4900 Kerberos Administration

Kerberos Grundlagen und Kommunikationsablauf

Peter Jahn, MSc
System Engineer
training@LinuxCampus.net

Was ist Kerberos

- Antwort 1
  - Dreiköpfiger Höllenhund aus der griechischen Mythologie der den Eingang zur Unterwelt bewacht

- Antwort 2
  - Ein in 1980 durch MIT entwickeltes Authentication System um in einem nicht sicheren TCP/IP Netzwerk sicher auf Ressourcen zugreifen zu können
  - Authentication, Authorization, Audit

Was ist Kerberos

- Eigenschaften von Kerberos
  - Basiert auf symmetrischen Schlüsseln (DES, 3DES,AES...)
  - Keine Übertragung von Passwörtern übers Netz
  - Erlaubt *Single Sign On* Implementierungen

Begriff	Bedeutung
KDC	Key Distribution Center
AS	Authentication Server/Service
TGS	Ticket Granting Server/Service
Ticket Cache	Gespeicherte Tickets ermöglichen Single Sign On
Keytab	Gespeicherter Anwendungsschlüssel
Kerberized Applikation	Ein Programm welches Kerberos unterstützt

Kerberos MIT

- MIT
  - Referenzimplementierung
  - v5 für Linux und Windows erhältlich
  - weit verbreitet
  - wird in den USA entwickelt
  - Verschlüsselungsverfahren sind DES, 3DES, AES und RC4
  - Prüfsummenverfahren sind MD5, SHA-1, HMAC und CRC32
  - wird von vielen Anwendungen unterstützt

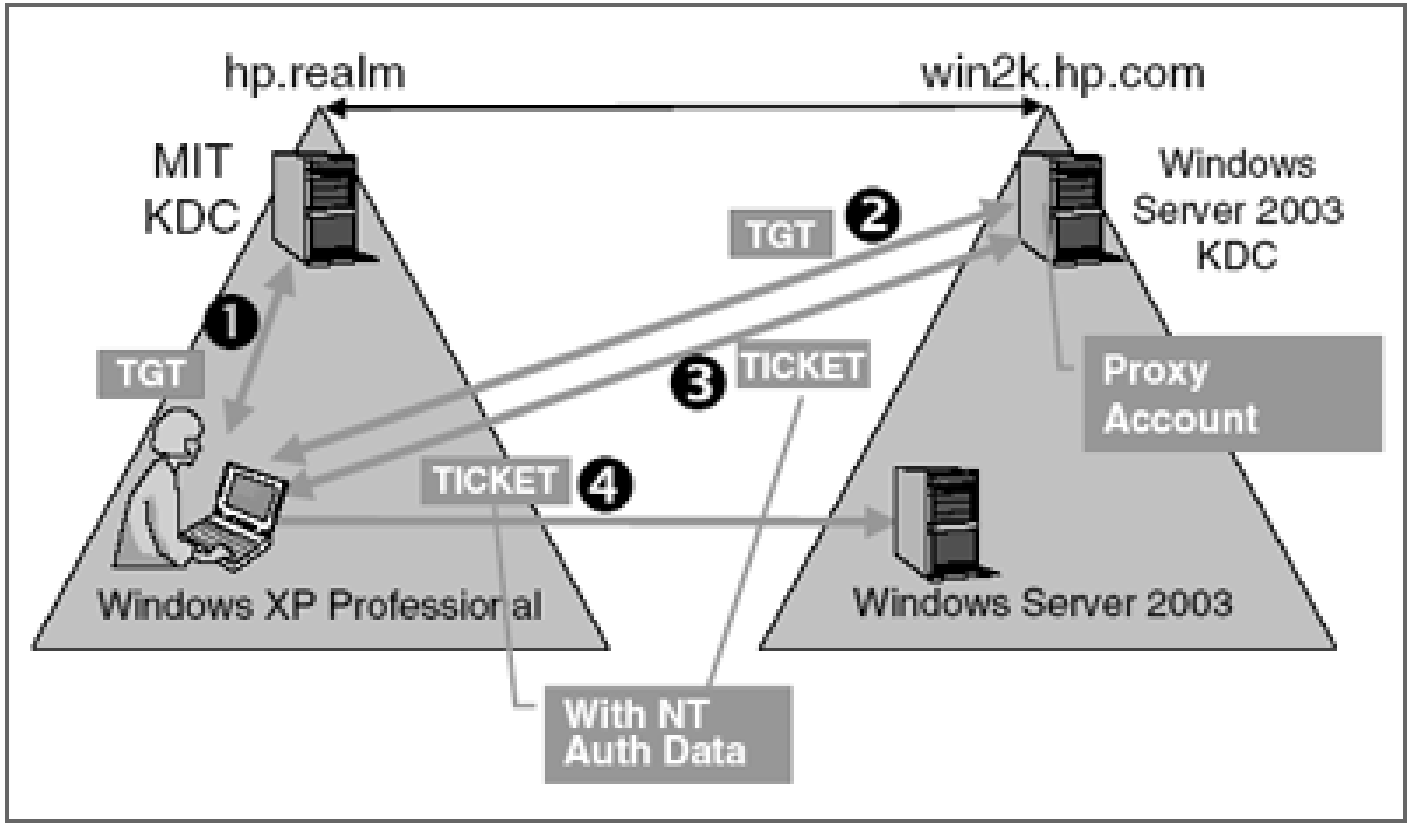
Kerberos Heimdal

- Heimdal
  - neuere Implementierung
  - nur für Linux erhältlich
  - wird außerhalb der USA entwickelt
  - bessere Master/Slave Synchronisierung
  - Unterstützung von (3)DES, AES und RC4
  - Kerberos API und GSSAPI unterscheiden sich jedoch von den MIT APIs
  - Standard bei Samba 4
    - MIT Unterstützung ist in Entwicklung

Kerberos Microsoft

- Microsoft
  - Implementierung seit Windows 2000
  - unterstützt nur Kerberos 5
  - unterstützt RC4, DES, AES, aber kein 3DES
  - Unix Clients Einbindung mit Windows KDC ist möglich
  - Windows Clients mit Unix KDC ist nicht ohne weiteres möglich
    - benötigt einen eigenen Kerberos Client
  - kein GSSAPI (dafür SSPI)

Cross Realm Trust MS/Linux



Kerberos Komponenten

Kerberos Komponenten

Ressource

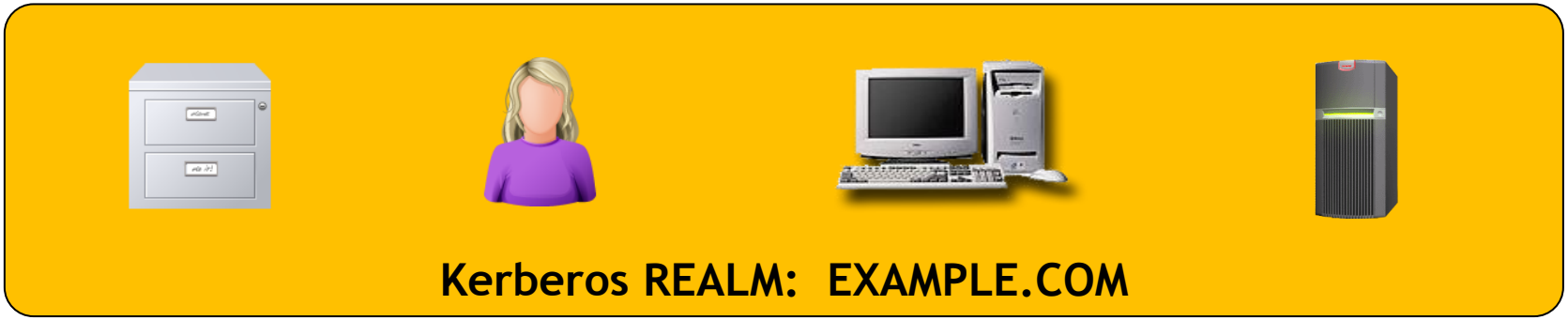
Kerberos Server

Maria

Maria's PC

Eine typische Kerberos Umgebung

Kerberos Realm

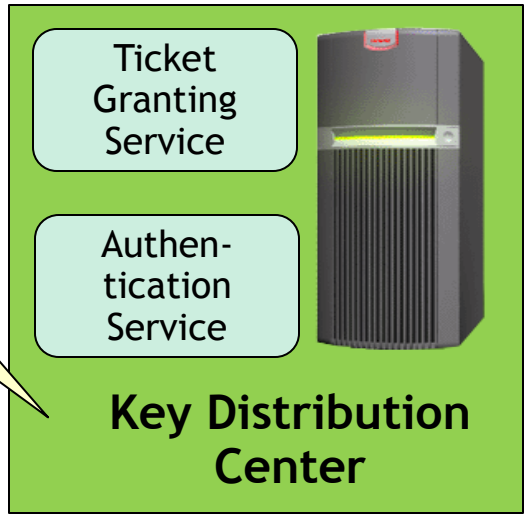


- Kerberos Realm
  - Ein administrativer Bereich zur Verwaltung von
    - Benutzerkennungen, Rechnern und Services
  - Eine Realm kann hierarchisch strukturiert sein
  - Jede REALM hat einen eindeutigen Namen, angelehnt an DNS Domainnamen wobei der Name der Realm in Großbuchstaben geschrieben wird
  - DNS: example.com -> REALM: EXAMPLE.COM

