Kerberos

Markus Schade

Agenda

- Einleitung
- Authentifizierung
- Protokoll
- Implementierungen
- Anwendungen

Vertraust Du mir?

Oder mir?

Historie

- 1988 am MIT im Rahmen des Athena Projekts entwickelt
- Client/Server-Architektur
  - Zentraler Authentifizierung
  - Trusted 3rd Partry
- Setzt auf symmetrische Verschlüsselung
- Gegenseitige Authentifizierung (Needham-Schroeder)
- Ermöglicht Single-Sign-On

Features

- Sicher
  - kein Identitätsdiebstahl durch Abfangen, Verändern oder Replay von Paketen
- Zuverlässig
  - Redundante Auslegung möglich
- Transparent
  - Im Idealfall unsichtbar für den Nutzer
- Skalierbar
  - Verteilte Architektur
  - Delegation

Steckbrief

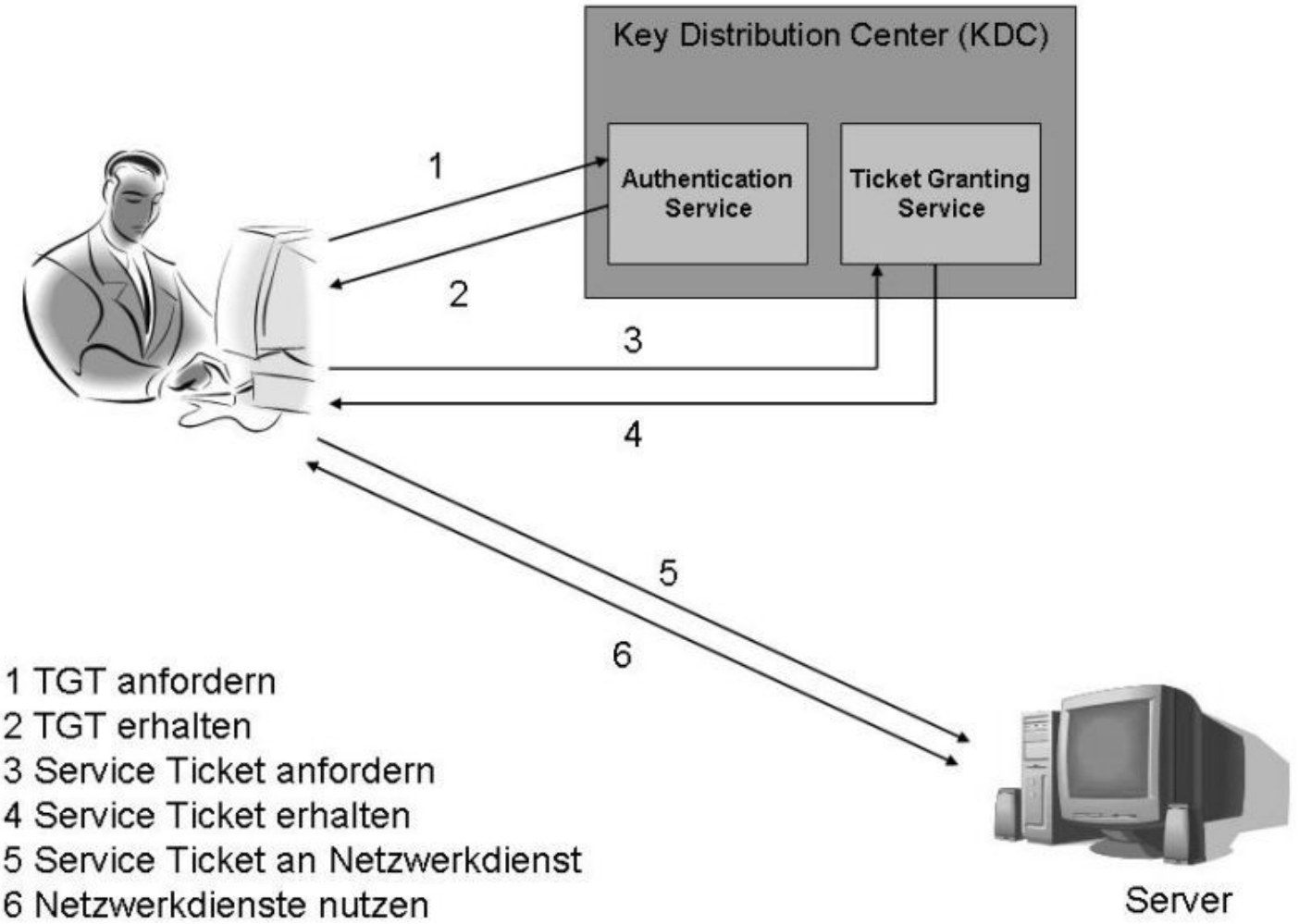
- Trusted third party: Key Distribution Center (KDC)
  - Authentication Server
  - Ticket Granting Server
- 2 Protokolle
  - v4: nutzt (3)DES und wird nicht mehr verwendet
  - v5: aktueller (modularer) Standard
    - RFC1510 (1993), RFC4120 (2005)
    - Mehr Sicherheit (Replay Schutz, Pre-Authentication)
- ABER:
  - keiner oder nur implizite Autorisierung (in Arbeit: PAD)
  - Keine Gruppen

Angriffe auf Netzwerkauthentifizierung

- Abfangen von Paketen / Man in the Middle
- Gefälschte/Modifizierte Pakete
- Replay von Paketen
- (DNS) Spoofing

Begriffe

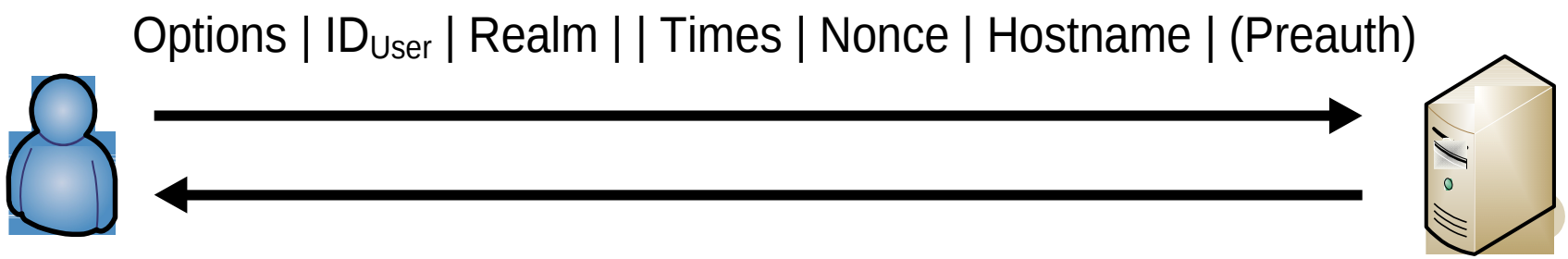
- Principal – eine eindeutige Entität (Nutzer oder Dienst)
  - primary/instance@REALM
  - z.B.: user/admin@EXAMPLE.COM, http/hostname@EXAMPLE.COM
- Instance – optionaler String zur Qualifizierung
- Realm – administrativer Bereich (i.d.R.= Domain)
- Keytab – eine Datei mit dem shared-secret eines Principal
- Ticket – verschlüsselter, zeitbegrenzter Ausweis für Dienst
- Ticket Granting Ticket – Ausweis für Service Tickets



1 TGT anfordern
2 TGT erhalten
3 Service Ticket anfordern
4 Service Ticket erhalten
5 Service Ticket an Netzwerkdienst
6 Netzwerkdienste nutzen

Ticket Granting Ticket

TGT Request



TGT – Schritt 1

- Anfrage Ticket Granting Ticket (TGT)
  - `foo@REALM, krbtgt@REALM, Hostname(s), Lifetime`
- Pre-Auth (v5)
  - $[timestamp]_{user}$
  - Schutz vor Offline Wörterbuchattacken
- Nonce
  - Zufallszahl, verschlüsselt im TGT
  - Schutz vor Replay-Angriffen

