REGIONALES RECHENZENTRUM ERLANGEN [RRZE]

Kerberos

Systemausbildung − Kerberos

Florian Löffler, RRZE, 22.07.2020

Kerberos - Geschichte

- Teil des Projekts “Athena” (1983-1991) am MIT
  → Kerberos als Sicherheitssystem
- Ab ca. 1988 als Version 4 außerhalb des MIT
- 1993 als Version 5 erstmals spezifiziert im RFC 1510
- Standardprotokoll für die Authentifizierung ab Windows-2000
- 2005 abgelöst durch RFC 4120 für Kerberos V5

  “to clarify aspects of the protocol and its intended use that require more detailed or clearer explanation”

Ziel: Aufbau einer Campus-weiten verteilten Rechnerumgebung

Kerberos in 16 Worten

`88/tcp,88/udp`

Bezeugen der Echtheit einer Entität

Kerberos ist ein Netzwerkprotokoll zur sicheren Authentifizierung von Teilnehmern in einem ungesicherten Netzwerk auf sicheren Rechnern.

Benutzer oder Dienste

Feindliche Übernahme des Rechners ist zu vermeiden ;)

Internet, WLAN und auch LAN (Firewall schützt nicht vor Bedrohung von innen)

Fertig! ;)

Kerberos - Vorteile

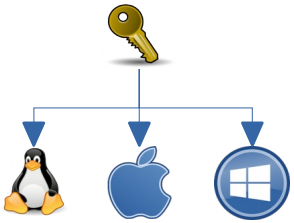
- Ermöglicht verschlüsselte Kommunikation (z.B. genutzt von NFSv4)

→ Session Keys (SK)

- Ermöglicht echtes “Single Sign On” (nicht nur Web)

→ Ticket Granting Tickets (TGT)

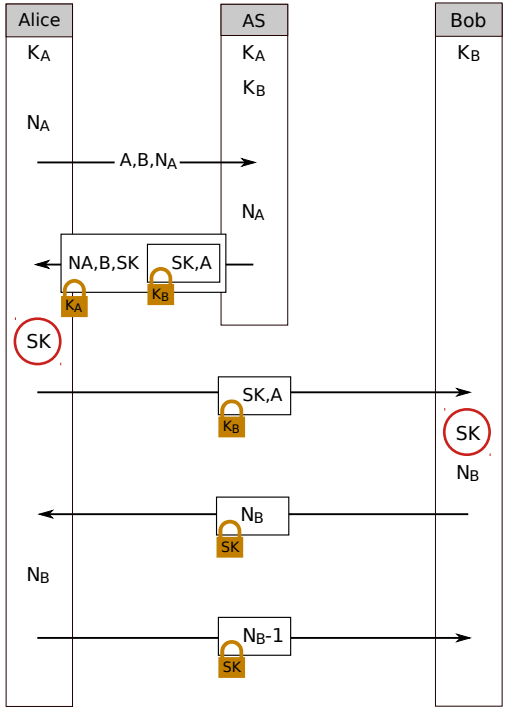
- Funktioniert auch über OS und Domain-Grenzen hinweg



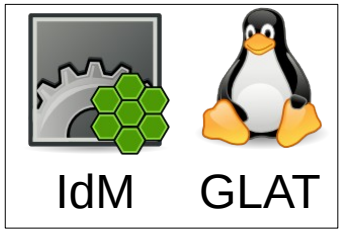
→ Cross-Realm Trusts

Symmetrisches Needham-Schroeder-Protokoll

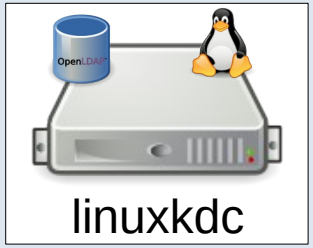
- Protokoll für sicheren Datenaustausch in einem dezentralen Netzwerk
- Zentrales Element: **Trusted Third Party**
  Die Teilnehmer der Kommunikation müssen über einen geheimen Schlüssel (**pre-shared secret**) mit einem AuthenticationService (**AS**) verfügen.
- Durch Nachweis, dass A K_A und B K_B besitzt, werden A und B gegeneinander authentifiziert und ein **SessionKey** (**SK**) ausgetauscht.