Key Distribution Center

Ressource

Ein KDC
besteht aus 2 Services



Maria

Maria's PC

Ein Kerberos Server wird als Key Distribution Center bezeichnet

Key Distribution Center

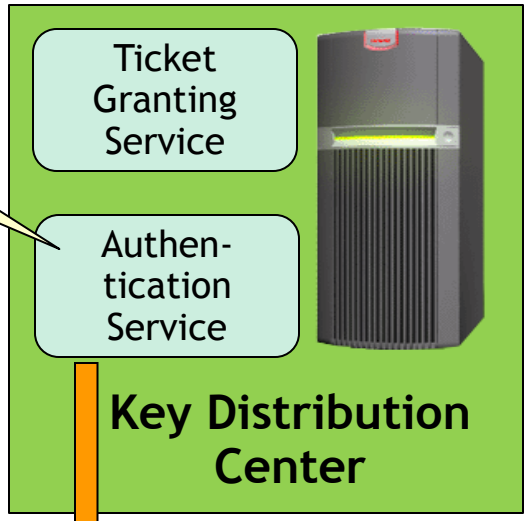
- Key Distribution Center (KDC)
  - Ein KDC besteht aus dem Authentication Server (AS) und dem Ticket Granting Server (TGS)
  - Der KDC muss besonders geschützt werden da es die Datenbank mit den User- und Serverprincipals der Realm enthält
  - Aus Redundanzgründen sind auch weitere KDC-Server (Slaves) möglich welche im RO-Modus betrieben werden
  - Die Slaves gleichen die Datenbank regelmäßig über ein Protokoll mit dem Master KDC ab

Key Distribution Center

Ressource

Ich überprüfe ob ein Benutzer/Client auch wirklich der ist für den er sich ausgibt

