redhat

Kerberos

Zbyšek MRÁZ

QE BaseOS Security

About

- principles
- usage & application
- ticket flow

NotAbout

- writing applications
- cryptography
- number theories

WhatIs

- MIT developed
- implementations
  - MIT krb5
  - Heimdal
  - Windows AD

WhatIs #2

- AUTHENTICATION
  - act of verifying identity
  - process of proving identity to service
- Benefits
  - Standards based strong authentication system
    - AES-256 -128, arcfour, 3DES
  - Various OS support
  - No password sent thru the network
  - Single sign-on

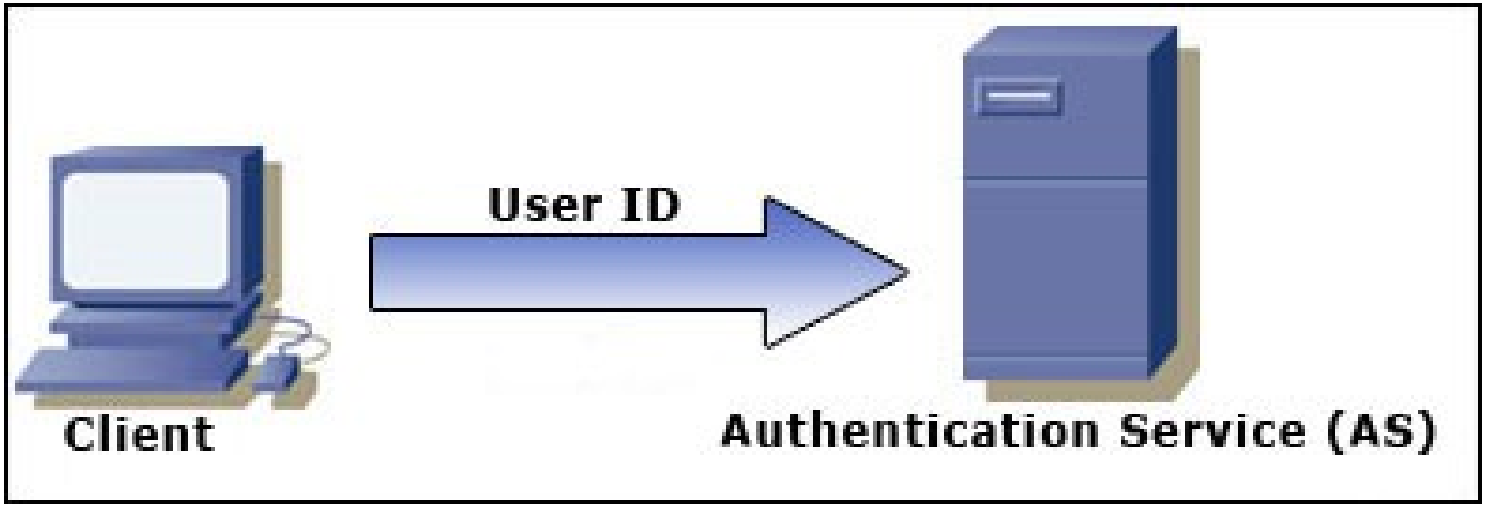
WorksWith (included in MIT implementation)

- eklogin
- gssftp
- krsh
- kshell
- ksu
- telnet

WorksWith (supported)

- ssh
- httpd + modules (perl, python, ruby)
- nfs
- versioning (git, svn, cvs)
- bind
- cups
- proxy
- mail (smtp, pop, imap)

HowWorks



- Client wants to prove his identity to AS
- User ID = name + realm
- AS checks the database
  - Krb (berkleydb)
  - ldap