Kerberos – beteiligte Entitäten



Zentrale Quellsysteme



Authentication Service (AS)

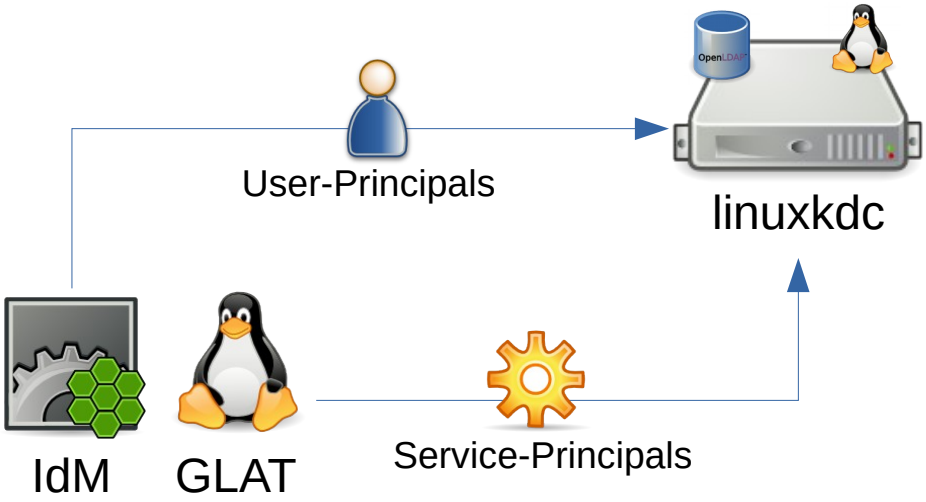
User/Client

Server/Services

Realm
LINUX.FAU.DE

Teilnehmer/Kommunikationspartner

Kerberos – Principals: Provisionierung



Principal	Shared-Secret
unrza249@LINUX.FAU.DE	Passwort (IdM)
host/linux6.rrze.fau.de@LINUX.FAU.DE	Passwort (Zufall)

Kerberos – Principals: Beispiel

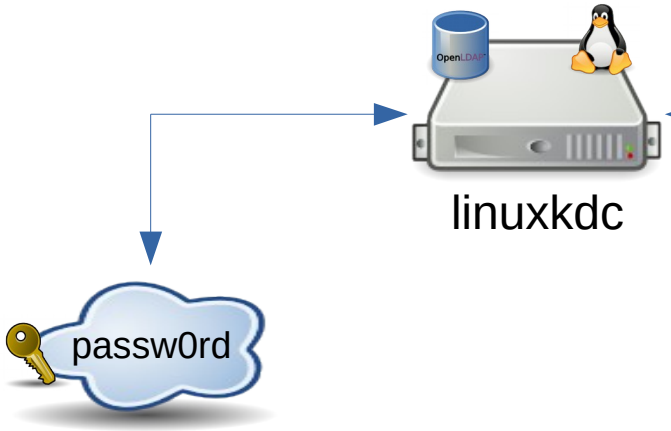
```
root@linuxkdc-master:~# kadmin.local
Authenticating as principal root/admin@LINUX.FAU.DE with password.
kadmin.local:  getprinc [redacted]
Principal: [redacted]@LINUX.FAU.DE
Expiration date: [never]
Last password change: Tue Nov 21 17:13:52 CET 2017
Password expiration date: [none]
Maximum ticket life: 1 day 00:00:00
Maximum renewable life: 7 days 00:00:00
Last modified: Tue Nov 21 17:13:52 CET 2017 (kadmind@LINUX.FAU.DE)
Last successful authentication: [never]
Last failed authentication: [never]
Failed password attempts: 0
Number of keys: 4
Key: vno 64, aes256-cts-hmac-sha1-96
Key: vno 64, aes128-cts-hmac-sha1-96
Key: vno 64, des3-cbc-sha1
Key: vno 64, arcfour-hmac
MKey: vno 1
Attributes:
Policy: [none]
```