Ich kenne die folgenden Principals:
- Benutzer Maria
- PC Maria
- Ressource X

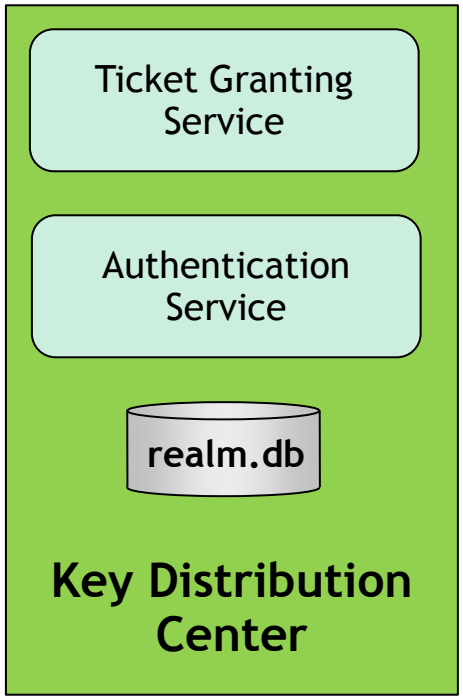


Maria

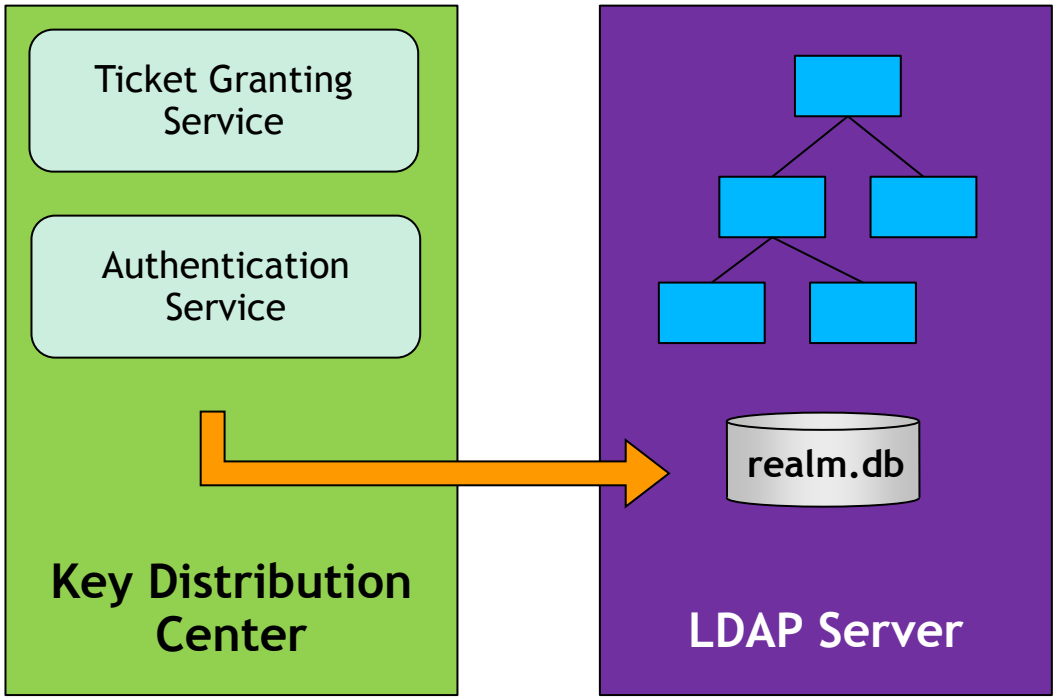
Maria's PC

Ein Authentication Service muss den Benutzer in (s)einer Datenbank finden

Datenbank Szenarien



Integrierte Datenbank



Ausgelagerte Datenbank

Terminologie

- Principals
  - sind Teilnehmer des Kerberos Authentisierungssystems
  - Eindeutige Identifikation eines Benutzers oder Dienstes innerhalb einer REALM
  - Jeder Principal muss vor der Nutzung in der REALM-Datenbank angelegt werden
  - Besitzt eine ID und ein "Geheimnis"
    - "Geheimnis" ist nur dem Principal und dem KDC bekannt
  - Principals werden durch kadmin erzeugt und in der KDC Datenbank gespeichert

Userprincipals

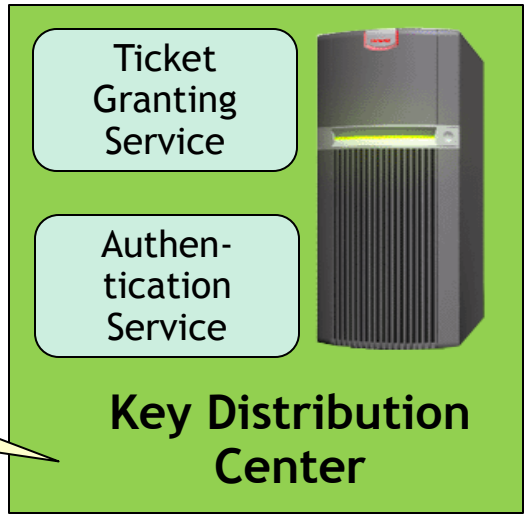
- Userprincipals
  - Die ID besteht aus einem Namen, einem @ und einer REALM
    - user@REALM -> jahn@EXAMPLE.COM
  - Als "Geheimnis" wird ein Passwort verwendet welches der Benutzer mit kpasswd ändern kann
  - BenutzerID kann eine Instance beinhalten
    - user/instance@realm -> jahn/admin@EXAMPLE.COM
    - Instanzen weisen Benutzer unterschiedliche Rollen zu
      - über Rollen können unterschiedliche Rechte definiert werden

Ressource

Ressource

Repräsentiert eine Ressource welche eine Kerberos Authentifizierung benötigt

Ich kenne diese Ressource und habe einen passenden Service-Principal dafür



Maria

Maria's PC

Ressource kann sein: Web Server, FTP-Server, SSH-Server, Datei-Freigabe

Serviceprincipals

- Serviceprincipals
  - Die ID besteht aus einer Serviceangabe, gefolgt von einem Slash '/' sowie dem vollqualifizierten Rechnernamen des Servers (FQDN)
  - Als Serviceangabe sind u.a. host, ftp, ldap, pop, ... erlaubt
    - Syntax: service/dns.domain.name@REALM
      - host/sv1.example.com@EXAMPLE.COM
      - ftp/sv2.example.com@EXAMPLE.COM
  - Als "Geheimnis" wird ein Zufallswert verwendet welches auf dem Server in der Datei /etc/keytab hinterlegt wird
    - Der Schlüssel eines Servers kann nicht geändert, sondern lediglich neu erzeugt werden

Serviceprincipals

- Serviceprincipals und Linux
  - Dienste auf Linux werden so konfiguriert, dass sie ihr Passwort aus einer sogenannten keytab-Datei lesen

```
# Beispiel Apache auf Debian
<Directory /var/www/>
    Krb5Keytab /etc/apache2/krb5.keytab
    ...
```

- Serviceprincipals und Windows Server
  - Server Dienste können ihr Passwort direkt aus dem Active Directory auslesen und es muss keine keytab-Datei erstellt werden

Key Distribution Center

Ressource

Ich erstelle Tickets damit User/Clients auf Kerberos geschützte Ressourcen zugreifen können