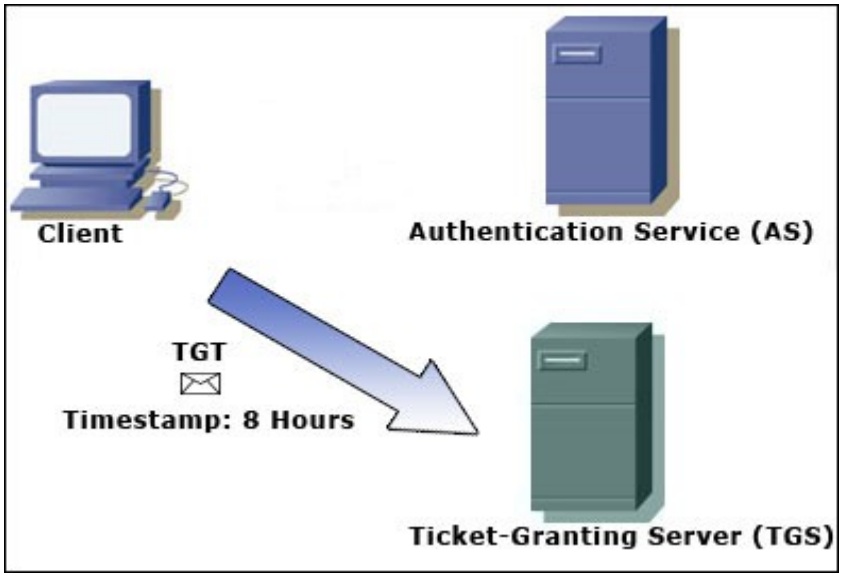
HowWorks #2

- Client enters the password to decrypt the TGT

HowWorks #3

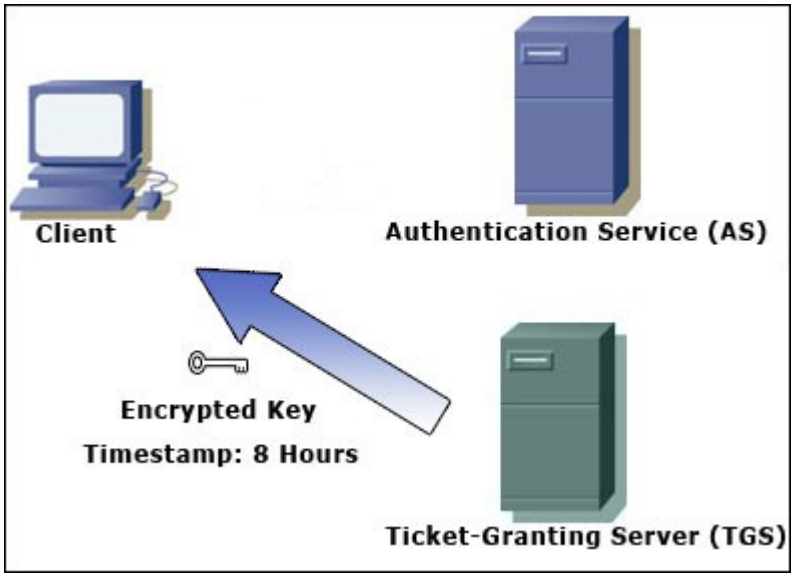
- Client ID
- Client network address
- Ticket validity period
- client/TGS session key

HowWorks #4



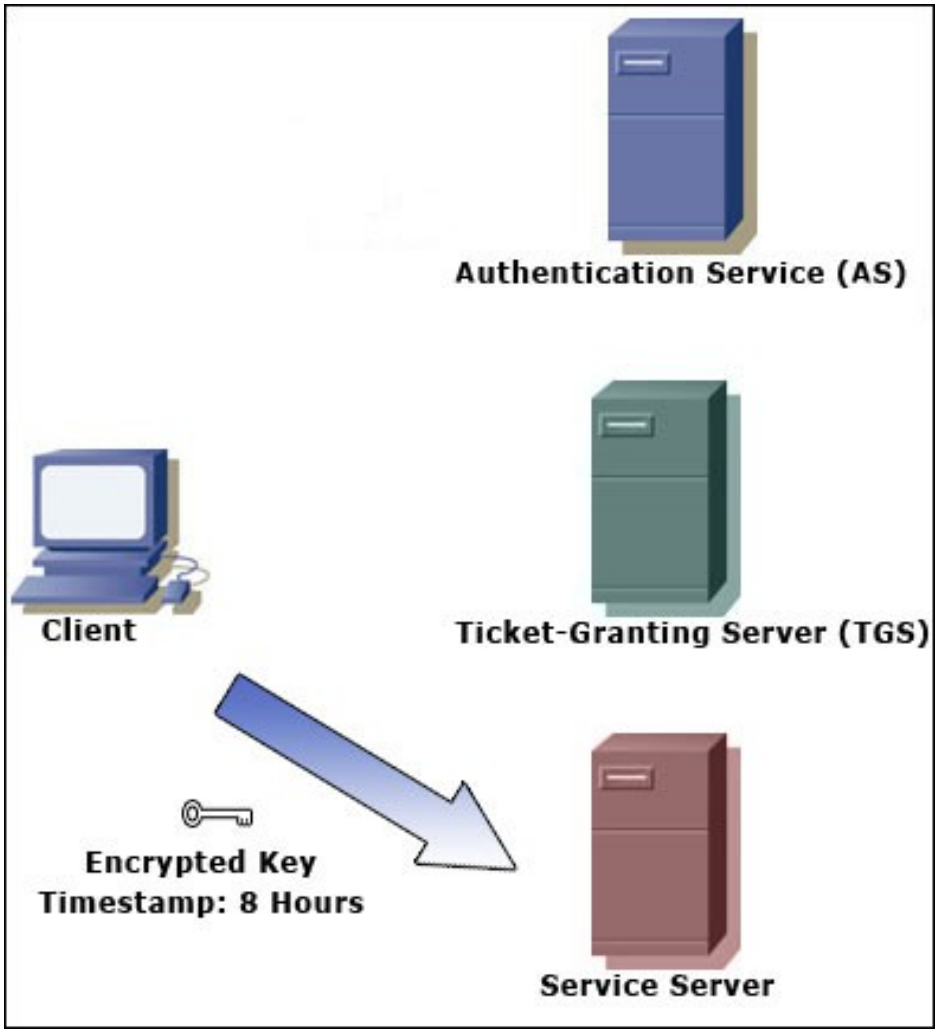
- The client submits the ticket-granting ticket mentioning the service name to the ticket-granting server (TGS), to get authenticated.
- KDC
  - key distribution center
  - KDC shares a key with each of all the other parties
  - The KDC produces a ticket based on a server key.