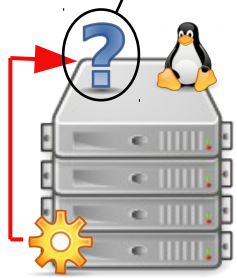
Name

Ticketlaufzeit

Schlüssel (generiert aus dem Passwort)

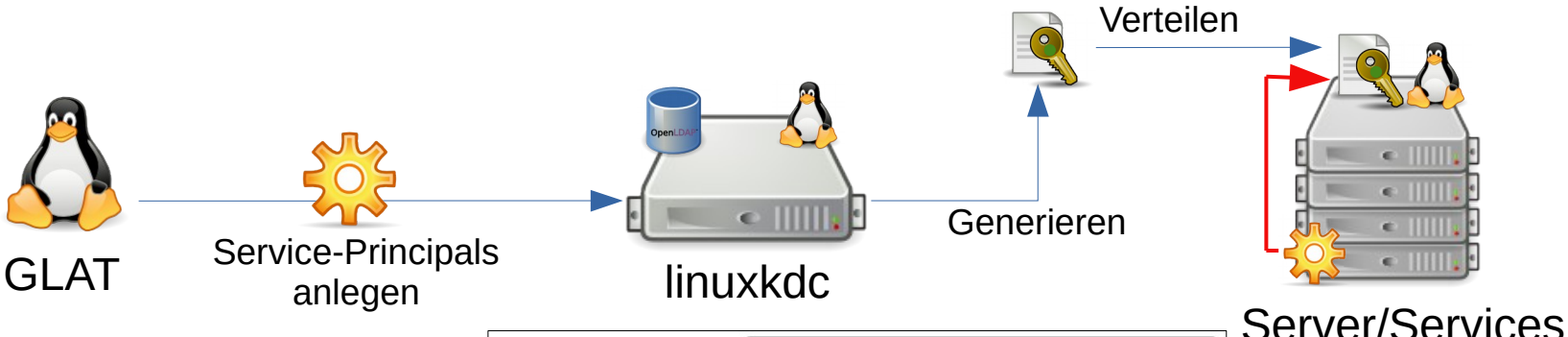
Kerberos – Shared Secrets





Keytab - Export in der kryptographischen Schlüssel des Principals in eine Datei und Verteilen and die Server/Services

Kerberos – Shared Secrets: Keytabs




```
root@linux-proxy[ ~ ]# klist -kte
Keytab name: FILE:/etc/krb5.keytab
KVNO Timestamp           Principal
---- ------------------- ------------------------------------------------------
   2 11/09/2017 10:24:49 host/linux-proxy.rrze.uni-erlangen.de@LINUX.FAU.DE (aes256-cts-hmac-sha1-96)
   2 11/09/2017 10:24:49 host/linux-proxy.rrze.uni-erlangen.de@LINUX.FAU.DE (aes128-cts-hmac-sha1-96)
   2 11/09/2017 10:24:49 host/linux-proxy.rrze.uni-erlangen.de@LINUX.FAU.DE (des3-cbc-sha1)
   2 11/09/2017 10:24:49 host/linux-proxy.rrze.uni-erlangen.de@LINUX.FAU.DE (arcfour-hmac)
   2 11/09/2017 10:24:49 nfs/linux-proxy.rrze.uni-erlangen.de@LINUX.FAU.DE (aes256-cts-hmac-sha1-96)
   2 11/09/2017 10:24:49 nfs/linux-proxy.rrze.uni-erlangen.de@LINUX.FAU.DE (aes128-cts-hmac-sha1-96)
   2 11/09/2017 10:24:49 nfs/linux-proxy.rrze.uni-erlangen.de@LINUX.FAU.DE (des3-cbc-sha1)
   2 11/09/2017 10:24:49 nfs/linux-proxy.rrze.uni-erlangen.de@LINUX.FAU.DE (arcfour-hmac)
```

Kerberos – TicketCache: ServiceTickets / TGT

TGT

ServiceTicket

TicketCache

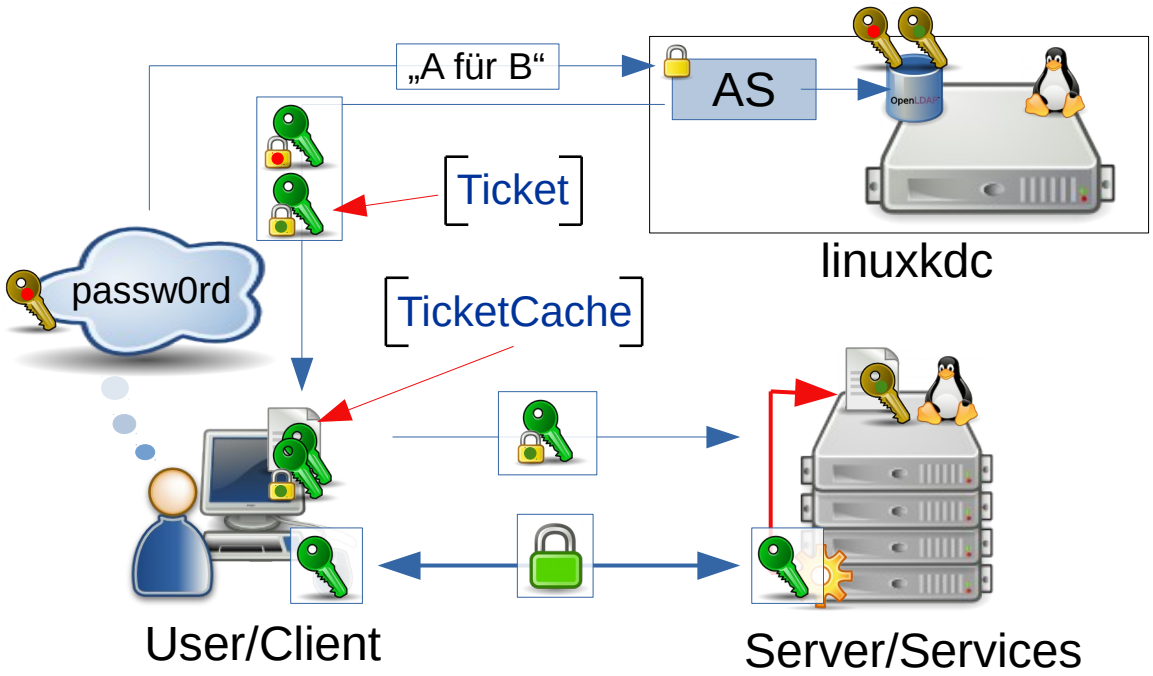
unverschlüsselter SessionKey

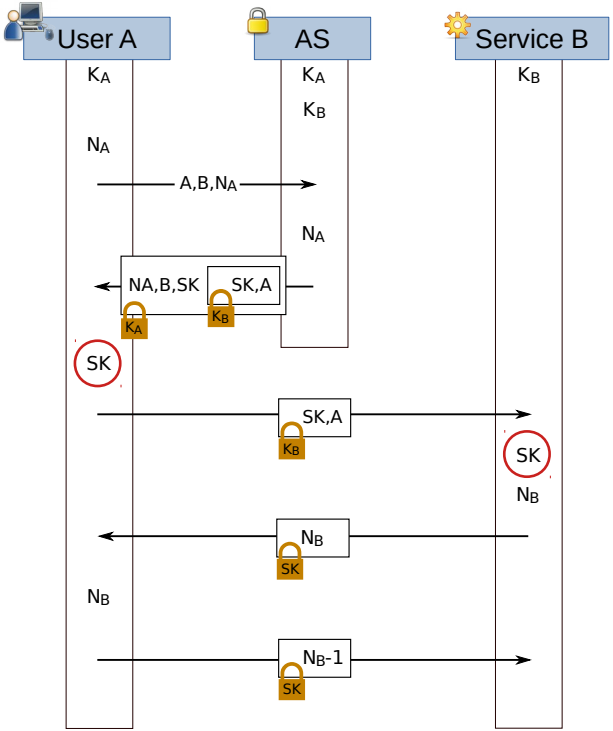
verschlüsselter SessionKey
(mit Schlüssel des Zieles)

```
unrza249@linux6[ ~ ]$ klist
Ticket cache: FILE:/tmp/krb5cc_12186
Default principal: unrza249@LINUX.FAU.DE

Valid starting       Expires              Service principal
15.07.2020 12:43:13  15.07.2020 22:43:13  krbtgt/LINUX.FAU.DE@LINUX.FAU.DE
15.07.2020 12:43:23  15.07.2020 22:43:13  host/dialog.rrze.uni-erlangen.de@LINUX.FAU.DE
```