Ticket Granting Service

Authen-tication Service

Key Distribution Center

Maria

Maria's PC

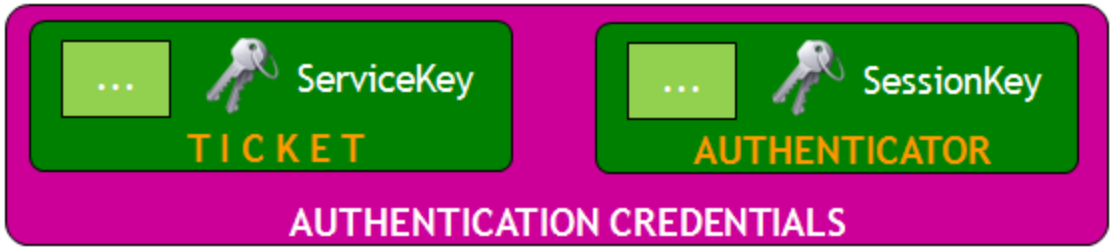
Ein spezielles Ticket wird pro geschützter Ressource benötigt

Ticket

- Authentisierung von Benutzer/Dienst via Ticket
  - Ein Ticket wird vom Client beim KDC für einen bestimmten Dienst angefordert und in einem Ticketcache gespeichert
  - Gegenseitige Authentisierung von Benutzer und Dienst (Mutual Authentication)
  - Das Ticket wird mit dem Serviceschlüssel des Servers verschlüsselt und enthält:

Ticket

- Authentication Credentials
  - Das Ticket wird vor der Authentisierung um einen Zeitstempel (Authenticator) ergänzt
  - Ticket + Authenticator wird Authentication Credential genannt



Ticket Cache

- Ticket Cache im Detail
  - Ein Ticket ist normalerweise 10-24 Stunden gültig und kann innerhalb dieser Zeit beliebig oft verwendet werden
  - Ausgestellt Tickets werden am Client gespeichert

- file based credential cache (MIT)
  - alle Tickets werden in einer Datei unter /tmp gespeichert
    - /tmp/krb5cc_UID
- memory based credential cache (Microsoft, Apple)
  - alle Tickets werden im Arbeitsspeicher hinterlegt und beim Abmelden gelöscht

Ticket Granting Ticket

- Ticket Granting Ticket (TGT)
  - wird beim Authentication Server (AS) angefordert, wenn sich der Benutzer mit seinem Passwort authentisiert
  - ist ein Service-Ticket für den Ticket Granting Server (TGS)
    - Automatische Beantragung beim Anmelden oder manuell durch kinit
  - TGT hat ein default Lifetime von 8 Stunden (änderbar) und wird im Ticketcache des Client gespeichert
  - Für die Authentisierung bei Anforderung von Servicetickets am TGT
  - Der Benutzer muss sein Passwort nicht noch einmal eingeben
  - Beim Abmelden wird der Ticketcache automatisch gelöscht

Beispiel - Anmelden

```
# Holen eines Ticket Granting Tickets
 root@tux# kinit peter
Password for peter@EXAMPLE.COM: <password>
```

```
# Anzeigen des Ticket Cache
 root@tux# klist -f
Ticket cache: FILE:/tmp/krb5cc_1456
Default principal: peter@EXAMPLE.COM

Valid starting       Expires              Service principal
12/20/11 10:48:29    12/20/11 22:48:29    krbtgt/EXAMPLE.COM@EXAMPLE.COM
Flags: FI
# Flag I: Initial
# Flag F: Forwardable
```

Beispiel - Telnet

```
# Telnetverbindung zu einem Rechner aufbauen (-a Attempt automatic login)
 root@tux#  telnet  -a  sv1.example.com
Connected to sv1.example.com (10.0.1.10).
[ Kerberos V5 accepts you as “peter@EXAMPLE.COM” ]
Welcome to SuSE Linux 11.3 (i386) - Kernel %r (%t).
*** Connection not encrypted! Communication may be eavesdropped.
peter@sv1:~>
```

```
# Verschlüsselte Telnetverbindung (-x) zu einem Rechner aufbauen
 root@tux#  telnet  -x  -a  sv1.example.com
Connected to sv1.example.com (10.0.1.10).
Waiting for encryption to be negotiated...
[ Kerberos V5 accepts you as “peter@EXAMPLE.COM” ]
Welcome to SuSE Linux 11.3 (i386) - Kernel %r (%t).
peter@sv1:~>
```

Beispiel - Ticketcache

```
# Anzeigen des Ticket Cache
 root@tux# klist  -f
Ticket cache: FILE:/tmp/krb5cc_1456
Default principal: peter@EXAMPLE.COM

Valid starting       Expires              Service principal
12/20/11 10:48:29    12/20/11 22:48:29    krbtgt/EXAMPLE.COM@EXAMPLE.COM
Flags: FI
12/20/11 10:52:29    12/20/11 22:52:29    host/sv1.example.com@EXAMPLE.COM
Flags: F
```

```
# Ticket Cache leeren
 root@tux#  kdestroy
 root@tux#  klist  -f
 root@tux#
```

Kerberos Vorraussetzungen

- Vorraussetzungen
  - Zeitsynchronisierte Systeme (max. 5 Minuten Differenz)
    - erlaubte Zeitdifferenz kann definiert werden
  - Client und Server müssen korrekt im DNS eingetragen sein
  - KDC's müssen speziell gesicherte Systeme sein
  - Mindestens zwei KDC's (Master, Slave)
  - Backupkonzept für REALM Datenbank

Protokolle

- Generic Security Services API (GSSAPI):
  - generisches Interface zur Unterstützung von "strong Authentication", wie z.B. Kerberos wird oft von Services verwendet um Kerberos zu unterstützen
- Security Support Provider Interface (SSPI):
  - Microsoft Pendant zu GSSAPI
- Simple and Protected GSSAPI Negotiation Mechanism (SPNEGO):
  - Übermittlung eines Kerberos-Tickets per HTTP

Kerberos Kommunikation im Detail

Ticket Granting Service

Ressource

Hallo AS, kann ich bitte ein Ticket haben damit ich damit Service Tickets beantragen kann? 3

Zuerst benötige ich aber ein Ticket vom Ticket Granting Service. 2

Ich würde gerne auf die Ressource zugreifen 1

Maria

Maria's PC

Ticket Granting Service

Authen-tication Service

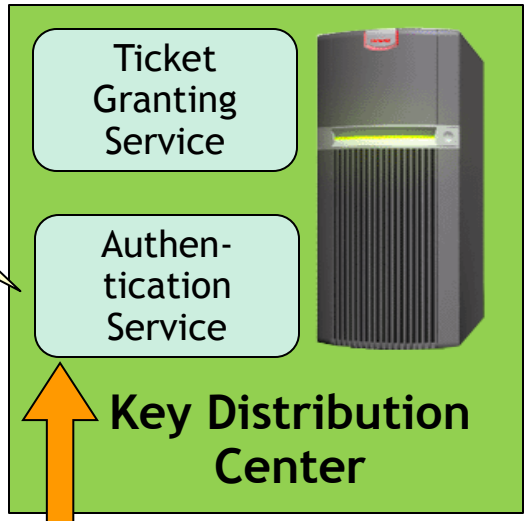
Key Distribution Center

Maria möchte auf eine Kerberos geschützte Ressource zugreifen

Ticket Granting Ticket

Ressource

Okay, Ich habe das Kuvert mit deinem geheimen Passwort gesperrt. Wenn du es öffnen kannst findest du darin mein Ticket Granting Ticket (TGT)



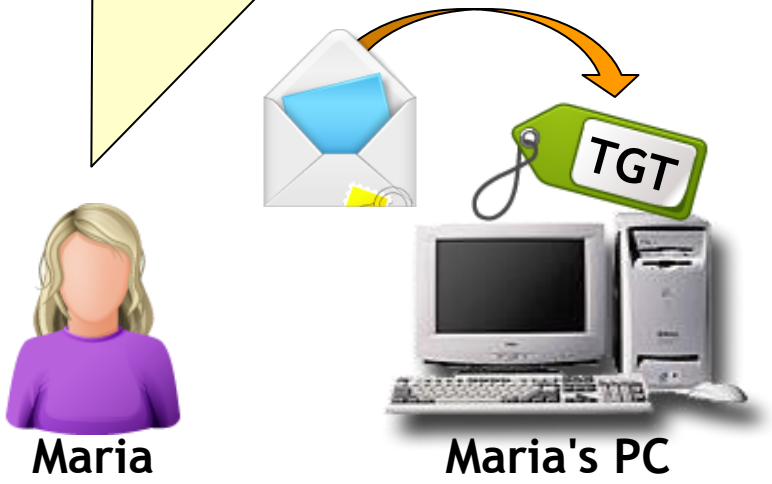
Maria

Maria's PC

Maria musste nicht ihr Passwort an das AS schicken. Das AS hat Zugriff auf die Datenbank und die Passwörter der User

Ticket Granting Ticket

Super, ich konnte das Kuvert öffnen (Nachricht entschlüsseln) und bin jetzt Besitzer eines Ticket Granting Ticket (TGT)
Damit habe ich bewiesen dass ich Maria bin :-)



Das TGT wird benutzt um Service Tickets zu beantragen
Das TGT beinhaltet kein Passwort