HowWorks #5



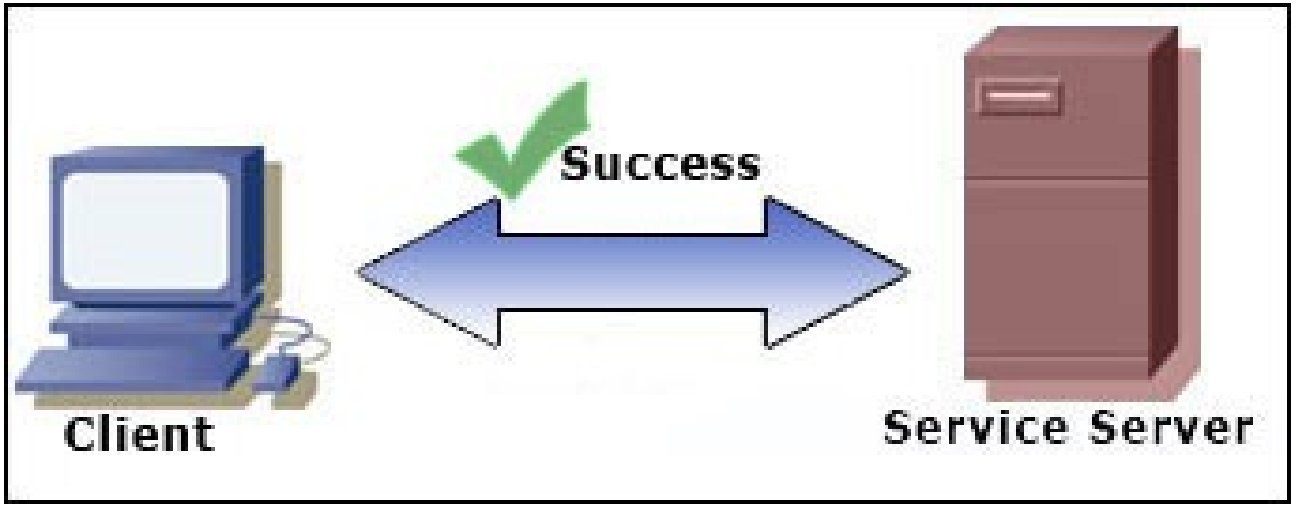
- The TGS creates an encrypted key with a timestamp, and grants the client a service ticket
- The client decrypts the ticket, tells the TGS it has done so, and then sends its own encrypted key to the service

HowWorks #6



- The service decrypts the key, and makes sure the timestamp is still valid.

HowWorks #7



- The client decrypts the ticket. If the keys are still valid, communication is initiated between client and server.
- After the communication is made between the client and server, no further need of transmitting logon information is needed. The client is authenticated until the session expires.

klist

```
[16:29:17] neo@lexi:~$ /opt/mit-krb5/klist -cafe
Ticket cache: FILE:/tmp/krb5cc_1001
Default principal: zbysek@MRAZ.COM

Valid starting     Expires            Service principal
02/08/11 16:07:59  02/09/11 02:07:59  krbtgt/MRAZ.COM@MRAZ.COM
    renew until 02/08/11 16:07:59, Flags: FRI
    Etype (skey, tkt): aes256-cts-hmac-sha1-96, aes256-cts-hmac-sha1-96
    Addresses: lexi.mraz.com, 100.20.40.50
02/08/11 16:16:42  02/09/11 02:07:59  imap/mail.mraz.com@MRAZ.COM
    Flags: T, Etype (skey, tkt): aes256-cts-hmac-sha1-96, aes256-cts-hmac-sha1-96
    Addresses: lexi.mraz.com, 100.20.40.50
02/08/11 16:28:35  02/09/11 02:07:59  HTTP/devel.mraz.com@MRAZ.COM
    Flags: T, Etype (skey, tkt): aes256-cts-hmac-sha1-96, aes256-cts-hmac-sha1-96
    Addresses: lexi.mraz.com, 100.20.40.50
02/08/11 16:29:21  02/09/11 02:07:59  host/s01.mraz.com@MRAZ.COM
    renew until 02/08/11 16:07:59, Flags: FRT
    Etype (skey, tkt): aes256-cts-hmac-sha1-96, aes256-cts-hmac-sha1-96
    Addresses: lexi.mraz.com, 100.20.40.50
02/08/11 16:35:02  02/09/11 02:07:59  cvs/cvs.devel.mraz.com@MRAZ.COM
    Flags: T, Etype (skey, tkt): aes256-cts-hmac-sha1-96, aes256-cts-hmac-sha1-96
    Addresses: lexi.mraz.com, 100.20.40.50
[16:37:22] neo@lexi:~$
```

WhatIsNot

- AUTHORIZATION

- Disadvantages
  - The only KDC handles authentication (might be solved by propagation to slaves)
  - Clock synchronization – up to 10 minutes (might be solved by proper time synchronization
  - No unencrypted service should be used

redhat

Thank you

- questions

HappyEnding