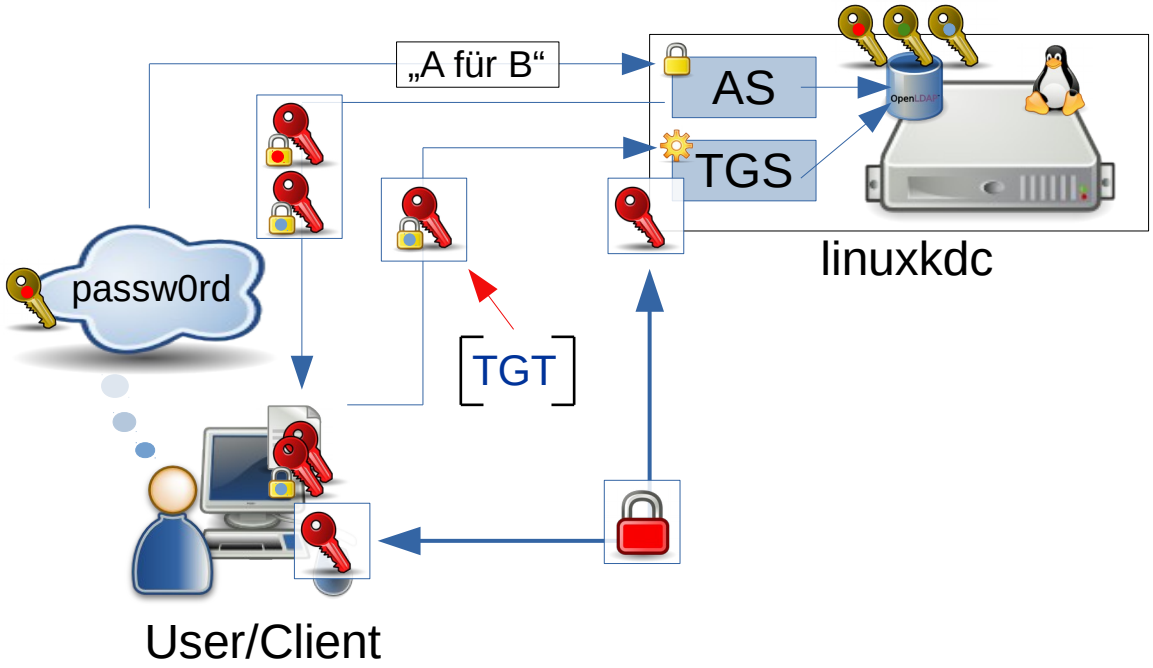
Kerberos – Protokoll: Ablauf

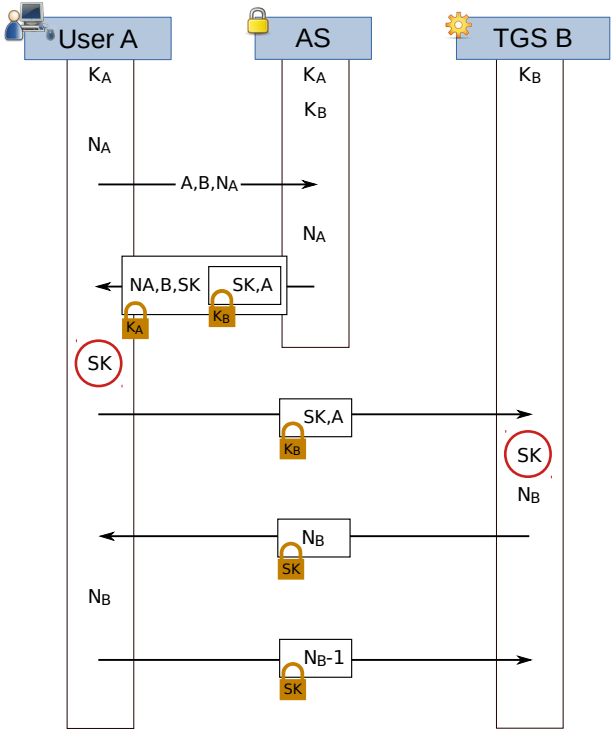




