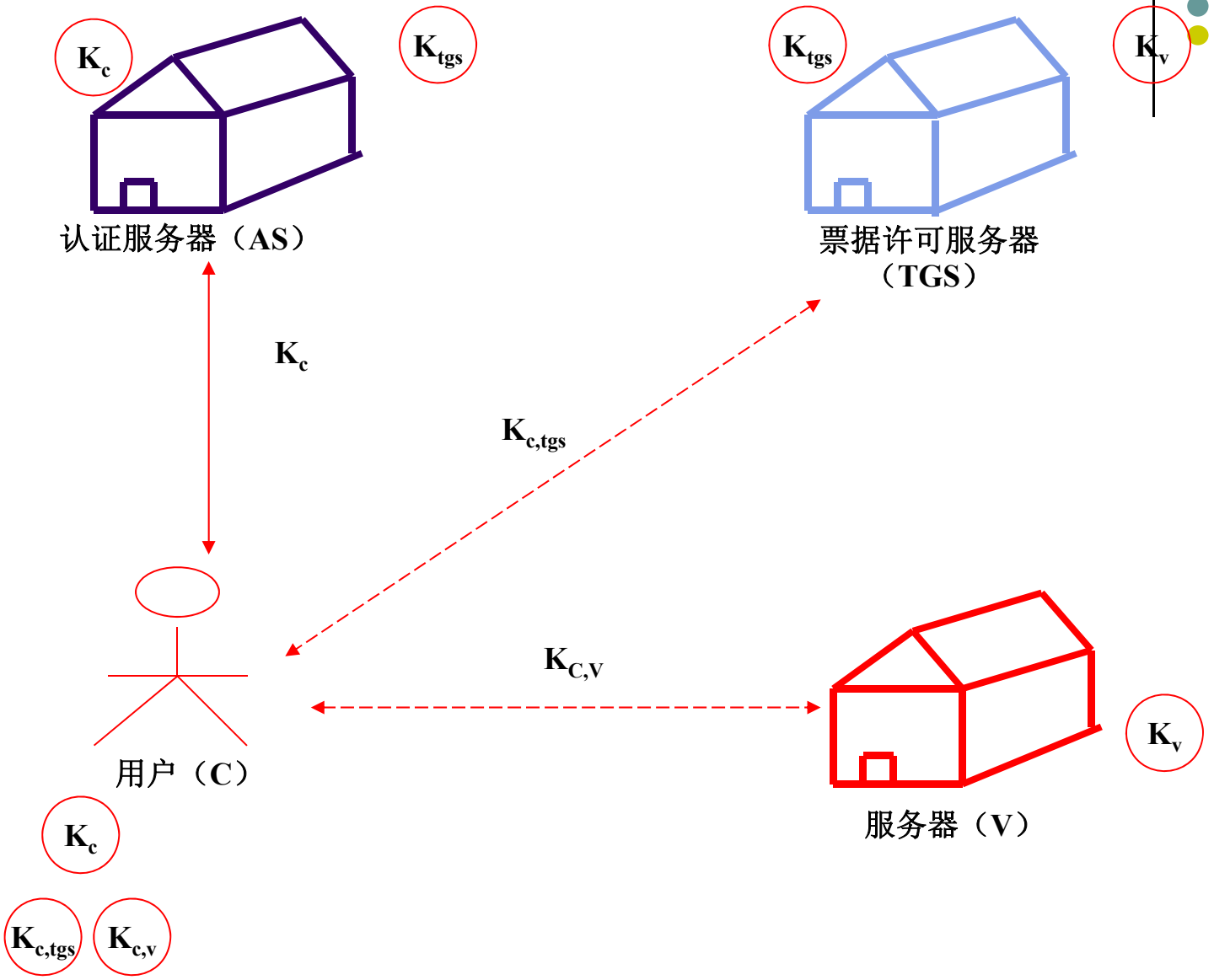
$Ticket_{tgs} = E_{Ktgs}\{K_{C,tgs}, ID_C, AD_C, ID_{tgs}, TS_2, LT_2\}$

