开放分布网络环境的安全保障

用户通过访问服务器提供的服务

- 服务器能够：

  对请求服务的认证

  限制非授权用户访问

- 用户工作站存在的威胁：
  - 一个用户可能冒充另一个用户操作；
  - 用户可改变一个工作站的网络地址，冒充另一台工作站工作；
  - 用户可窃听他人的信息交换，并用回放（重放）攻击获得对一个服务器的访问权或中断服务器的运行

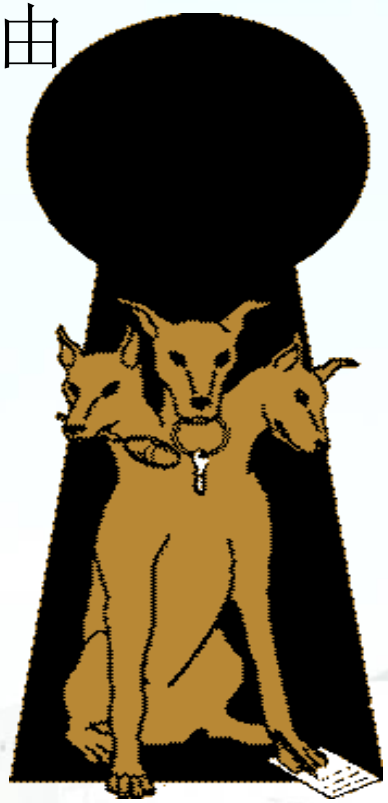
C/S环境下的安全方案

- 每一工作站保证识别其用户，并依赖于服务器强制实施一个基于用户标识的安全策略
- 客户端系统向服务器作身份认证
- 每一用户对每一服务证明其标识身份，并要求服务器向客户端证明其标识身份

Kerberos

- trusted key server system from MIT (80年代由MIT开发)

守卫冥王大门的长有三头的看门狗
(希腊神话)



- provides centralised private-key third-party authentication in a distributed network
  - allows users access to services distributed through network
- two versions in use: 4 & 5
- The Kerberos Network Authentication Service (V5)

  1993-RFC 1510, 2005-RFC 4120
- RFC4537: Kerberos Cryptosystem Negotiation Extension

  RFC 5021: Extended Kerberos Version 5 Key Distribution Center (KDC)
- Exchanges over TCP