= SessionKey = SharedSecret

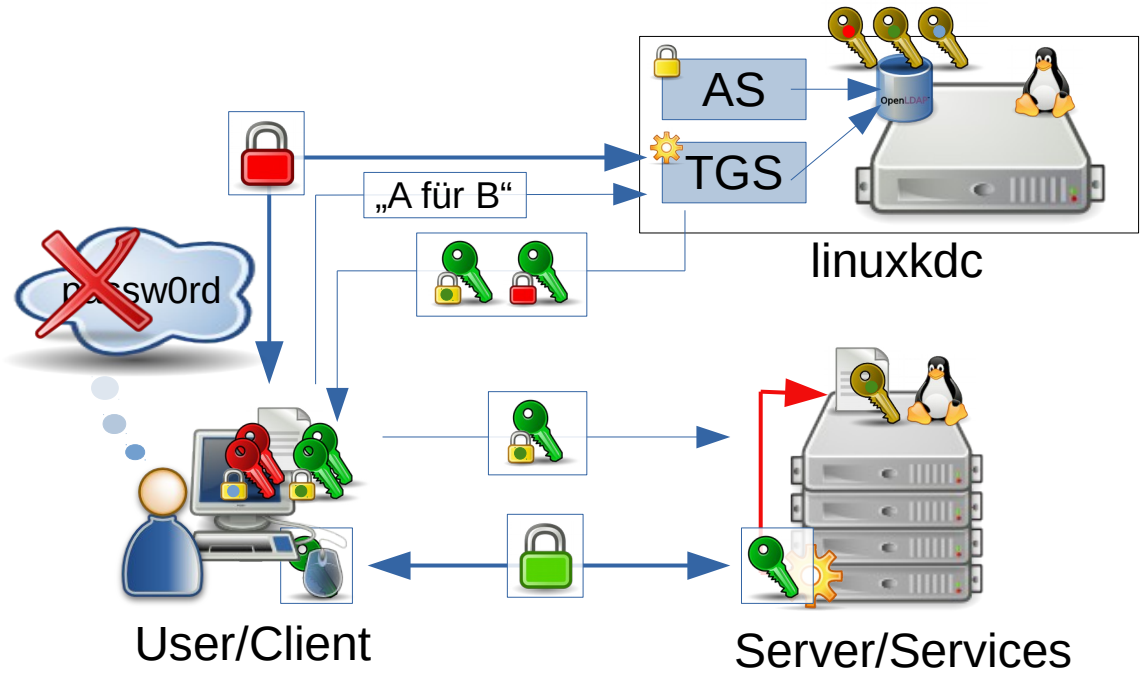
Kerberos – Protokoll: SSO?