Ticket Cache

- Ticket Granting Ticket (TGT)
  - Das erhaltene TGT wird im Ticket Cache abgelegt
  - Der Inhalt des Cache kann mit kinit angezeigt werden
    - -f ...zusätzlich Ticket Flags anzeigen

```
# Anzeigen des Ticket Cache
 root@tux# klist  -f
Ticket cache: FILE:/tmp/krb5cc_1456
Default principal: maria@EXAMPLE.COM

Valid starting       Expires              Service principal
12/20/11 10:48:29    12/20/11 22:48:29    krbtgt/EXAMPLE.COM@EXAMPLE.COM
Flags: FI
```

Ticket Flags

- Ticket Flags
  - Neben den allgemeinen Ticket Informationen wie z.B den Gültigkeitszeitraum enthält ein Ticket auch noch spezielle Flags
    - Manche Flags können bei Bedarf vom Client angefragt werden und manche werden automatisch vom KDC gesetzt
  - Kerberisierte Dienste müssen oder können diese Flags auswerten

Flag	Bedeutung
R	Renewable
A	Allow-Postdate
d	Postdated
i	Invalid
F	Forwardable
A	Pre-Authenticated
H	HW-Authent
f	Forwarded
P	Proxiable
p	Proxy
0	Ok-As-Delegate
I	Inital
T	Transited Policy Checked

Ticket Granting Service

Ressource

Lass mich bitte der Ressource beweisen das ich Maria bin!
Hier ist eine Kopie meines TGT das beweist das ich wirklich Maria bin

TGT

Ticket Granting Service

Authen-tication Service

Key Distribution Center

Maria

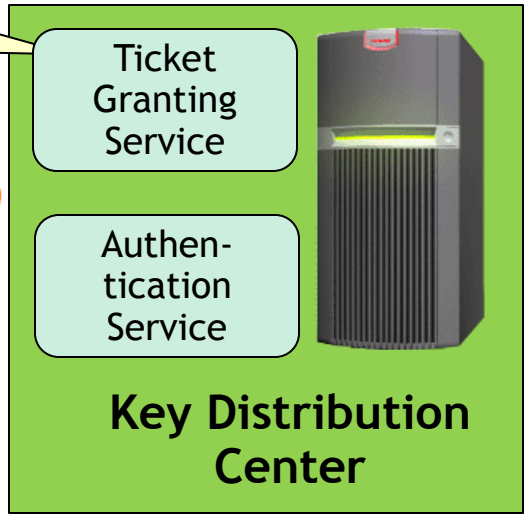
Maria's PC

Tickets für Ressourcen werden vom TGS ausgestellt

Service Ticket

Ressource

Ja stimmt, du bist wirklich Maria!
Hier hast du ein bestätigtes TGS
für die gewünschte Ressource



Maria

Maria's PC

TGS erstellt ein zeitlich beschränktes Service Ticket

Ticket Cache

- Service Ticket
  - Für jede Ressource auf die zugegriffen werden soll, muss zuerst ein Service Ticket geholt werden
  - Alle Service Tickets werden im Cache abgelegt

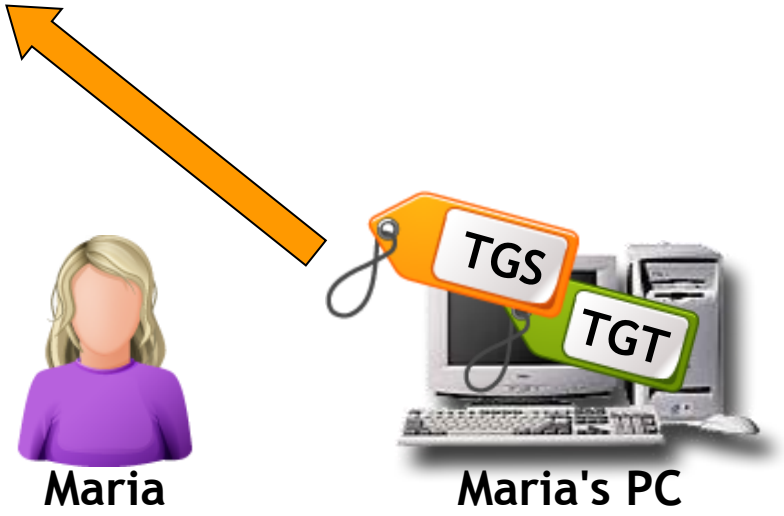
```
# Anzeigen des Ticket Cache
 root@tux# klist  -f
Ticket cache: FILE:/tmp/krb5cc_1456
Default principal: maria@EXAMPLE.COM

Valid starting       Expires              Service principal
12/20/11 10:48:29    12/20/11 22:48:29    krbtgt/EXAMPLE.COM@EXAMPLE.COM
Flags: FI
12/20/11 10:52:29    12/20/11 22:52:29    host/sv1.example.com@EXAMPLE.COM
Flags: F
```