Kerberos解决的问题

- **一个非授权用户不能够获得其无权访问的服务或数据**

- ✓ **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**

- ✓ **在一个分布式的client/server体系机构中采用一个或多个Kerberos服务器提供认证服务**

- ✓ **提供一个可信第三方认证服务**

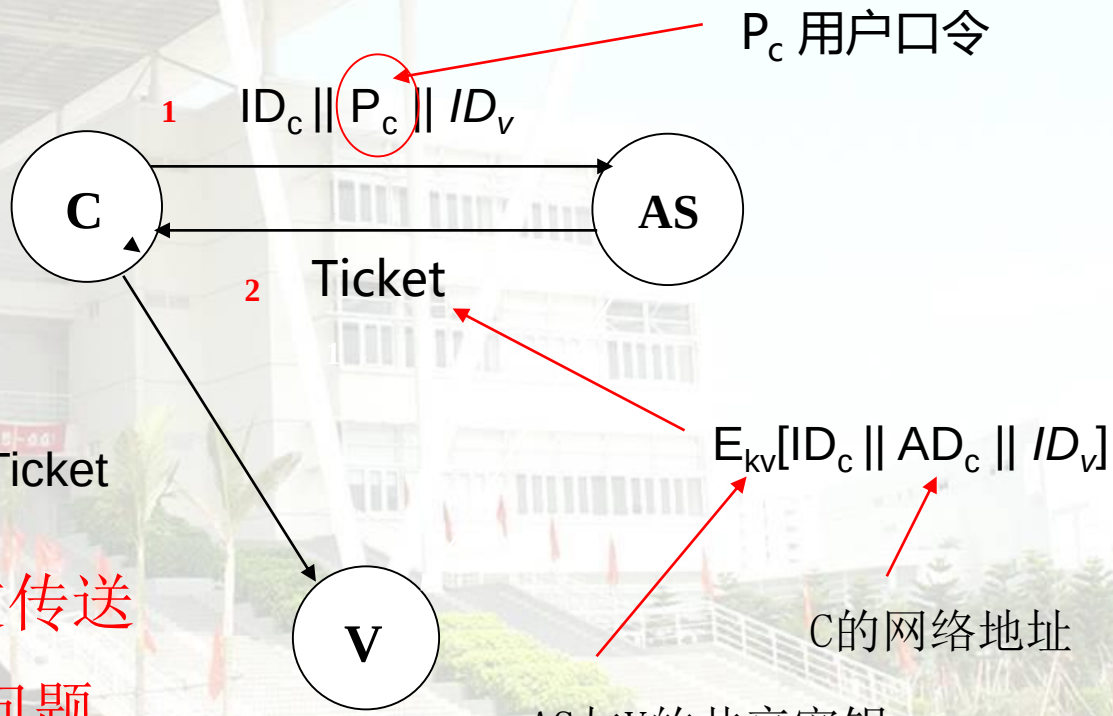
- Kerberos的原理对话

Kerberos Requirements

- **Secure 安全性**
  黑客不能获得必要信息以假冒其它用户