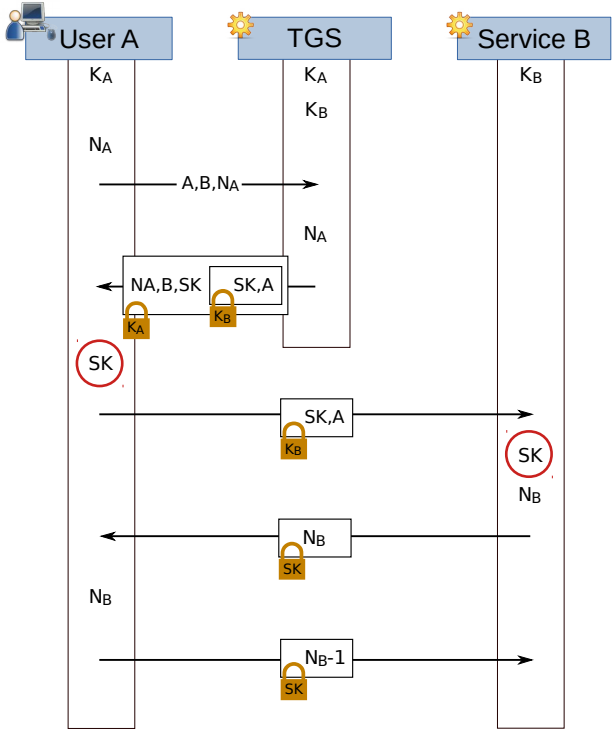




= SessionKey/TGT = SharedSecret

Kerberos – Protokoll: SSO!





= TGT (SK) = Service (SK) = SharedSecret

Kerberos – Authentication vs. Authorization

- Kerberos liefert nur **Authentifizierung**
  (und kann LDAP in diesem Punkt ersetzen)
- **Authorisierung** muss anders gelöst werden!
- Viele bestehende Systeme nutzen gruppenbasierte Authorisierung eines bestehenden Verzeichnisdienstes
- Deshalb zusätzliche LDAP-Anbindung meistens sinnvoll

Kerberos - Mapping-Probleme

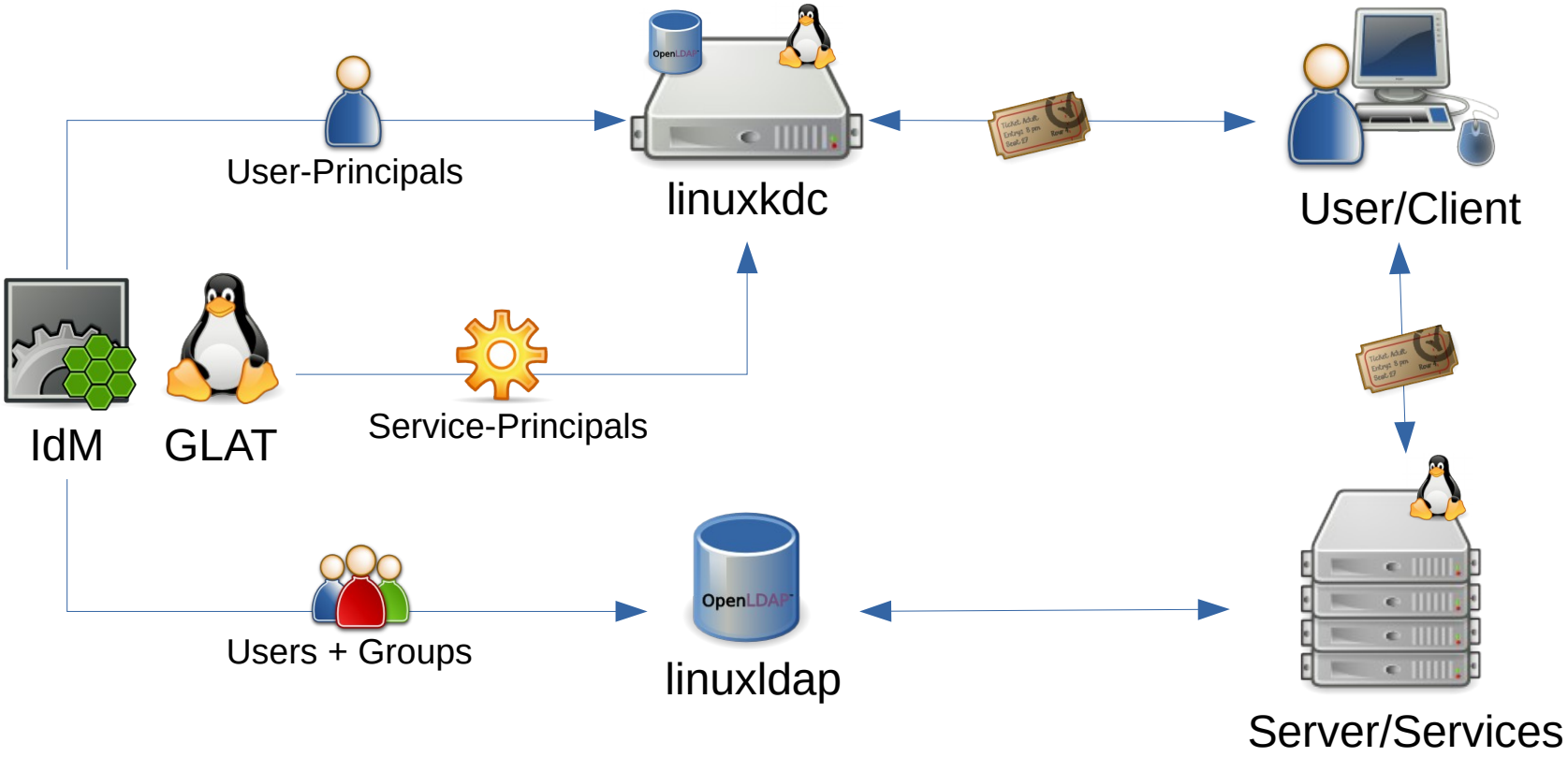
- Woher weiß der Client A welcher KDC (AS) für die Authentifizierung mit Service B kontaktiert werden muss?

  Lösung: **Domain-Realm Mapping** (File oder DNS – TODO ;)

- Woher weiß ein Dienst welcher Benutzer im LDAP welchem Principal entspricht?

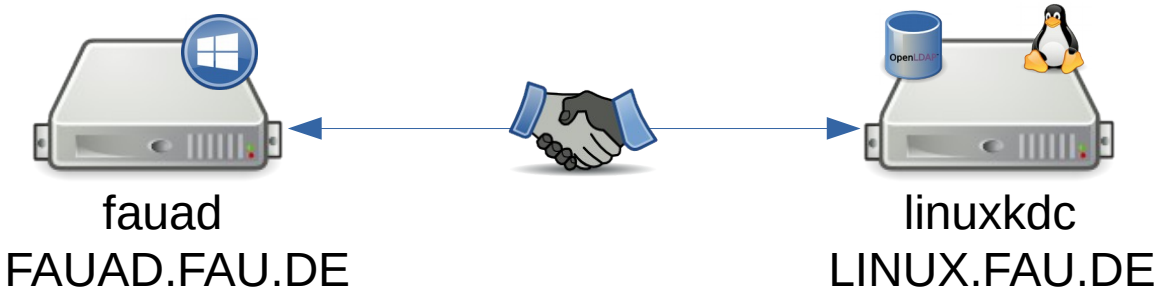
  Lösung: **Principal-Id-Mapping**

Kerberos/LDAP Infrastruktur (Linux@RRZE)



Kerberos - Cross-Realm Trust

- Windows/Mac am RRZE nutzen Windows Active-Directory (AD) Infrastruktur
- Anderer Realm: **FAUAD.FAU.DE**
- Bidirektionaler Trust mit **LINUX.FAU.DE** Realm



Vielen Dank für Ihre Aufmerksamkeit!

Regionales Rechenzentrum Erlangen [RRZE]
Martensstraße 1, 91058 Erlangen
www.rrze.fau.de