Kerberos

کربروس

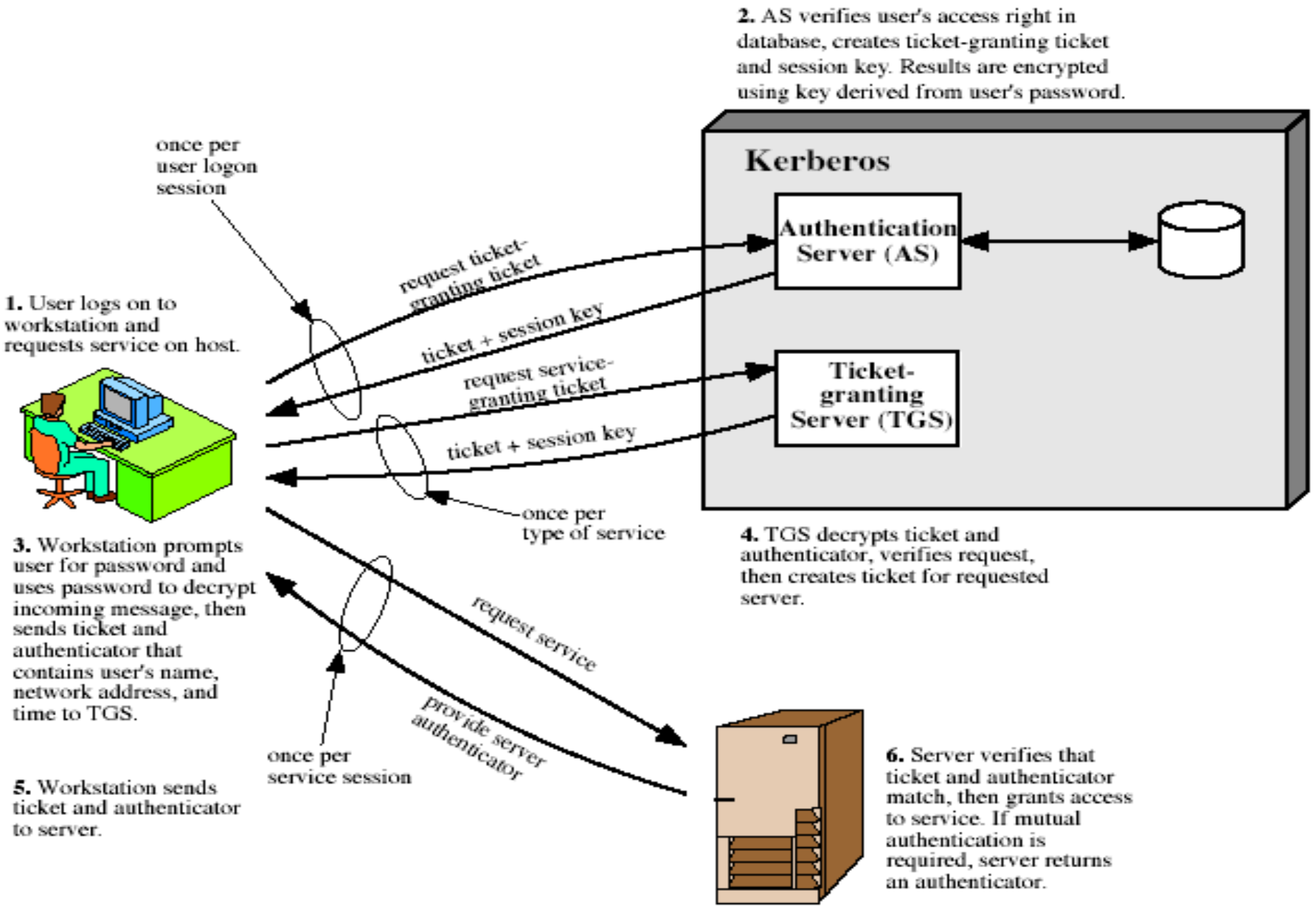
- پروتکل احراز هویت بر اساس رمز نگاری کلید متقارن
- طراحی شده در MIT
- به جای احراز هویت در هر کارگزار به صورت توزیع شده، یک کارگزار خاص را به احراز هویت اختصاص میدهیم
- نسخه های ٤ و ٥ آن در حال استفاده هستند
- احراز هویت دو جانبه (mutual) برقرار میشود.
  - کارگزاران و کارفرمایان هردو از هویت طرف مقابل اطمینان حاصل میکنند

ویژگیهای عمومی کربروس

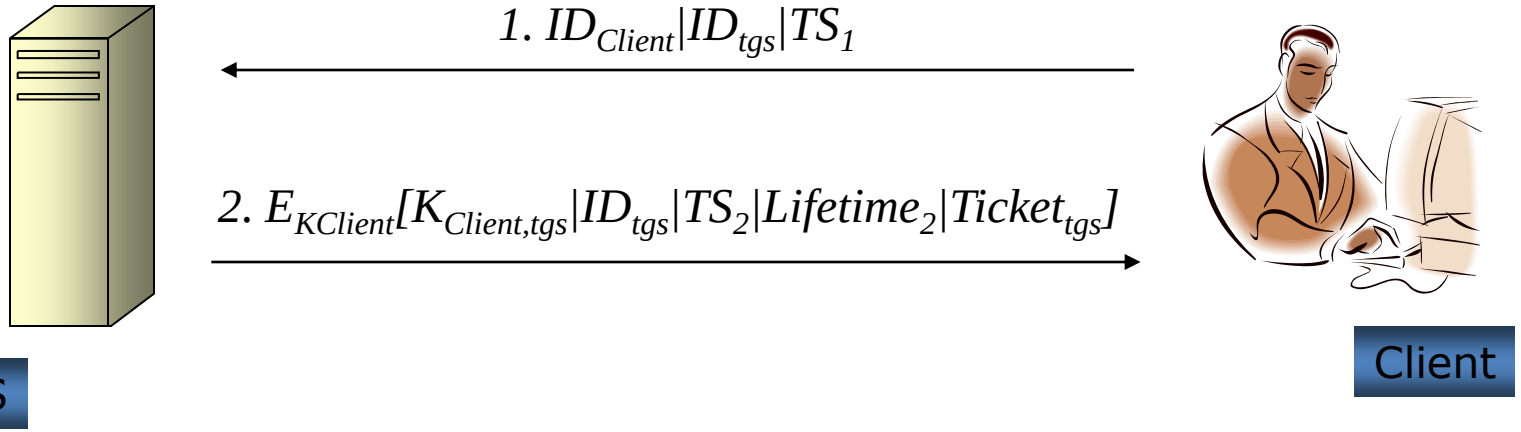
چند تعریف

- دامنه: یک محدوده دسترسی را مشخص می کند. به نوعی معادل دامنه های تعریف شده در ویندوز می باشد.
- مرکز توزیع کلید: معادل کارگزار کربروس می باشد.
- Principal : به سرویس ها، دستگاه ها، کاربران و کلیه عناصری که احتیاج به شناساندن خود به کارگزار کربروس دارند، گفته می شود.
- بلیط: در واقع نوعی گواهی است که هنگام ورود کاربر به قلمرو کربروس به او داده می شود که بیانگر اعتبار او برای دسترسی به منابع شبکه می باشد.

شمای کلی:کربروس نسخه ٤

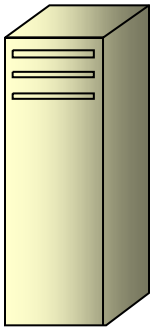


کربروس نسخه ٤: بررسی الگوریتم-١



$$Ticket_{tgs}=E_{Ktgs}[K_{Client,tgs}|ID_{Client}|Addr_{Client}|ID_{tgs}|TS_2|Lifetime_2]$$

بدست آوردن بلیط "اعطاء خدمات"



Tgs-Server

۳. $ID_{server} | Ticket_{tgs} | Authenticator_{Client}$

٤. $E_{K_{Client,tgs}} [K_{Client,server} | ID_{server} | TS_4 | Ticket_{server}]$

Client

$Ticket_{Server} =$

$E_{Kserver}[K_{Client,server} | ID_{Client} | Addr_{Client} | ID_{server} | TS_4 | Lifetime_4]$

$Authenticator_{Client} =$

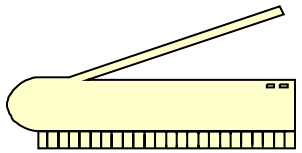
$E_{KClient,tgs}[ID_{Client} | Addr_{Client} | TS_3]$

دستیابی به خدمات سرور

Client

$5.\ Ticket_{Server} | Authenticator_{Client}$

$6.\ E_{K_{Client,Server}}\ [TS_5+1]$



Server

شمای کلی:کربروس نسخه ٤

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

(1) C → AS: $ID_c \| ID_{tgs} \| TS_1$

(2) AS → C: $\mathrm{E}_{K_c}\left[K_{c,tgs} \| ID_{tgs} \| TS_2 \| Lifetime_2 \| Ticket_{tgs}\right]$

$$Ticket_{tgs} = \mathrm{E}_{K_{tgs}}\left[K_{c,tgs} \| ID_C \| AD_C \| ID_{tgs} \| TS_2 \| Lifetime_2\right]$$

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

(3) C → TGS: $ID_v \| Ticket_{tgs} \| Authenticator_c$

(4) TGS → C: $\mathrm{E}_{K_{c,tgs}}\left[K_{c,v} \| ID_v \| TS_4 \| Ticket_v\right]$

$$Ticket_{tgs} = \mathrm{E}_{K_{tgs}}\left[K_{c,tgs} \| ID_C \| AD_C \| ID_{tgs} \| TS_2 \| Lifetime_2\right]$$

$$Ticket_v = \mathrm{E}_{K_v}\left[K_{c,v} \| ID_C \| AD_C \| ID_v \| TS_4 \| Lifetime_4\right]$$

$$Authenticator_c = \mathrm{E}_{K_{tgs}}\left[ID_C \| AD_C \| TS_3\right]$$

(c) Client/Server Authentication Exchange: to obtain service

(5) C → V: $Ticket_v \| Authenticator_c$

(6) V → C: $\mathrm{E}_{K_{c,v}}\left[TS_5 + 1\right]$ (for mutual authentication)

$$Ticket_v = \mathrm{E}_{K_v}\left[K_{c,v} \| ID_C \| AD_C \| ID_v \| TS_4 \| Lifetime_4\right]$$

$$Authenticator_c = \mathrm{E}_{K_{c,v}}\left[ID_C \| AD_C \| TS_5\right]$$

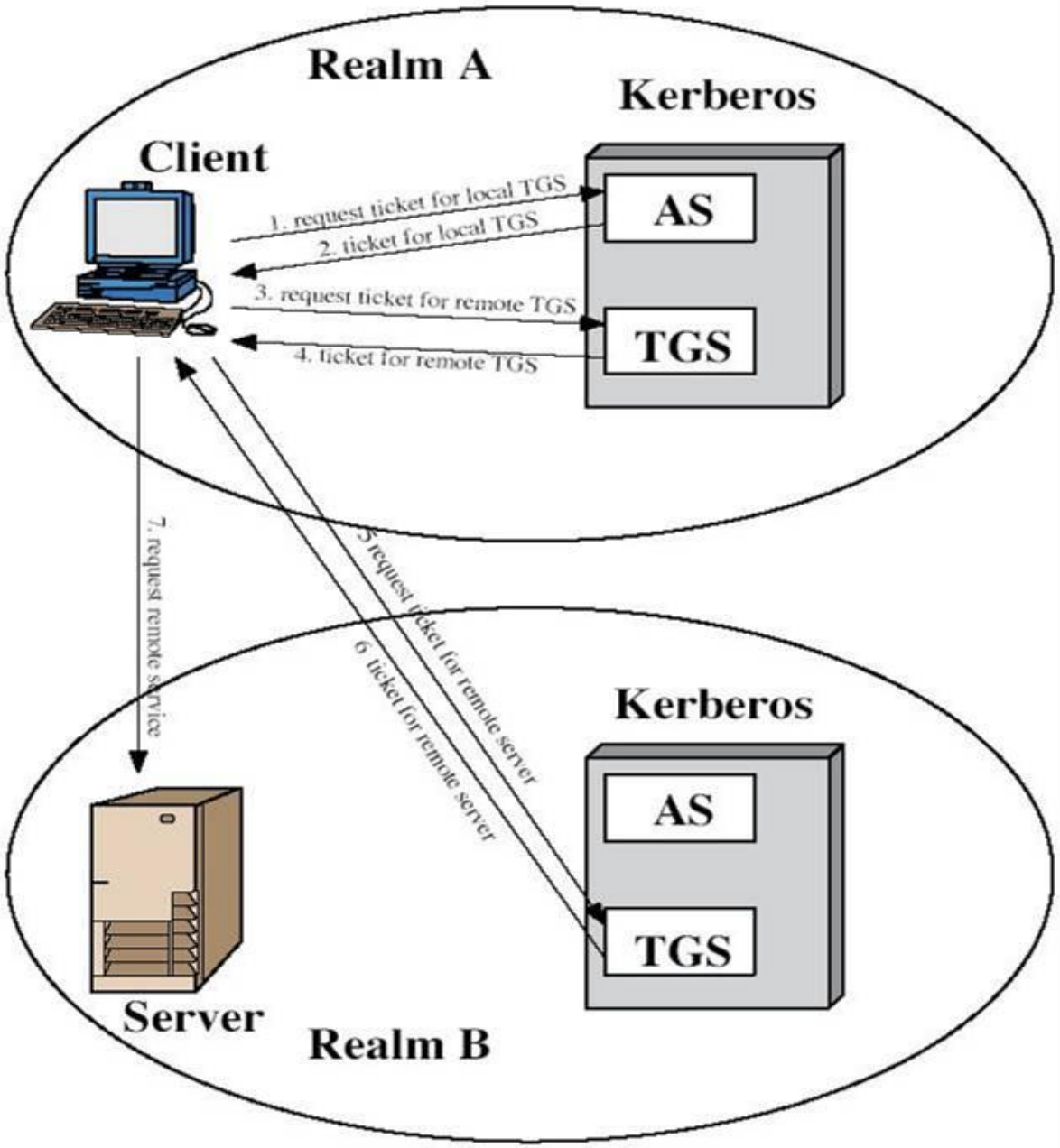
قلمرو کربروس

- قلمرو کربروس از بخشهای زیر تشکیل شده است:
  - کارگزار کربروس
  - کارفرمایان
  - کارگزاران کاربردی Application Servers
- کارگزار کربروس گذرواژه تمام کاربران را در پایگاه داده خود دارد.
- معمولاً هر قلمرو معادل یک حوزه مدیریتی می باشد.

هویت شناسی بین قلمرویی(InterRealm)

- امکان اینکه کاربران بتوانند از خدمات موجود در قلمروهای دیگر استفاده کنند.
  - کارگزاران کربروس هر قلمرو یک کلید مخفی با کارگزاران کربروس قلمرو همکار مقابل به اشتراک میگذارند.
  - وجود ***N*** قلمرو همکار نیازمند ***N(N-1)/2*** کلید مخفی است.
  - دو کارگزار کربروس همدیگر را ثبت نام مینمایند.

هویت شناسی بین قلمرویی



کربروس نسخه ۵

- مشخصات
  - در اواسط ۱۹۹۰ مطرح شد
  - نقص ها و کمبودهای نسخه قبلی را برطرف کرده است
  - به عنوان استاندارد اینترنتی ***RFC 1510*** در نظر گرفته شده است.
  - ویندوز ۲۰۰۰ از استاندارد اینترنتی کربروس نسخه ۵ بعنوان روش اصلی هویت شناسی کاربران استفاده می کند.

مشکلات Kerberos v4 و نحوه رفع آنها در نسخه ۵

- وابستگی به یک سیستم رمزنگاری خاص(DES)

  + در نسخه ۵ می توان از هر الگوریتم متقارن استفاده کرد

- وابستگی به IP

  + در نسخه ۵ می توان از هر نوع آدرس شبکه ای استفاده کرد

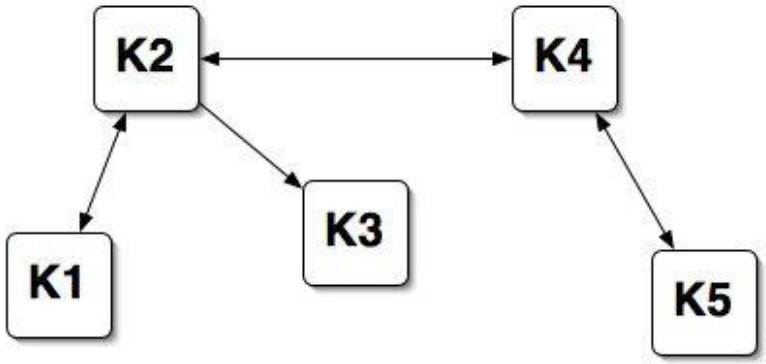
  – امکان استفاده از اعتبار کاربر متفاوت در دسترسی به یک سرور خاص

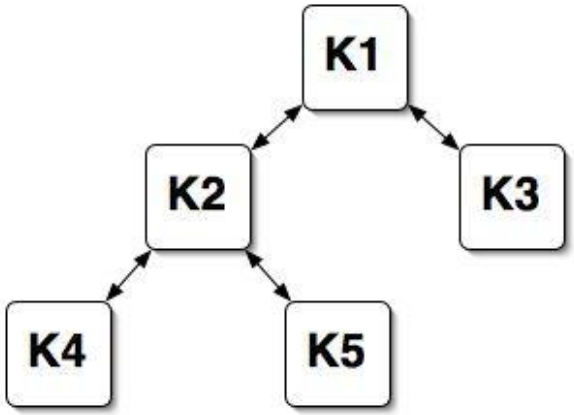
  + در نسخه ۵ اجازه داده می شود که سرویس از حساب کاربر متفاوت از کاربر login کرده انجام شود.

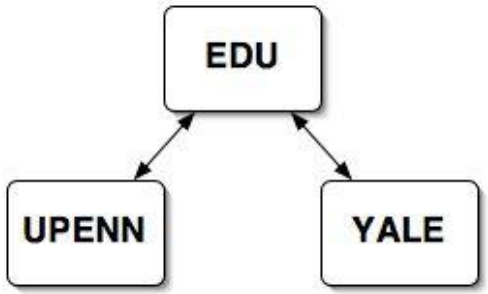
  – با افزایش تعداد قلمروها، تعداد کلیدها بصورت تصاعدی افزایش می یابد

  + در نسخه ۵ با استفاده از معماری سلسه مراتبی این مشکل حل شده است.

Hierarchy/Chain of Realms







شمای کلی:کربروس نسخه ۵

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

(1) $C \rightarrow AS$: $Options \| ID_c \| Realm_c \| ID_{tgs} \| Times \| Nonce_1$

(2) $AS \rightarrow C$: $Realm_c \| ID_c \| Ticket_{tgs} \| E_{Kc}[K_{c,tgs} \| Times \| Nonce_1 \| Realm_{tgs} \| ID_{tgs}]$

$Ticket_{tgs} = E_{K_{tgs}}[Flags \| K_{c,tgs} \| Realm_c \| ID_c \| AD_c \| Times]$

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

(3) $C \rightarrow TGS$: $Options \| ID_v \| Times \| \| Nonce_2 \| Ticket_{tgs} \| Authenticator_c$

(4) $TGS \rightarrow C$: $Realm_c \| ID_c \| Ticket_v \| E_{K_{c,tgs}}[K_{c,v} \| Times \| Nonce_2 \| Realm_v \| ID_v]$

$Ticket_{tgs} = E_{K_{tgs}}[Flags \| K_{c,tgs} \| Realm_c \| ID_c \| AD_c \| Times]$

$Ticket_v = E_{K_v}[Flags \| K_{c,v} \| Realm_c \| ID_c \| AD_c \| Times]$

$Authenticator_c = E_{K_{c,tgs}}[ID_c \| Realm_c \| TS_1]$

(c) Client/Server Authentication Exchange: to obtain service

(5) $C \rightarrow TGS$: $Options \| Ticket_v \| Authenticator_c$

(6) $TGS \rightarrow C$: $E_{K_{c,v}}[TS_2 \| Subkey \| Seq\#]$

$Ticket_v = E_{K_v}[Flags \| K_{c,v} \| Realm_c \| ID_c \| AD_c \| Times]$

$Authenticator_c = E_{K_{c,V}}[ID_c \| Realm_c \| TS_2 \| Subkey \| Seq\#]$