- **Reliable 可靠性**
  借助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
- **Transparent 透明性**
  用户除了要求输入口令以外应感觉不到认证的发生
- **Scalable 可伸缩性**
  系统应能够支持大数量的客户和服务器

Authentication Server (AS)

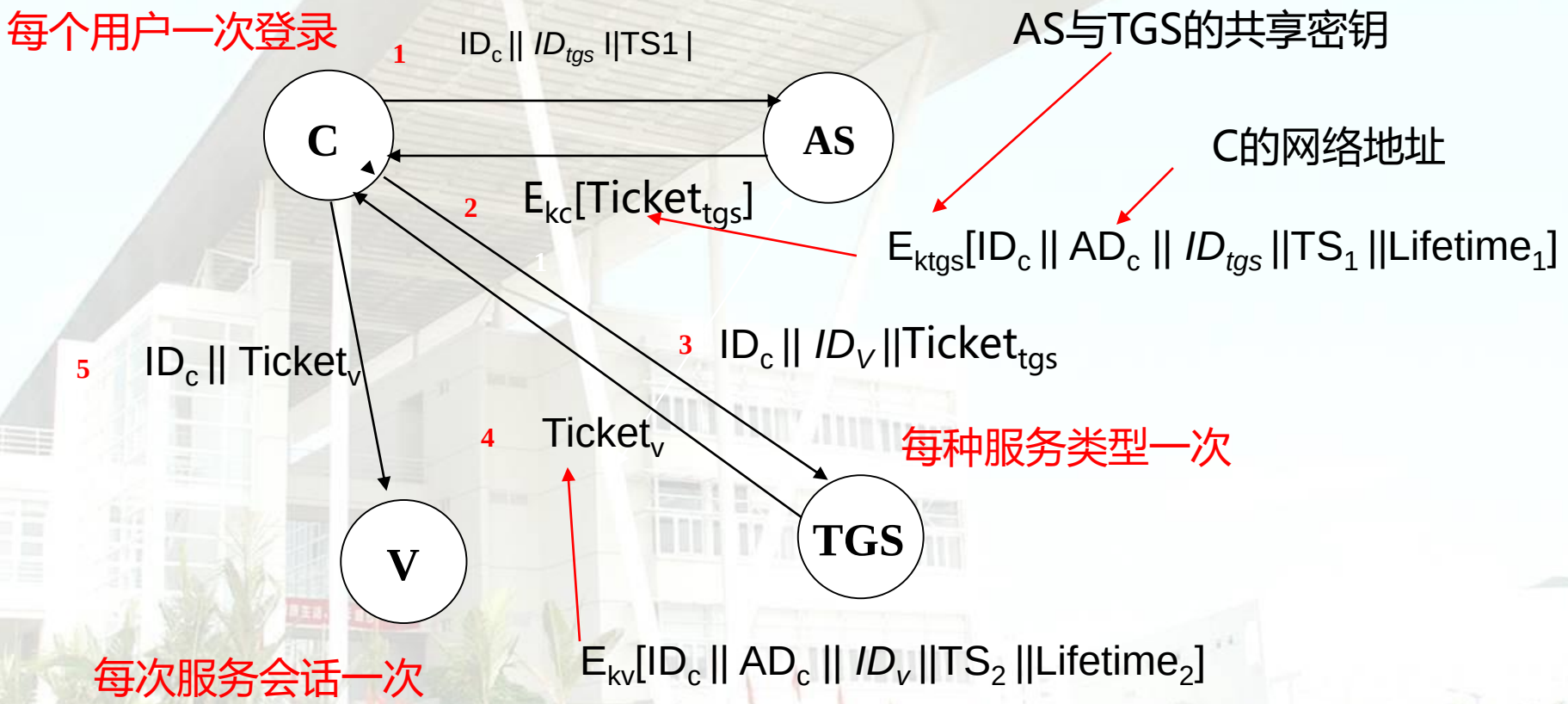
认证服务器(AS)：知道所有用户的口令
与每一个服务器共有一个唯一的保密密钥，
通过物理上或以更安全的手段分发



