KERBEROS

Authentication Application

Overview

- Introduction
- Motivation
- Requirement
- Kerberos Version 4
- Kerberos Realms
- Kerberos V4 V/s V5
- Kerberos Version 5
- Strength
- Conclusion

Introduction

- Authentication: can be defined as determining an identity to the required level of assurance.

- Authentication Application : Deals with the authentication function that have been developed to support application-level authentication

Introduction to Kerberos

- An authentication service developed for Project Athena at MIT
- Provides
  - strong security on physically insecure network
  - a centralized authentication server which authenticates
    - Users to servers
    - Servers to users
- Relies on conventional encryption rather than public-key encryption

Why Kerberos is needed ?

Problem: Not trusted workstation to identify their users correctly in an open distributed environment

3 Threats:

- Pretending to be another user from the workstation
- Sending request from the impersonated workstation
- Replay attack to gain service or disrupt operations

Why Kerberos is needed ? Cont.

Solution:

- Building elaborate authentication protocols at each server
- A centralized authentication server (Kerberos)

Requirements for KERBEROS

- Secure:
  - An opponent does not find it to be the weak link
- Reliable:
  - The system should be able to back up another
- Transparent:
  - An user should not be aware of authentication
- Scalable:
  - The system supports large number of clients and severs

KERBEROS VERSION 4

- Version 4 is most widely used version
- Version 4 uses of DES
- Version 4 build up to the full protocol by looking at several hypothetical dialogues
- Version 5 corrects some of the security deficiencies of Version 4

- Problem:

An opponent can pretend to be another client and obtain unauthorized privileges on server machine.

- Solution :

Server must be able to confirm the identities of client who request service.

- Problem:

1. the no. of times the password should be entered should be minimized.
2. Plaintext transmission of password

- Solution :

1. Ticket-granting Server; Issues ticket to user who have been authenticated to AS
2. The client can use this ticket to request multiple service granting ticket.

- Problem:

1. Lifetime associated with ticket granting ticket
2. Requirement for servers to authenticate themselves to user.

Tickets:

- Contains information which must be considered private to the user
- Allows user to use a service or to access TGS
- Reusable for a period of particular time
- Used for distribution of keys securely

Authenticators

- Proves the client's identity
- Proves that user knows the session key
- Prevents replay attack
- Used only once and has a very short life time
- One authenticator is typically built per session of use of a service

