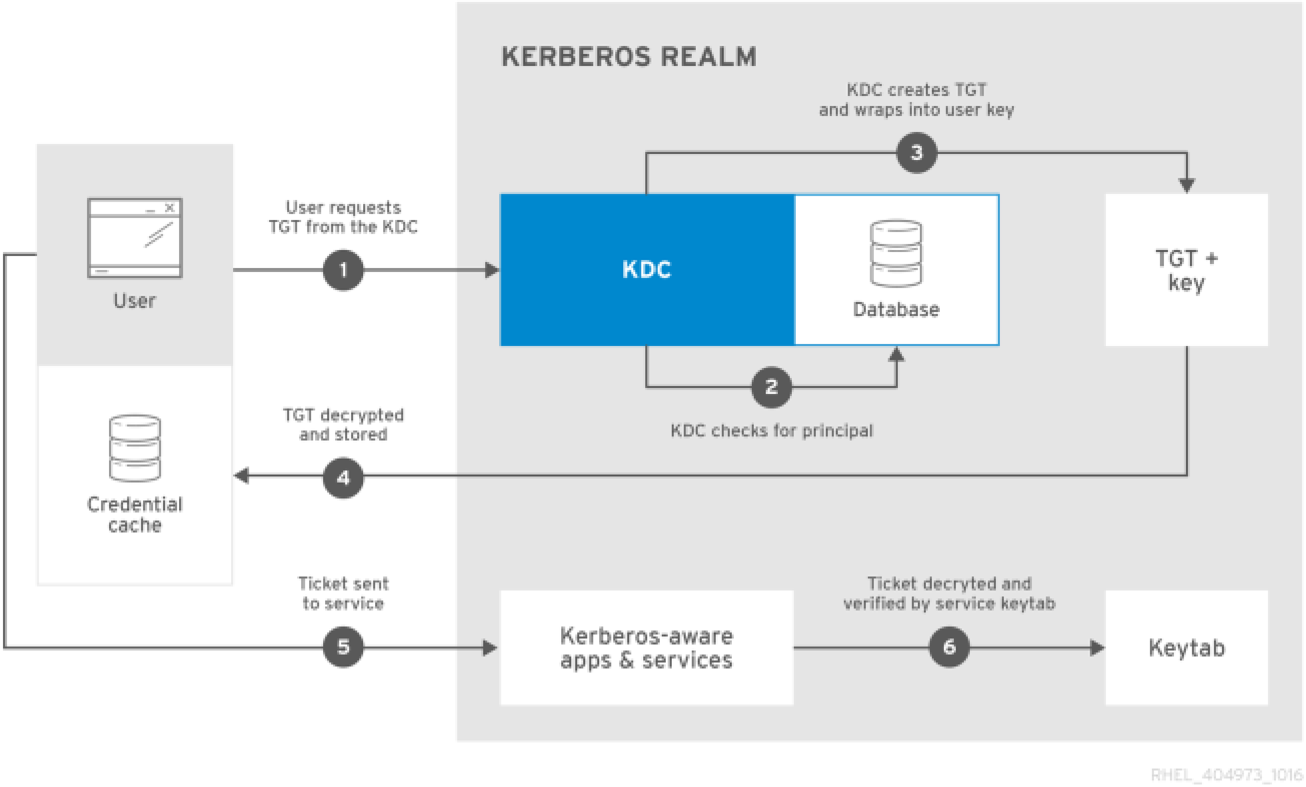
Kerberos unter Redhat / CentOS

Kerberos itself is a **network protocol** that enables **authentication for users** of client/server applications through the use of **secret-key cryptography**.



Kerberos is usually used **for authenticating desktop users on networks**, but through the use of **some additional tools**, it can be used to **authenticate users to web applications** and to provide **SSO** for a set of web applications. This essentially allows users who have already authenticated on their desktop network to seamlessly access secured resources in web applications without having to re-authenticate. **This concept is known as Desktop-Based SSO** since the user is being authenticated via a desktop-based authentication mechanism, and their authentication token or ticket is being used by the web application as well. *This differs from other SSO mechanisms such as Browser-Based SSO, which authenticates users and issues tokens all via the browser.*

The Kerberos protocol defines several components that it uses in authentication and authorization:

See → KERBEROS COMPONENTS

Einrichten von OpenLDAP

1. Install the **basic LDAP server** installation, type the following at a shell prompt:

```
# yum install openldap openldap-clients openldap-servers
```

Fix Me!