用户口令明文传送

Ticket重用问题

Ticket Granting Server (TGS)



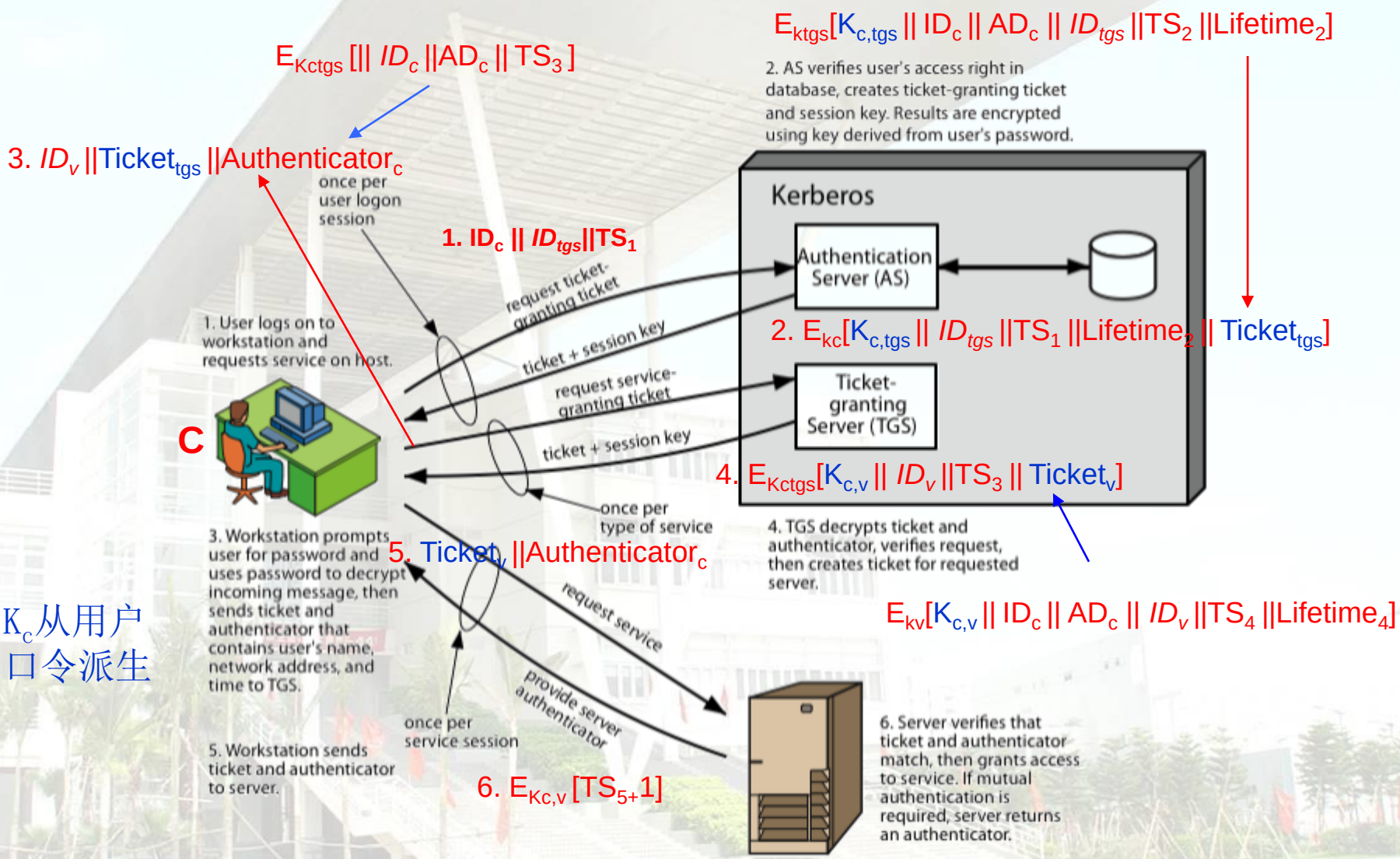
Kerberos v4 Overview

- basic third-party authentication scheme
- have an Authentication Server (AS)
  - users initially negotiate with AS to identify self
  - AS provides a non-corruptible authentication credential (ticket granting ticket -TGT)
- have a Ticket Granting server (TGS)
  - users subsequently request access to other services from TGS on basis of user's TGT

Kerberos v4 Dialogue

- obtain ticket granting ticket from AS
  once per session
- obtain service granting ticket from TGT
  for each distinct service required
- client/server exchange to obtain service
  on every service request

Kerberos 4 Overview



K_c从用户口令派生

Kerberos 4 Message Exchanges

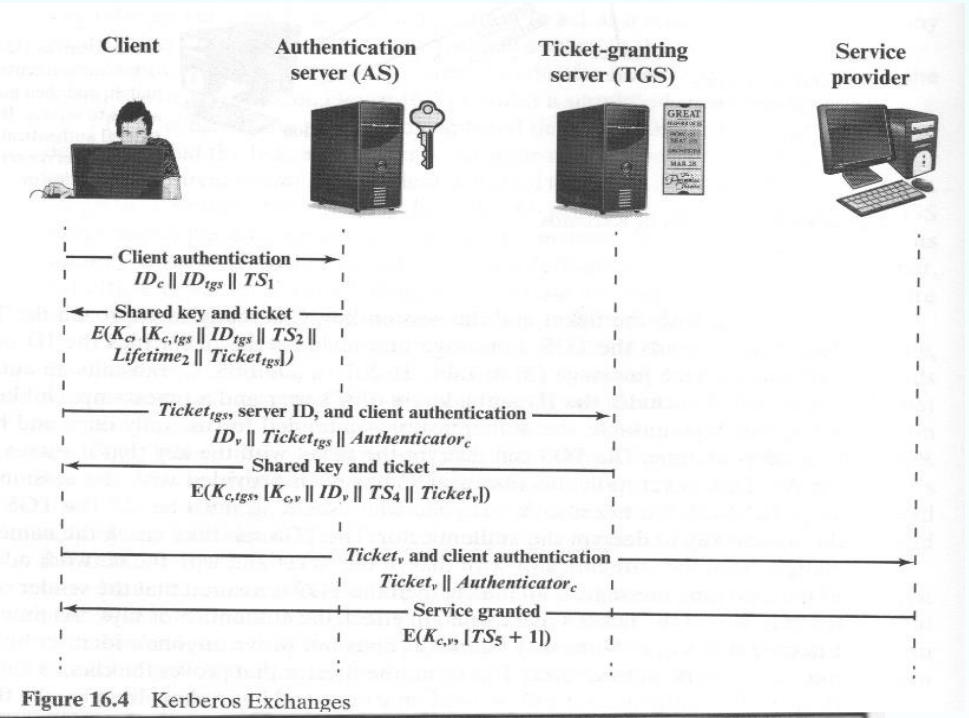


Figure 16.4 Kerberos Exchanges

(1) $C \rightarrow AS \quad ID_c \| ID_{tgs} \| TS_1$

(2) $AS \rightarrow C \quad E(K_c, [K_{c,tgs} \| ID_{tgs} \| TS_2 \| Lifetime_2 \| Ticket_{tgs}])$

$$Ticket_{tgs} = E(K_{tgs}, [K_{c,tgs} \| ID_C \| AD_C \| ID_{tgs} \| TS_2 \| Lifetime_2])$$

(a) Authentication Service Exchange to obtain ticket-granting ticket

(3) $C \rightarrow TGS \quad ID_v \| Ticket_{tgs} \| Authenticator_c$

(4) $TGS \rightarrow C \quad E(K_{c,tgs}, [K_{c,v} \| ID_v \| TS_4 \| Ticket_v])$

$$Ticket_{tgs} = E(K_{tgs}, [K_{c,tgs} \| ID_C \| AD_C \| ID_{tgs} \| TS_2 \| Lifetime_2])$$

$$Ticket_v = E(K_v, [K_{c,v} \| ID_C \| AD_C \| ID_v \| TS_4 \| Lifetime_4])$$

$$Authenticator_c = E(K_{c,tgs}, [ID_C \| AD_C \| TS_3])$$

(b) Ticket-Granting Service Exchange to obtain service-granting ticket

(5) $C \rightarrow V \quad Ticket_v \| Authenticator_c$

(6) $V \rightarrow C \quad E(K_{c,v}, [TS_5 + 1])$ (for mutual authentication)

$$Ticket_v = E(K_v, [K_{c,v} \| ID_C \| AD_C \| ID_v \| TS_4 \| Lifetime_4])$$

$$Authenticator_c = E(K_{c,v}, [ID_C \| AD_C \| TS_5])$$

(c) Client/Server Authentication Exchange to obtain service

Kerberos Realms

a Kerberos environment consists of:

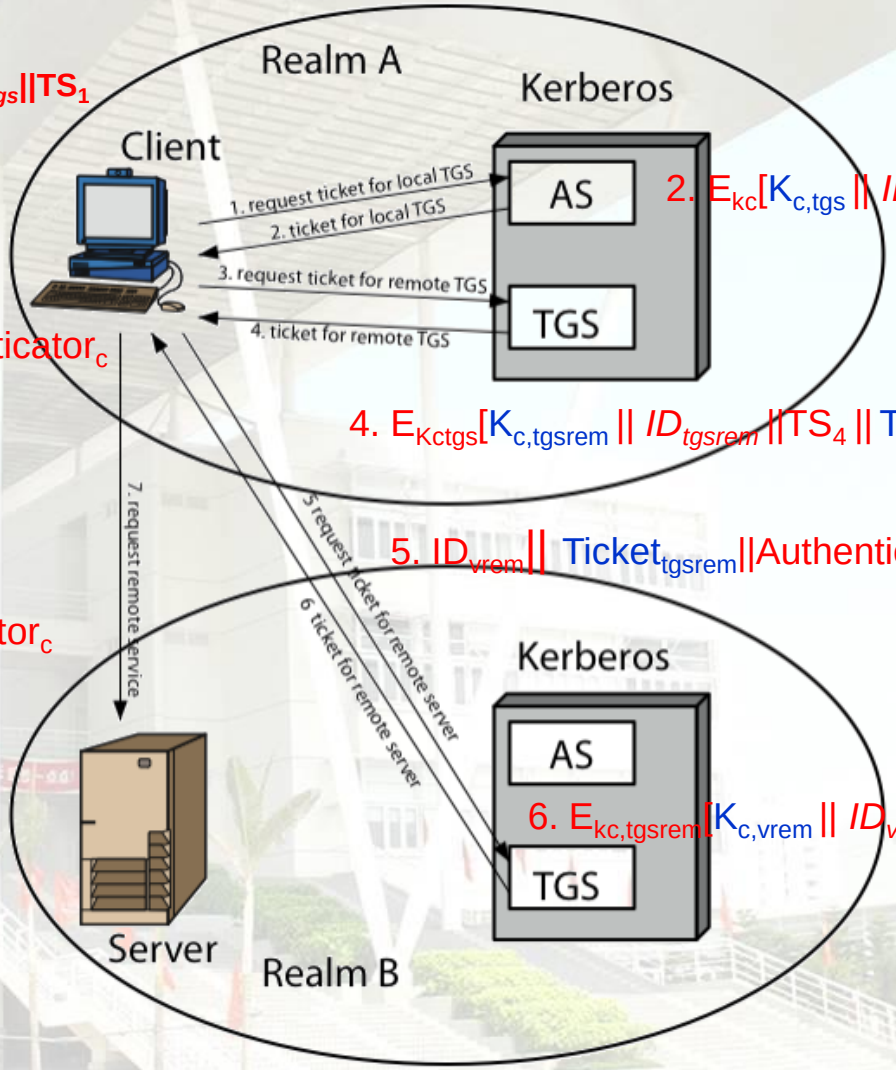
- a Kerberos server
- a number of clients, all registered with server:

  数据库:用户标识ID(UID), 口令（散列表）

  所有用户均在Kerberos服务器上注册
- application servers, sharing keys with server

  共享一个保密密钥

Kerberos Realms



1. $ID_c \| ID_{tgs} \| TS_1$

2. $E_{kc}[K_{c,tgs} \| ID_{tgs} \| TS_2 \| Lifetime_2 \| Ticket_{tgs}]$

3. $ID_{tgsrem} \| Ticket_{tgs} \| Authenticator_c$

4. $E_{Kctgs}[K_{c,tgsrem} \| ID_{tgsrem} \| TS_4 \| Ticket_{tgsrem}]$

5. $ID_{vrem} \| Ticket_{tgsrem} \| Authenticator_c$

6. $E_{kc,tgsrem}[K_{c,vrem} \| ID_{vrem} \| TS_6 \| Ticket_{vrem}]$

7. $Ticket_{vrem} \| Authenticator_c$

获得另一域的认证服务

（1）获得本地TGS的访问权；

（2）请求一张远程TGS的票据；

（3）向远程TGS申请其领域内的服务许可票据

Kerberos Version 5

- developed in mid 1990's
- specified as Internet standard RFC-1510(1993), RFC-4120(2005)
- RFC4537: Kerberos Cryptosystem Negotiation Extension

  RFC 5021: Extended Kerberos Version 5 Key Distribution Center (KDC) Exchanges over TCP
- provides improvements over v4
  - ✓ 加密系统依赖性: V4:DES, V5多种算法
  - ✓ Internet协议依赖性: V4:IP 地址, V5任何类型
  - ✓ 消息字节次序: V5: ASN.1, BER编码
  - ✓ Ticket的时效性: V5任何长度的生命期
  - ✓ V5 有很灵活的认证机制: 包括Authentication forwarding:Inter-realm authentication
  - ✓ V5对密钥使用及管理很完善

  V4,V5均基于**口令->密钥**, 有受攻击的隐患,需改进 1,2

V5 VS V4

Table 14.3 Summary of Kerberos Version 5 Message Exchanges

(a) Authentication Service Exchange: to obtain ticket-granting ticket
(1) C → AS: $\text{Options} \| ID_c \| Realm_c \| ID_{tgs} \| Times \| Nonce_1$
(2) AS → C: $Realm_c \| ID_C \| Ticket_{tgs} \| E_{K_c}\left[K_{c,tgs} \| Times \| Nonce_1 \| Realm_{tgs} \| ID_{tgs}\right]$
$Ticket_{tgs} = E_{K_{tgs}}\left[Flags \| K_{c,tgs} \| Realm_c \| ID_C \| AD_C \| Times\right]$
(b) Ticket-Granting Service Exchange: to obtain service-granting ticket
(3) C → TGS: $\text{Options} \| ID_v \| Times \| \| Nonce_2 \| Ticket_{tgs} \| Authenticator_c$
(4) TGS → C: $Realm_c \| ID_C \| Ticket_v \| E_{K_{c,tgs}}\left[K_{c,v} \| Times \| Nonce_2 \| Realm_v \| ID_V\right]$
$Ticket_{tgs} = E_{K_{tgs}}\left[Flags \| K_{c,tgs} \| Realm_c \| ID_C \| AD_C \| Times\right]$
$Ticket_v = E_{K_v}\left[Flags \| K_{c,v} \| Realm_c \| ID_C \| AD_C \| Times\right]$
$Authenticator_c = E_{K_{c,tgs}}\left[ID_C \| Realm_c \| TS_1\right]$
(c) Client/Server Authentication Exchange: to obtain service
(5) C → V: $\text{Options} \| \text{Ticket}_v \| \text{Authenticator}_c$
(6) V → C: $E_{K_{C,V}}[\ TS_2 \| \text{Subkey} \| \text{Seq\#}\ ]$
$Ticket_v = E_{K_v}\left[Flags \| K_{c,v} \| Realm_c \| ID_C \| AD_C \| Times\right]$
$Authenticator_c = E_{K_{c,v}}\left[ID_C \| Realm_c \| TS_2 \| Subkey \| Seq\#\right]$

Table 14.1 Summary of Kerberos Version 4 Message Exchanges

(a) Authentication Service Exchange: to obtain ticket-granting ticket
(1) C → AS: $ID_c \| ID_{tgs} \| TS_1$
(2) AS → C: $E_{K_c}\left[K_{c,tgs} \| ID_{tgs} \| TS_2 \| Lifetime_2 \| Ticket_{tgs}\right]$
$Ticket_{tgs} = E_{K_{tgs}}\left[K_{c,tgs} \| ID_C \| AD_C \| ID_{tgs} \| TS_2 \| Lifetime_2\right]$
(b) Ticket-Granting Service Exchange: to obtain service-granting ticket
(3) C → TGS: $ID_v \| Ticket_{tgs} \| Authenticator_c$
(4) TGS → C: $E_{K_{c,tgs}}\left[K_{c,v} \| ID_v \| TS_4 \| Ticket_v\right]$
$Ticket_{tgs} = E_{K_{tgs}}\left[K_{c,tgs} \| ID_C \| AD_C \| ID_{tgs} \| TS_2 \| Lifetime_2\right]$
$Ticket_v = E_{K_v}\left[K_{c,v} \| ID_C \| AD_C \| ID_v \| TS_4 \| Lifetime_4\right]$
$Authenticator_c = E_{K_{tgs}}\left[ID_C \| AD_C \| TS_3\right]$
(c) Client/Server Authentication Exchange: to obtain service
(5) C → V: $Ticket_v \| Authenticator_c$
(6) V → C: $E_{K_{c,v}}\left[TS_5 + 1\right]$ (for mutual authentication)
$Ticket_v = E_{K_v}\left[K_{c,v} \| ID_C \| AD_C \| ID_v \| TS_4 \| Lifetime_4\right]$
$Authenticator_c = E_{K_{c,v}}\left[ID_C \| AD_C \| TS_5\right]$