Zugriff auf die Ressource

Ressource

Hallo Ressource! Ich bin Maria und als Beweis bekommst du mein gültiges TGS



Maria

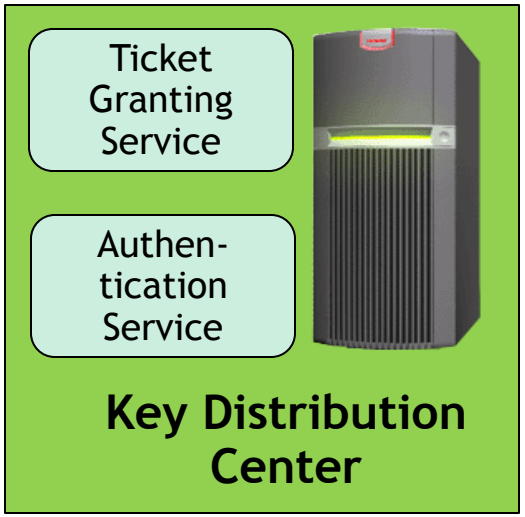
Maria's PC

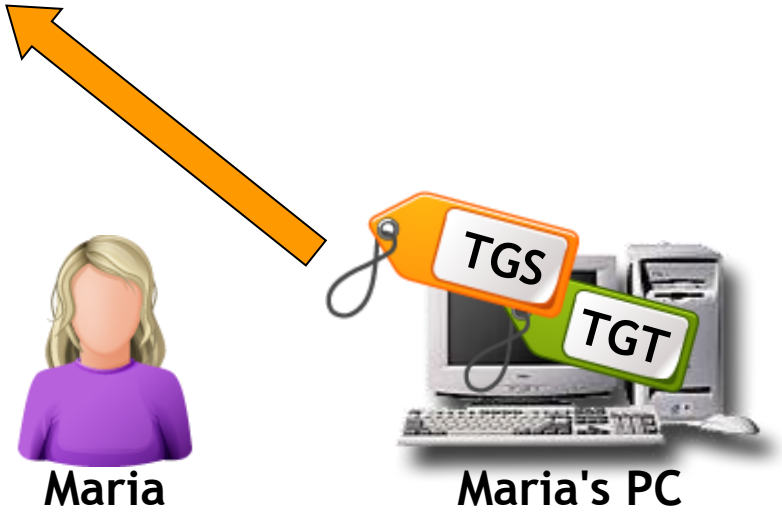
Maria möchte der Ressource beweisen, dass sie wirklich Maria ist

Ressource Authorization

Ressource

Hallo Maria! Lass mich kurz überprüfen ob der Benutzer Maria überhaupt auf mich zugreifen darf!



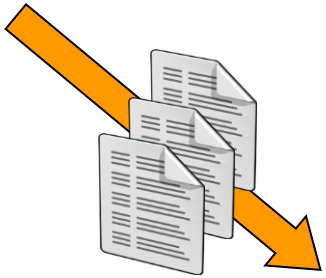


Nur weil Maria sich authentifiziert hat, bedeutet das aber nicht dass sie auch das Rechte hat auf die gewünschte Ressource zu zugreifen

Ressourcen Freigabe

Ressource

Ja du darfst auf die Ressourcen zugreifen. Hier sind die gewünschten Daten



Maria

Maria's PC

Maria darf auf die Ressource zugreifen

Erneuter Zugriff auf die Ressource

Ressource

Hallo ich bin es schon wieder, und ich möchte wieder auf die Ressource zugreifen. Anbei eine Kopie meines TGS für die Ressource

Nochmal auf die Ressource zugreifen

Maria

Maria's PC

Jedes Ticket hat ein Ablaufdatum. Solange es nicht abgelaufen ist kann es immer wieder verwendet werden

Erneute Ressource Authorization

Ressource

Aha schon wieder Maria...mal schauen ob sie noch immer auf die Ressource zugreifen darf

Maria

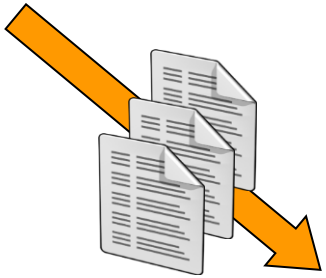
Maria's PC

In der Zwischenzeit könnte sich ja die Authorisierung verändert haben

Ressourcen Freigabe

Ressource

Ja du darfst noch immer auf die Ressourcen zugreifen. Hier sind die gewünschten Daten



Maria

Maria's PC

Maria darf noch immer auf die Ressource zugreifen


LinuxCampus.net
trainings from the experts