Schritt 2

- KDC sendet Ticket Granting Ticket und Session Key
- TGT: $[\texttt{ClientPrinc, Hostname(s), Timestamp, Lifetime, SessKey1}]_{TGS}$
- TGT ist verschlüsselt, für den Nutzer nicht lesbar
- Angreifter könnte TGT abfangen, aber:
  - Kann Session-Key nicht entschlüsseln
  - Kann damit Nonce-Challenge des Service nicht lösen

Implementationen

- MIT
  - Ursprüngliche (und ehemals einzige) Implementierung
  - US-Exportverbot bis 2001/2003
  - Master/Slave KDCs
- Heimdal
  - Reimplementierung zur Umgehung des Verbots
  - Multi-Master KDC
- Microsoft Active Directory
  - Basiert auf MIT
  - inkompatible Erweiterungen
    - PAC (Privilege Attribute Certificate)

Anwendungen

- GSSAPI – Authentifizierungsframework
  - „generisch“ - aber hautpsächlich Kerberos

- (fast) jeder Dienst, der Nutzer authentifiziert
  - SSH
  - LDAP
  - HTTP
  - FTP
  - NFS
  - Subversion
  - etc.

Client HowTo

- kinit – Anfordern eines Ticket Granting Tickets
  - -R – erneuern eines noch gültigen Tickets
  - -k <file> - Ticket via Keytab
- klist – Übersicht der Tickets im Cache
- ksu – kerberized su
- kdestroy – Tickets verwerfen
- kpasswd – Password ändern

Server Quickstart

- Kerberos DB anlegen
  - `kdb5_util -r FOO.BAR create -s`
  - `.k5.FOO.BAR –enthält KDC DB master key`
  - `kadm.keytab –enthält kadmin/* principals`
- In ACL */admin DB-Änderungen erlauben
  - `*/admin@FOO.BAR *`
- Principals via `kadmin.local` hinzufügen
  - `kadmin: addprinc john/admin`
  - `kadmin: addprinc john`
- `/etc/kdc.conf` oder DNS anpassen
- KDC starten und loslegen

Server HowTo

- Service Principals anlegen
  - `kadmin: addprinc -randkey host/baz.foo.bar`
  - `kadmin: addprinc http/www.foo.bar`
- Export in Keytab
  - `Kadmin: ktadd -k /tmp/baz.tab \ host/baz.foo.bar`
- Sicherer Transfer der Keytab auf Server
  - sshd_config: `GSSAPIAuthentication yes`
    - System-Keytab:`/etc/krb5.keytab`
  - mod_auth_kerb: `Krb5Keytab /path/to/keytab`
    - Firefox: network.negotiate-auth.trusted-uris

Bugs / TODO / Wishlist

- mod_auth_kerb kaputt in Apache 2.2
- SSH: PrivilegeSeparation: sandbox → yes
- Gruppen / PAD (RFC Draft)

Fragen? Fragen!

Vielen Dank und

viel Spaß noch!

HETZNER ONLINE

Wir suchen Mitarbeiter!

- Softwareentwickler
- System Engineer
- IT-Servicetechniker
- Abschlußarbeiten / Praktika / Ferienarbeit

mehr unter *https://jobs.hetzner.de*

WE LOVE BITS. DO YOU?

DAS UNTERNEHMEN

Hetzner Online ist ein professioneller Webhosting-Dienstleister und erfahrener Rechenzentrenbetreiber. Wir bieten Lösungen an, die durch Qualität, Stand der Technik und Sicherheit überzeugen. Dabei reicht das Angebot für Homepage-Einsteiger bis zum professionellem Webentwickler:

- Root, Managed und vServer
- Colocation
- Shared Hosting
- Internet Domains
- SSL-Zertifikate