Kerberos Overview

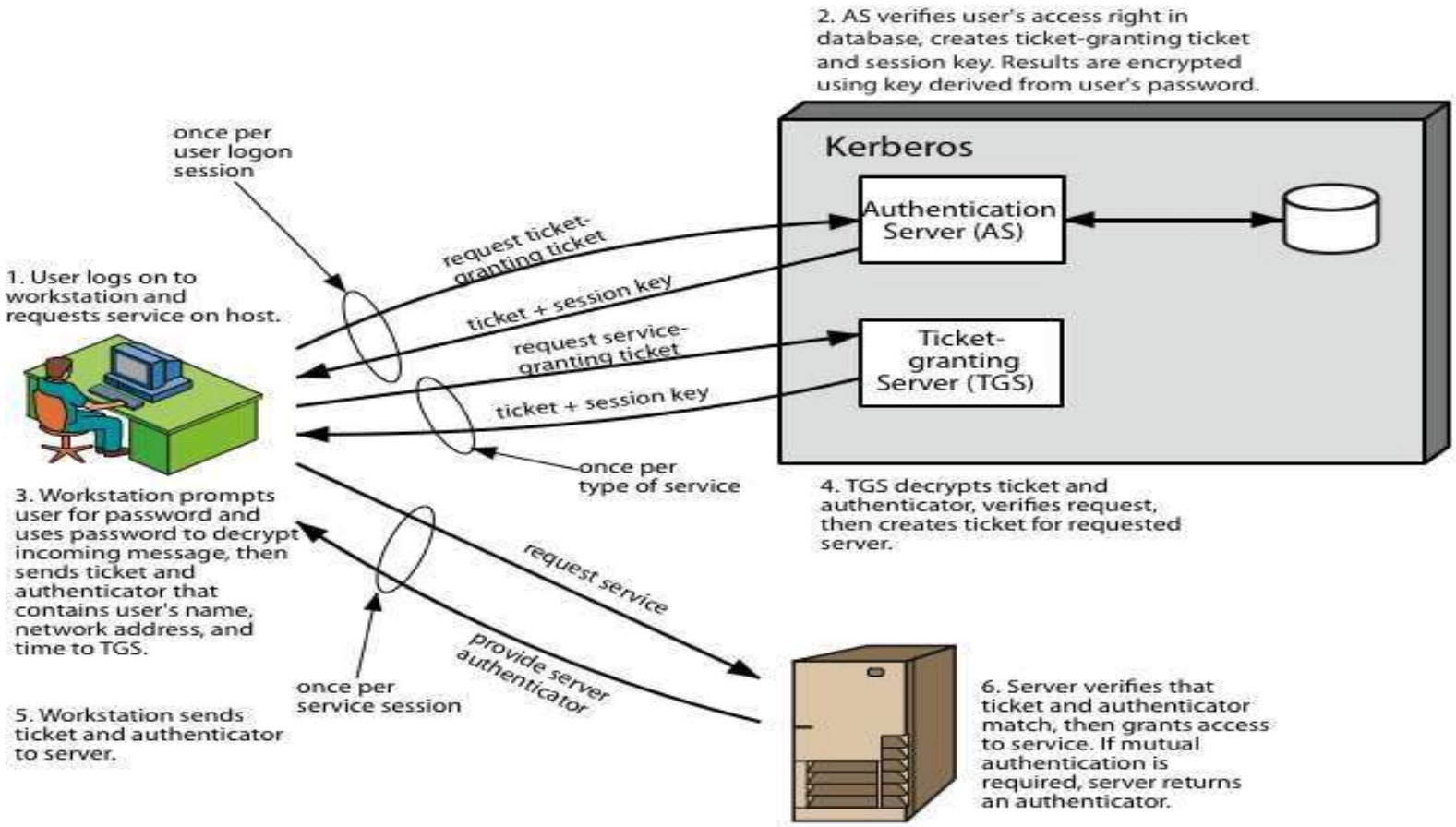


Figure 14.1 Overview of Kerberos

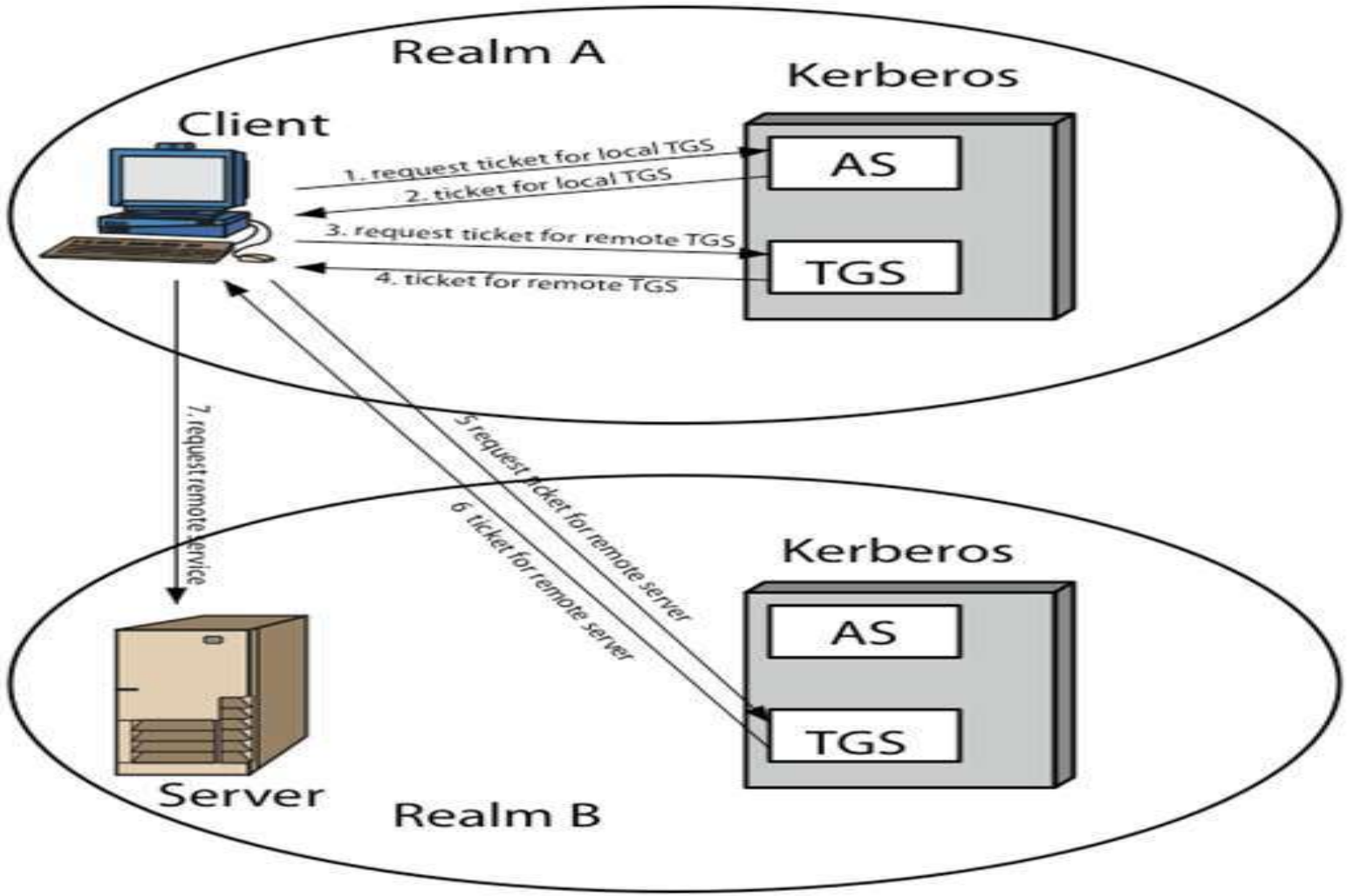
Kerberos Realms

- A single administrative domain includes:
  - a Kerberos server
  - a number of clients, all registered with server
  - application servers, sharing keys with server
- What will happen when users in one realm need access to service from other realms?:
  - Kerberos provide inter-realm authentication

Inter-realm Authentication:

- Kerberos server in each realm shares a secret key with other realms.
- It requires
  - Kerberos server in one realm should trust the one in other realm to authenticate its users
  - The second also trusts the Kerberos server in the first realm

Request for Service in another realm:



KERBEROS Version 5 versus Version4

- Environmental shortcomings of Version 4:
  - Encryption system dependence: DES
  - Message byte ordering
  - Internet protocol dependence
  - Ticket lifetime
  - Authentication forwarding
  - Inter-realm authentication

KERBEROS Version 5 versus Version4

- