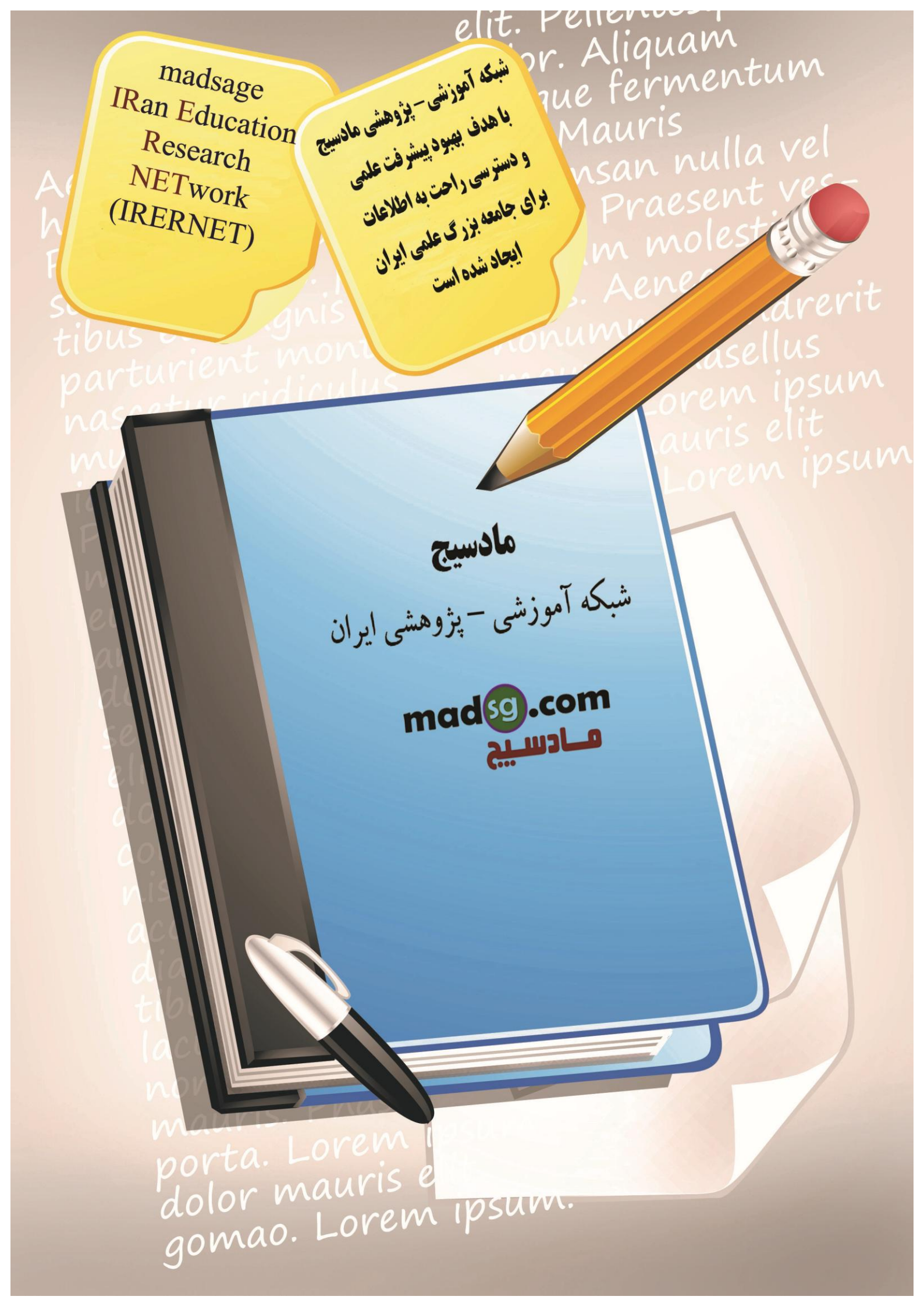

madsage
IRan Education
Research
NETwork
(IRERNET)
شبکه آموزشی-پژوهشی مادسیج
با هدف بهبود پیشرفت علمی
و دسترسی راحت به اطلاعات
برای جامعه بزرگ علمی ایران
ایجاد شده است
مادسیج
شبکه آموزشی - پژوهشی ایران
madsg.com
مادسیج

یا ذاالامن والامان

امنیت داده و شبکه

پروتکل کربروس

مرتضی امینی - نیمسال اول ۹۱-۹۰

مرکز امنیت داده و شبکه شریف
http://dnsl.ce.sharif.edu

فهرست

- مقدمه و نیازمندی‌ها
- دیالوگ‌های ساده احراز اصالت
- کربروس نسخه ۴
- کربروس نسخه ۵

افسانه یونانی

- سگ سه سر افسانه یونانی: محافظ دروازه‌های جهنم!
- سرها نماد:
  - احراز اصالت (Authentication)
  - مجازشماری (Authorization)
  - حسابرسی (Accounting)
- اگرچه در عمل تنها احراز اصالت اِعمال شد.

انگیزه

- محیط‌های جدید: به صورت توزیع‌شده
- در یک محیط توزیع‌شده سه روش برای امنیت:
  - اعتماد به ایستگاه کاری در معرفی کردن کاربران خود و اعمال سیاست امنیتی مبتنی بر شناسه کاربران در کارگزار
  - نیاز به احراز اصالت ایستگاه‌های کاری توسط کارگزار
    ولی اعتماد به ایستگاه‌های کاری نسبت به احراز اصالت کاربران خود
  - نیاز به احراز اصالت هر یک از کاربران توسط کارگزار سرویس درخواستی و احراز اصالت کارگزار سرویس توسط کاربر متقاضی

کربروس

- احراز اصالت بر اساس رمزنگاری کلید مخفی (متقارن)
- طراحی شده در MIT
- به جای احراز اصالت در هر کارگزار به صورت توزیع شده، یک کارگزار خاص را به احراز اصالت اختصاص می‌دهیم.
- نسخه‌های ۴ و ۵ آن در حال استفاده هستند.

نیازمندیها/ویژگیهای عمومی کربروس

- امنیت (Security)
  - با شنود در شبکه، امکان جعل کاربر توسط مهاجم وجود نداشته باشد.
- اطمینان (Reliability)
  - اطمینان از دسترس‌پذیری کارگزار احراز اصالت کربروس با پشتیبانی از خدمت‌رسانی توزیع شده و کارگزار پشتیبان.
- پنهانی (Transparency)
  - کاربران باید سیستم را همانند یک سیستم ساده مبتنی بر گذرواژه ببینند.
- مقیاس‌پذیری (Scalability)
  - قابلیت کار با تعداد زیادی ایستگاه کاری و کارگزار با پشتیبانی از ساختاری پیمانه‌ای و توزیع‌شده.

ویژگیهای عمومی کربروس

- چند تعریف
  - **دامنه:** یک محدوده دسترسی را مشخص می‌کند. به نوعی معادل دامنه‌های تعریف شده در ویندوز است.
  - **مرکز توزیع کلید:** معادل کارگزار کربروس است.
  - **عامل (Principal):** به سرویس‌ها، دستگاه‌ها، کاربران و کلیه عناصری که نیاز به شناساندن و احراز خود به کارگزار کربروس دارند، عامل گفته می‌شود.

کربروس

- برای معرفی کربروس به صورت گام به گام از پروتکلهای ساده شروع می‌کنیم و سعی می‌کنیم اشکالات هر یک را برطرف کنیم تا به کربروس برسیم.

فهرست

- مقدمه و نیازمندی‌ها
- **دیالوگ‌های ساده احراز اصالت**
- کربروس نسخه ۴
- کربروس نسخه ۵

دیالوگ ساده احراز اصالت

- فرض: بین کارگزار احراز اصالت (AS) و هر کارگزار سرویس (V) یک کلید مشترک وجود دارد.
- درخواست خدمات توسط کاربر از کارگزار سرویس:
  - **C→AS:** $ID_C \parallel P_C \parallel ID_V$
  - **AS→C:** Ticket
  - **C→V:** $ID_C \parallel$ Ticket
  - Ticket = $E(K_v, [ID_C \parallel AD_C \parallel ID_V])$

C = client AS= authentication server

V =server P_C= password of user on C

AD_C = network address of C

K_v = secret encryption key shared by AS and V

بلیط

- در واقع نوعی گواهی است که هنگام ورود کاربر به دامنه کربروس به او داده می‌شود که بیانگر اعتبار او برای دسترسی به منابع شبکه است.

بررسی دیالوگ

- چرا آدرس کارفرما (Client) در بلیط ذکر می‌شود؟
  - در غیر این صورت هر شخصی که بلیط را از طریق شنود به دست آورد نیز می‌تواند از امکانات استفاده کند. اما اکنون تنها خدمات به آدرس ذکر شده در بلیط ارایه می‌شود.
    - مشکل جعل آدرس
- چرا شناسه کارفرما ID_C در گام سوم به صورت رمز نشده ارسال می‌شود؟
  - زیرا این اطلاعات به صورت رمز شده در بلیط وجود دارد.
  - اگر شناسه با بلیط مطابقت نداشته باشد خدمات ارایه نمی‌شوند.

مشکلات دیالوگ ساده احراز اصالت

- ناامنی
  - ارسال کلمه عبور بدون رمزگذاری
  - امکان حمله تکرار (با شنود، جعل شناسه و جعل آدرس کاربر)
- ناکارآیی
  - لزوم تقاضای بلیط جدید برای هر خدمت

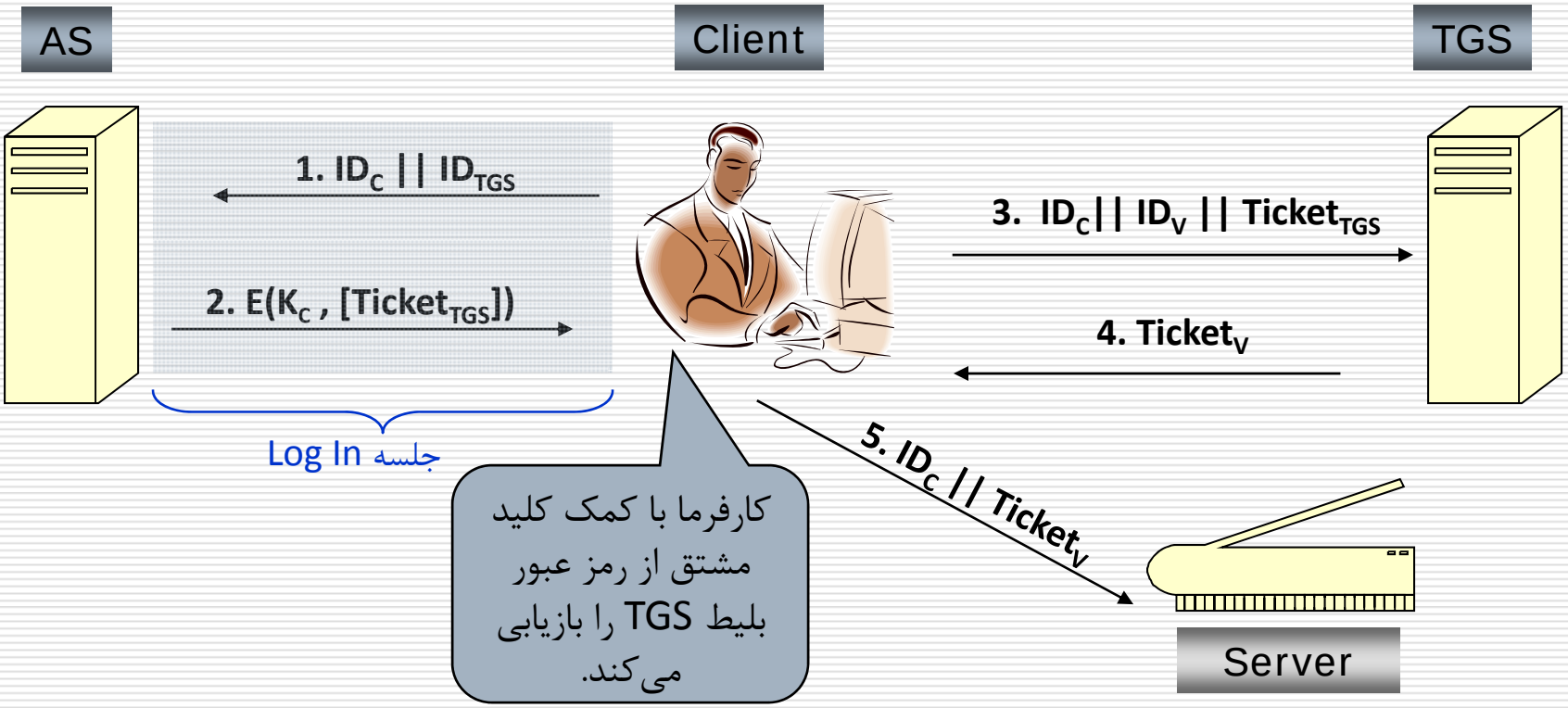
استفاده مجدد از بلیط‌ها

- چرا استفاده مجدد از بلیط‌ها اهمیت دارد؟
  - جلوگیری از تایپ مجدد کلمه عبور در یک بازه زمانی کوتاه
  - پنهانی احراز اصالت
    - کاربر متوجه فرآیندهای احراز اصالت نشود.

ارتقای امنیت–دیالوگ ۱

- استفاده از یک کارگزار جدید با نام کارگزار اعطا کننده بلیط
  - TGS: Ticket Granting Server
- کارگزار احراز اصالت، AS، کماکان وجود دارد.
  - بلیط "اعطاء بلیط" ticket-granting ticket توسط آن صادر می‌شود.
- بلیط‌های اعطاء خدمات توسط TGS صادر می‌شوند.
  - بلیط "اعطاء خدمات" service-granting ticket
- اجتناب از انتقال کلمه عبور با رمز کردن پیام کارگزار احراز اصالت (AS) به کارفرما توسط کلید مشتق شده از کلمه عبور (K_C)

ارتقای امنیت–دیالوگ۱



$$Ticket_{TGS} = E\left(K_{TGS}, [ID_C \| Addr_C \| ID_{TGS} \| Timestamp_1 \| Lifetime_1]\right)$$

$$Ticket_V = E\left(K_V, [ID_C \| Addr_C \| ID_V \| Timestamp_2 \| Lifetime_2]\right)$$

ارتقای امنیت–دیالوگ۱

- پیام‌های شماره ۱ و ۲ به ازاء هر جلسه Log on رد و بدل می‌شوند.
- پیام‌های شماره ۳ و ۴ به ازاء هر نوع خدمات رد و بدل می‌شوند.
- پیام شماره ۵ به ازاء هر جلسه خدمات رد و بدل می‌شود.

1. **C→AS:** $ID_C \parallel ID_{TGS}$
2. **AS→C:** $E(K_c, Ticket_{TGS})$
3. **C→TGS:** $ID_C \parallel ID_V \parallel Ticket_{TGS}$
4. **TGS→C:** $Ticket_v$
5. **C→V:** $ID_C \parallel Ticket_v$

محتوی بلیط ها

- بلیط اعطای بلیط:

$$Ticket_{TGS} = E\left(K_{TGS}, [ID_C \| Addr_C \| ID_{TGS} \| Timestamp_1 \| Lifetime_1]\right)$$

- بلیط اعطای خدمات:

$$Ticket_V = E\left(K_V, [ID_C \| Addr_C \| ID_V \| Timestamp_2 \| Lifetime_2]\right)$$

ویژگی های دیالوگ ۱

- دو بلیط صادر شده ساختار مشابهی دارند. در اساس به دنبال هدف واحدی هستند.
- رمزنگاری $Ticket_{TGS}$ جهت احراز اصالت
  - تنها کارفرما می‌تواند به بلیط رمزشده دسترسی پیدا کند.
- رمز نمودن محتوای بلیط‌ها صحت را فراهم می‌کند.
- استفاده از مُهر زمانی (Timestamp) در بلیط‌ها، آنها را برای یک بازه زمانی تعریف شده قابل استفاده مجدد می‌کند.
- هنوز از آدرس شبکه برای احراز اصالت بهره می‌گیرد.
  - چندان جالب نیست زیرا آدرس شبکه جعل (Spoof) می‌شود.
  - با این حال، درجه‌ای از امنیت مهیا می‌شود.

نقاط ضعف دیالوگ ۱

- مشکل زمان اعتبار بلیط‌ها:
  - زمان کوتاه: نیاز به درخواست‌های زیاد گذرواژه
  - زمان بلند: خطر حمله تکرار

- احراز اصالت یکطرفه: عدم احراز اصالت کارگزار توسط کارفرما
  - رسیدن درخواست‌ها به یک کارگزار غیرمجاز

فهرست

کربروس نسخه ۴

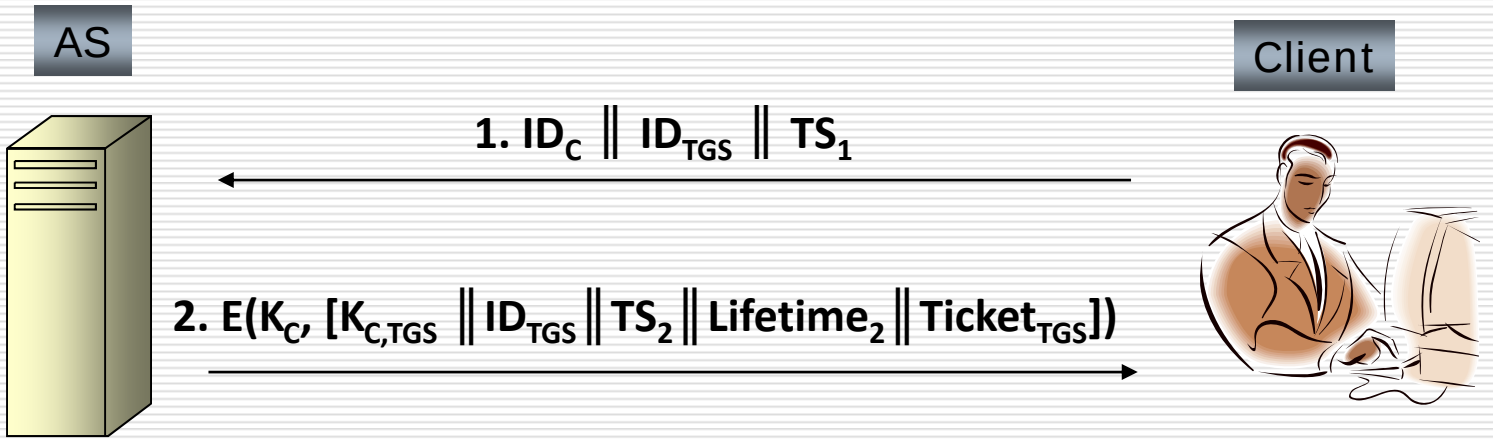
- توسعه یافته پروتکل‌های قبلی است.
- مشکل حمله تکرار حل شده است.
- احراز اصالت دو جانبه صورت می‌گیرد.
  - کارگزاران و کارفرمایان هر دو از اصالت هویت طرف مقابل اطمینان حاصل می‌کنند.

مقابله با حمله تکرار

- یک نیاز جدید: کارگزار یا TGS باید اطمینان حاصل نمایند که کاربرِ بلیط، همان کسی است که بلیط برای او صادر شده.

- مفهوم جدیدی به نام اعتبارنامه (Authenticator) ابداع شده است:
  - علاوه بر بلیط‌ها از مفهوم کلید جلسه بهره می‌جوید.

بدست آوردن بلیط اعطاء بلیط

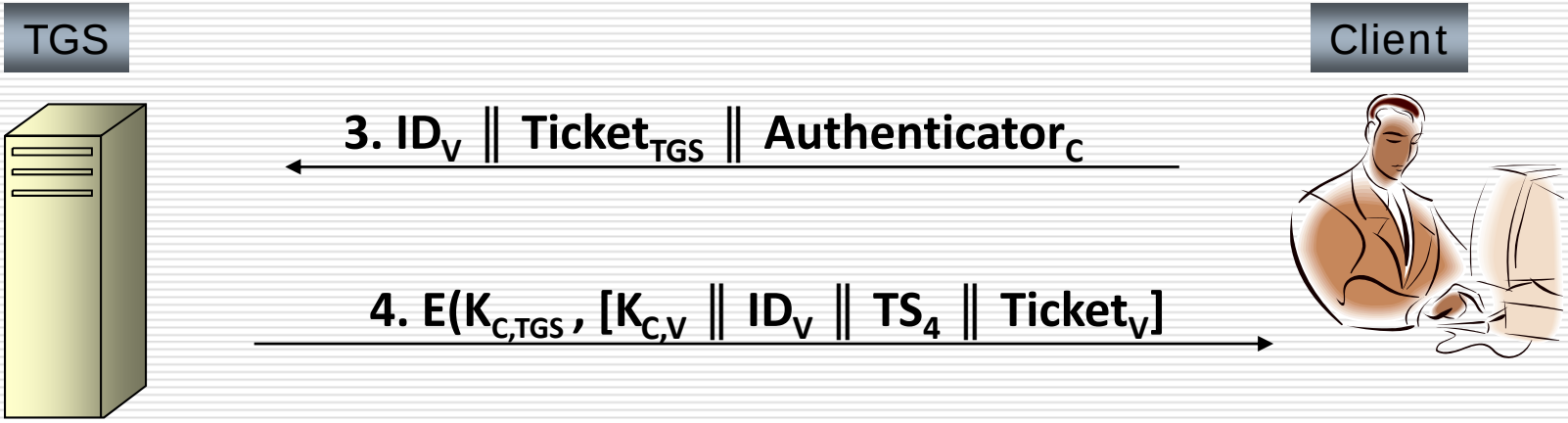


$$Ticket_{TGS}=E(K_{TGS}, [K_{C,TGS} \| ID_C \| Addr_C \| ID_{TGS} \| TS_2 \| Lifetime_2])$$

بدست آوردن بلیط اعطاء بلیط

- نتایج این مرحله برای کارفرما
  - بدست آوردن امن بلیط "اعطاء بلیط" از AS
  - بدست آوردن زمان انقضای بلیط (TS_2)
  - بدست آوردن کلید جلسه امن بین کارفرما و TGS

بدست آوردن بلیط اعطاء خدمات



$$Ticket_V = E(K_V, [K_{C,V} \| ID_C \| Addr_C \| ID_V \| TS_4 \| Lifetime_4])$$

$$Authenticator_C = E(K_{C,TGS}, [ID_C \| Addr_C \| TS_3])$$

بدست آوردن بلیط اعطاء خدمات

- نتایج این مرحله برای کارفرما
  - جلوگیری از حمله تکرار با استفاده از یک اعتبار نامه (Authenticator) یکبار مصرف که عمر کوتاهی دارد.
  - بدست آوردن کلید جلسه برای ارتباط با کارگزار V

دستیابی به خدمات کارگزار

$Ticket_V = E(K_V, [K_{C,V} \| ID_C \| Addr_C \| ID_V \| TS_4 \| Lifetime_4])$

$Authenticator_C = E(K_{C,V}, [ID_C \| Addr_C \| TS_5])$

دستیابی به خدمات کارگزار

- نتایج این مرحله برای کارفرما
  - احراز اصالت کارگزار در گام ششم با برگرداندن پیغام رمزشده
  - جلوگیری از بروز حمله تکرار

کربروس نسخه ۴: شمای کلی

(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

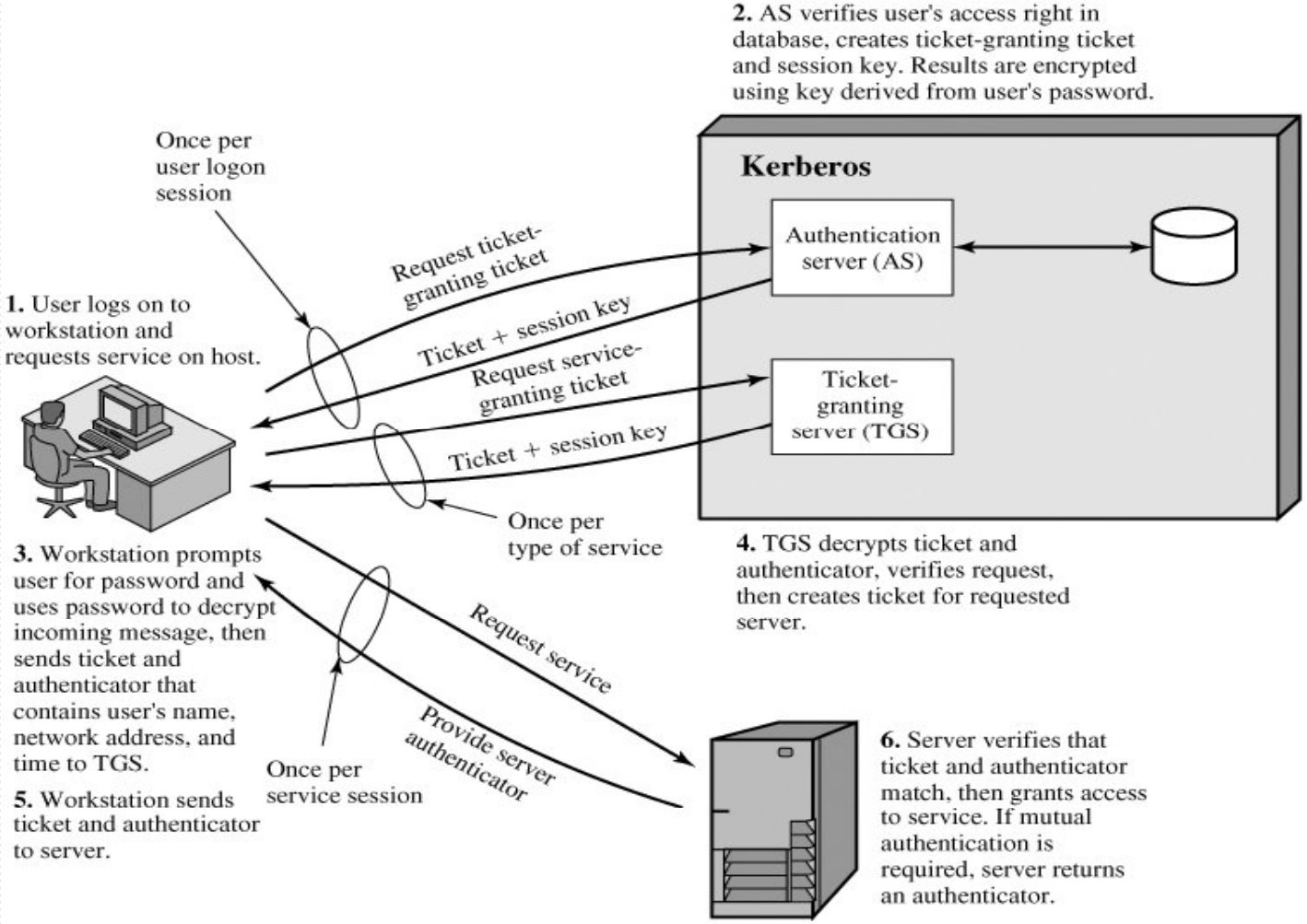
(5) C → V: $Ticket_v \| Authenticator_c$

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

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

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

کربروس نسخه ۴: شمای کلی



دامنه کربروس (realm)

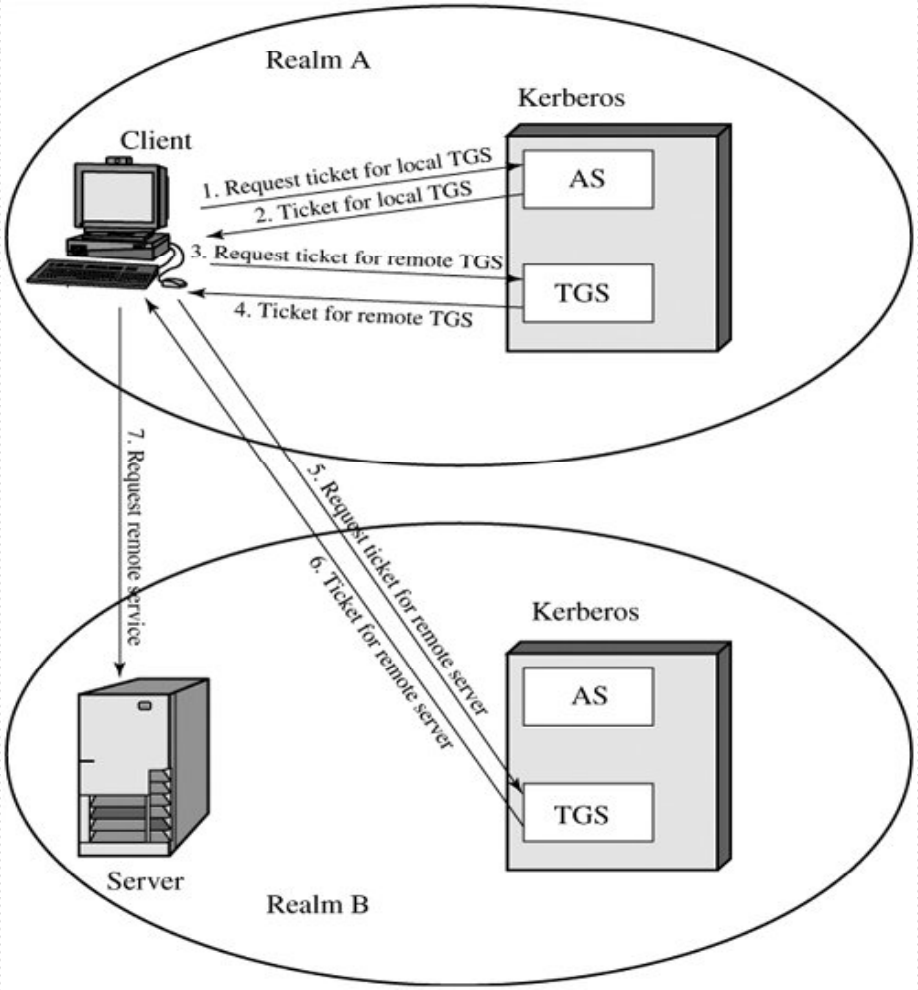
- دامنه کربروس از بخش‌های زیر تشکیل شده است:
  - کارگزار کربروس
  - کارفرمایان
  - کارگزاران برنامه‌های کاربردی (Application Servers)
- کارگزار کربروس گذرواژه تمام کاربران را در پایگاه‌داده‌های خود دارد.
- کارگزار کربروس با هر کارگزار برنامه کاربردی کلیدی مخفی به اشتراک گذاشته است.
- معمولاً هر دامنه معادل یک حوزه مدیریتی است.

تعامل بین دامنه‌ای

- وجود بیش از یک دامنه کربروس
- نیاز به دریافت سرویس از کارگزاری در دامنه دیگر
- نیاز به وجود کلید مشترک بین هر دو کارگزار کربروس
- رویه پیشنهادی:
  - احراز اصالت کاربر توسط کارگزار کربروس
  - دریافت بلیط از TGS محلی برای ارتباط با TGS دامنه بیرونی
  - ارتباط با TGS دامنه بیرونی برای دریافت بلیط برای دریافت سرویس
  - ارائه بلیط به کارگزار سرویس دامنه بیرونی برای دریافت سرویس

احراز اصالت بین دامنه‌ای

- **C→AS:** $ID_C \| ID_{TGS} \| TS_1$
- **AS→C:** $E(K_C, [K_{C,TGS} \| ID_{TGS} \| TS_2 \| Lifetime_2 \| Ticket_{TGS}])$
- **C→TGS:** $ID_{TGSrem} \| Ticket_{TGS} \| Authenticator_C$
- **TGS→C:** $E(K_{C,TGS}, [K_{C,TGSrem} \| ID_{TGSrem} \| TS_4 \| Ticket_{TGSrem}])$
- **C→TGS_{rem}:** $ID_{Vrem} \| Ticket_{TGSrem} \| Authenticator_C$
- **TGS_{rem} → C :** $E(K_{C,TGSrem}, [K_{C,Vrem} \| ID_{Vrem} \| TS_6 \| Ticket_{Vrem}])$
- **C→V_{rem}:** $Ticket_{Vrem} \| Authenticator_C$



فهرست

- مقدمه و نیازمندی‌ها
- دیالوگ‌های ساده احراز اصالت
- کربروس نسخه ۴
- **کربروس نسخه ۵**

کربروس نسخه ۵

مشخصات

- در اواسط دهه 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 به عنوان روش اصلی احراز اصالت کاربران استفاده می‌کند.

مشکلات کربروس نسخه ۴ و نحوه رفع آنها در نسخه ۵

- وابستگی به یک سیستم رمزنگاری خاص (DES)
  - + در نسخه ۵ می‌توان از هر الگوریتم متقارن استفاده کرد.

- وابستگی به IP
  - + در نسخه ۵ می‌توان از هر نوع آدرس شبکه (مثلا OSI یا IP) استفاده کرد.

- محدود بودن زمان اعتبار بلیط‌ها (مضربی از ۵ دقیقه تا سقف ۲۱ ساعت)
  - + در نسخه ۵ می‌توان ابتدا و انتهای بازه را مشخص کرد.

مشکلات کربروس نسخه ۴ و نحوه رفع آنها در نسخه ۵

- امکان انتقال اعتبار یک کاربر به یک سرور دیگر وجود ندارد.
  - + مثلا DBMS نیاز دارد برای پاسخ دادن به پرس‌وجوی کاربر، برخی داده‌ها را از یک پایگاه داده دیگر بگیرد.

- با افزایش تعداد دامنه‌ها، تعداد کلیدها بصورت تصاعدی افزایش می‌یابد.
  - + در نسخه ۵ این مشکل حل شده است.

مشکلات کربروس نسخه ۴ و نحوه رفع آنها در نسخه ۵

- رمزگذاری مضاعف در مرحله ۲ و ۴ (با کلید کارگزار سرویس و کلید کاربر)
  - + در نسخه ۵ از این هزینه اضافی جلوگیری شده است.
- استفاده از مُد غیراستاندارد PCBC در استفاده از DES، که آسیب پذیر است.
  - + در نسخه ۵ از مُد CBC استفاده می شود و قبل از رمز شدن چکیده پیام نیز به آن اضافه می شود.
- امکان حمله دیکشنری برای استخراج گذرواژه و جعل کاربر.
  - + در نسخه ۵ با استفاده از پیش‌احراز اصالت این حمله را سخت‌تر کرده، ولی به طور کامل جلوی آن را نگرفته است.

کربروس نسخه ۵: شمای کلی

(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) $C \rightarrow V$: $Options \| Ticket_v \| Authenticator_c$

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

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

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

کربروس نسخه ۵

Authentication Service Exchange

- Realm: دامنه کاربر
- Options: تقاضای وجود برخی پارامترها در بلیط درخواستی
- Times: زمان شروع و پایان اعتبار بلیط
- Nonce: عدد تصادفی برای اطمینان از تازگی پیام دوم

Client/Server Authentication Exchange

- Subkey: کلید اختیاری کاربر برای حفاظت از نشست جاری. در صورت خالی بودن این فیلد، از $K_{C,V}$ استفاده می‌شود.
- Seq#: شماره سریال آغازین برای استفاده در پیام‌های ارسالی از کاربر به کارگزار و بالعکس.

پیاده‌سازی‌های موجود

- دانشگاه MIT: اولین پیاده سازی کربروس که هنوز به عنوان مرجع مورد استفاده قرار می‌گیرد.
- Heimdal: پیاده‌سازی انجام شده در خارج آمریکا.
- Active Directory: پیاده‌سازی ارائه شده توسط مایکروسافت که در RFC 1510 آمده است.

پایان

مرکز امنیت داده و شبکه شریف

http://dnsl.ce.sharif.edu

پست الکترونیکی

m_amini@ce.sharif.edu

شبکه آموزشی- پژوهشی مادسیج
با هدف بهبود پیشرفت علمی
و دسترسی راحت به اطلاعات
برای جامعه بزرگ علمی ایران
ایجاد شده است

madsg.com

مادسیج

IRan Education & Research NETwork
(IRERNET)