Einrichten von Kerberos

ACHTUNG! Bevor noch überhaupt irgendwie, mit dem Einrichten des Kerberos begonnen werden kann, müssen zuerst folgende Voraussetzungen zwingend erfüllt werden:

- **DNS Auflösung** - Muss für alle Server/Clients funktionieren! (**A** und **PTR** Recorts/Auflösung)
- **Die NTP Zeitsynchronisierung** - Muss in jedem Fall, auf dem zukünftigen KDC eingerichtet sein!

- https://access.redhat.com/solutions/46681
- https://access.redhat.com/solutions/1365423

Konfiguration Master KDC-Server

1. **Install the required packages for the KDC:**

   ```
   # yum install krb5-server krb5-libs krb5-workstation
   ```

2. **Edit** the **/etc/krb5.conf** **and** **/var/kerberos/krb5kdc/kdc.conf** configuration files to reflect the realm name and domain-to-realm mappings. For example:

   ```
   [logging]
   default = FILE:/var/log/krb5libs.log
   kdc = FILE:/var/log/krb5kdc.log
   admin_server = FILE:/var/log/kadmind.log

   [libdefaults]
   default_realm = EXAMPLE.COM
   dns_lookup_realm = false
   dns_lookup_kdc = false
   ticket_lifetime = 24h
   renew_lifetime = 7d
   forwardable = true
   allow_weak_crypto = true

   [realms]
   EXAMPLE.COM = {
   kdc = kdc.example.com.:88
   admin_server = kdc.example.com
   ```

```
default_domain = example.com
}

[domain_realm]
.example.com = EXAMPLE.COM
example.com = EXAMPLE.COM
```

A simple realm can be constructed by replacing instances of EXAMPLE.COM and example.com with the correct domain name — being certain to keep **uppercase** and **lowercase** names **in the correct format** — *and by changing the KDC from kerberos.example.com to the name of the Kerberos server*. **By convention, all realm names are uppercase and all DNS host names and domain names are lowercase.**

3. **Create the database** using the kdb5_util utility:

   ```
   # kdb5_util create -s
   ```

   The **create** command creates the database that stores keys for the Kerberos realm. The **-s argument** creates a **stash file** in which the master server key is stored. If no stash file is present from which to read the key, the Kerberos server (krb5kdc) prompts the user for the master server password *(which can be used to regenerate the key)* every time it starts.

4. **Edit** the **/var/kerberos/krb5kdc/kadm5.acl** file. This file is used by **kadmind** to determine which principals have administrative access to the Kerberos database and their level of access. For example:

   ```
   */admin@EXAMPLE.COM *
   ```

   Most users are represented in the database by a single principal (*with a NULL, or empty, instance, such as joe@EXAMPLE.COM*). **In this configuration, users with a second principal with an instance of admin** (for example, **joe/admin@EXAMPLE.COM**) **are able to exert full administrative control over the realm's Kerberos database.** After **kadmind** has been started on the server, any user can access its services by running **kadmin** on any of the clients or servers in the realm. However, only users listed in the **kadm5.acl** file can modify the database in any way, except for changing their own passwords.

5. **Create the first principal** using **kadmin.local** at the KDC terminal: ……
   Fix Me!

- https://www.rootusers.com/how-to-configure-linux-to-authenticate-using-kerberos/
- https://www.theurbanpenguin.com/configuring-a-centos-7-kerberos-kdc/
- https://gist.github.com/ashrithr/4767927948eca70845db
- https://www.youtube.com/watch?v=yS5mLBh-yGo

Konfiguration Kerberos Client

Fix Me!

Weiteres

- https://www.tecmint.com/setting-up-nfs-server-with-kerberos-based-authentication/

Redhat Dokumentation zum Thema

red_hat_enterprise_linux-7-system-level_authentication_guide-en-us.pdf

Last update: **2017/10/27 13:30**