$Ticket_V = E_{KV}\{K_{C,V}, ID_C, AD_C, ID_V, TS_4, LT_4\}$

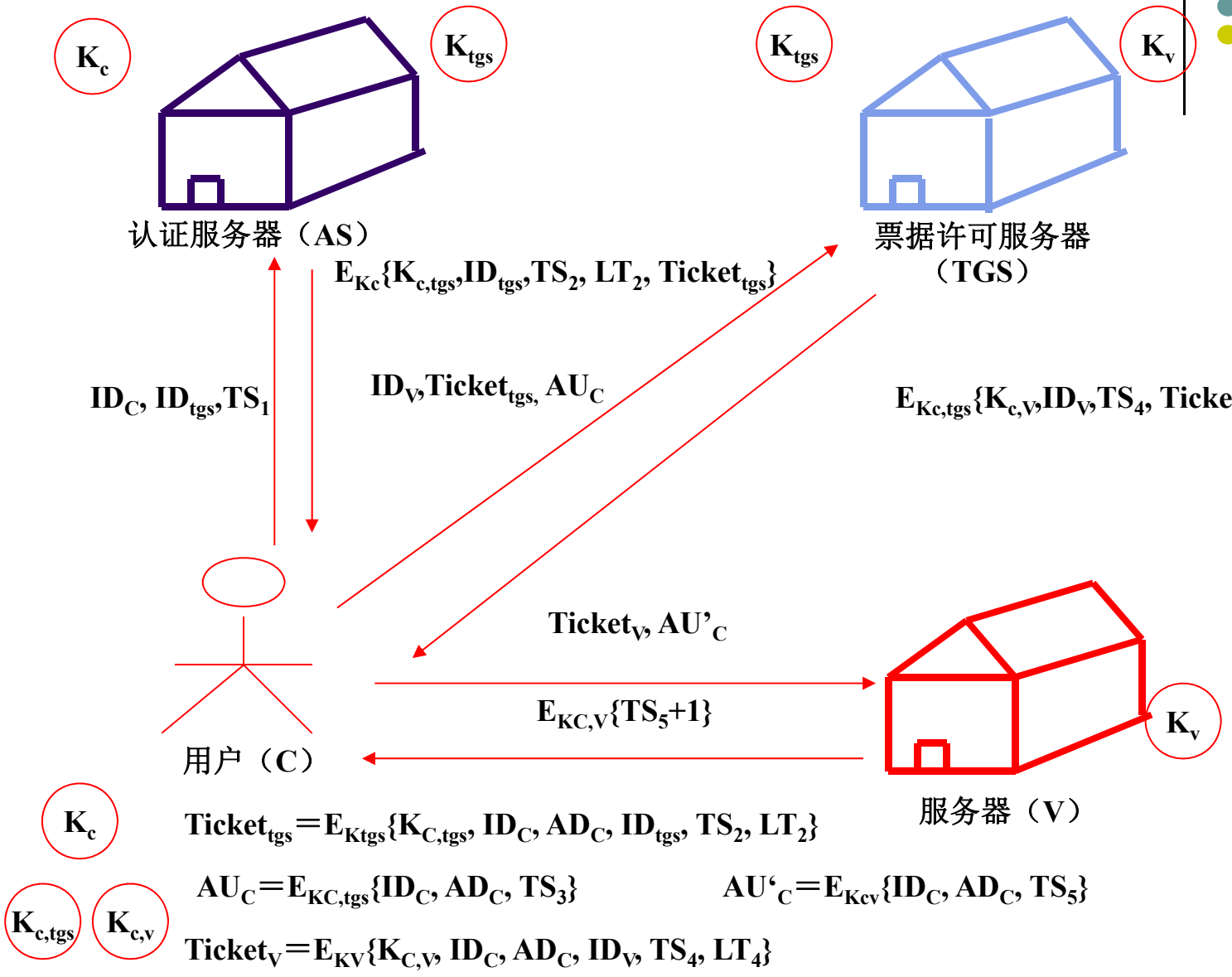
$K_{c,tgs}$ $K_{c,v}$

$AU_C = E_{Kc,v}\{ID_C, AD_C, TS_5\}$

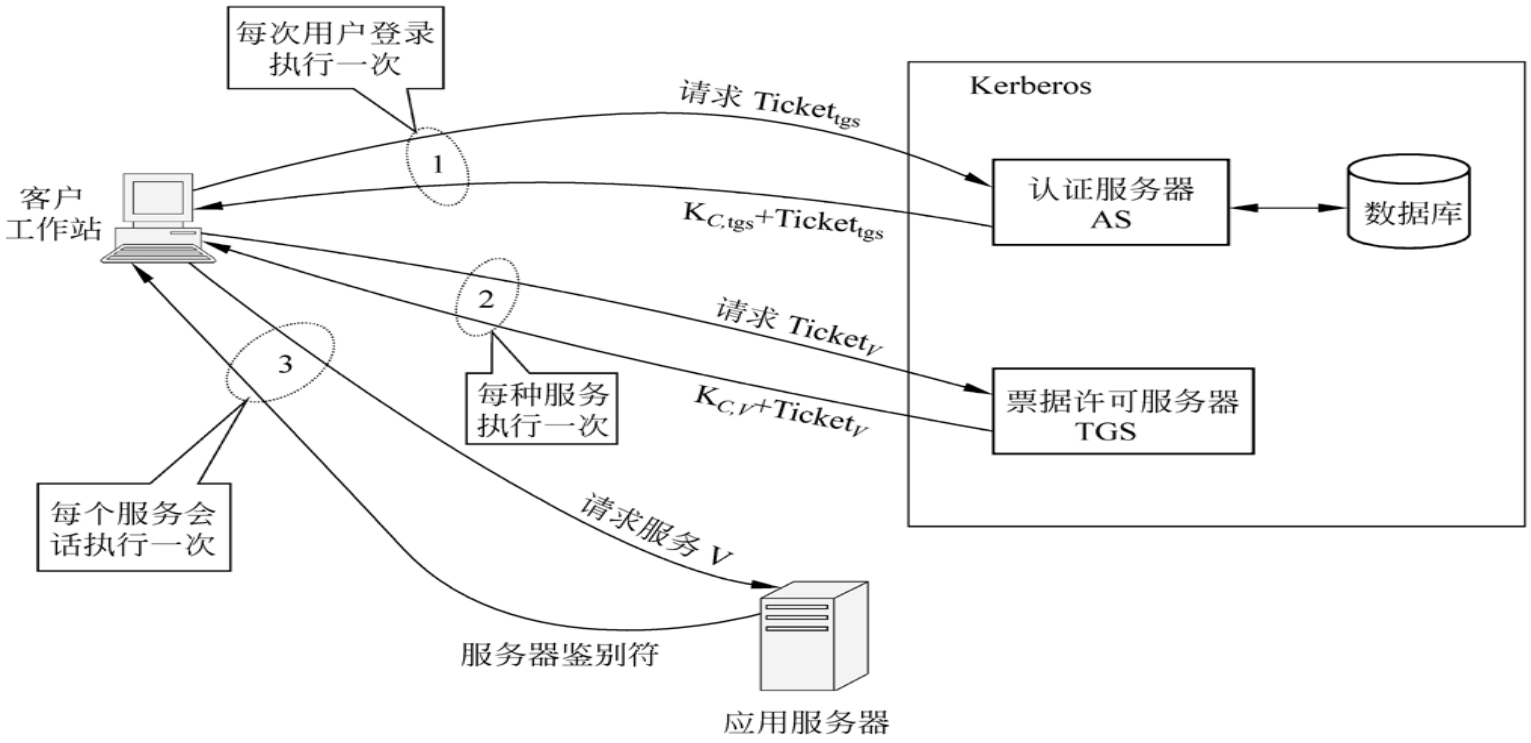
Kerberos V4协议描述：共享密钥及会话密钥



Kerberos设计思路



Kerberos（V4）协议交互过程



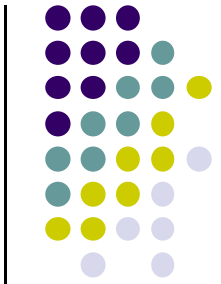
Kerberos（V4）协议的缺陷

- 依赖性
  - 加密系统的依赖性（DES）、对 IP 协议的依赖性和对时间依赖性。
- 字节顺序：没有遵循标准
- 票据有效期
  - 有效期最小为5分钟，最大约为21小时, 往往不能满足要求
- 认证转发能力
  - 不允许签发给一个用户的鉴别证书转发给其他工作站或其他客户使用

Kerberos（V4）协议的缺陷（续）

- 领域间的鉴别
  - 管理起来困难
- 加密操作缺陷
  - 非标准形式的 **DES** 加密（传播密码分组链接 **PCBC**）方式，易受攻击
- 会话密钥
  - 存在着攻击者重放会话报文进行攻击的可能
- 口令攻击
  - 未对口令提供额外的保护，攻击者有机会进行口令攻击

Kerberos（V5）协议的改进

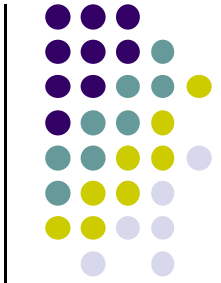


- 加密系统
  - 支持使用任何加密技术。
- 通信协议
  - **IP** 协议外，还提供了对其他协议的支持。
- 报文字节顺序
  - 采用抽象语法表示（**ASN.1**）和基本编码规则（**BER**）来进行规范。

Kerberos（V5）协议的改进（续）

- 票据的有效期
  - 允许任意大小的有效期，有效期定义为一个开始时间和结束时间。
- 鉴别转发能力
- 更有效的方法来解决领域间的认证问题
- 口令攻击
  - 提供了一种预鉴别（**preauthentication**）机制，使口令攻击更加困难。

Kerberos 领域(realm)



- 构成：一个完整的 Kerberos 环境包括一个 Kerberos 服务器，一组工作站和一组应用服务器。
- Kerberos 服务器数据库中拥有所有参与用户的 UID 和口令散列表。
- Kerberos服务器必须与每一个服务器之间共享一个保密密钥。
- 所有用户均在 Kerberos 服务器上注册。
- 所有服务器均在 Kerberos 服务器上注册。
- 领域的划分是根据网络的管理边界来划定的。

Kerberos 领域间的互通

- 跨领域的服务访问
  - 一个用户可能需要访问另一个 **Kerberos** 领域中应用服务器；
  - 一个应用服务器也可以向其他领域中的客户提供网络服务。
- 领域间互通的前提
  - 支持不同领域之间进行用户身份鉴别的机制；
  - 互通领域中的 **Kerberos** 服务器之间必须共享一个密钥；
  - 同时两个 **Kerberos** 服务器也必须进行相互注册。

远程服务访问的认证过程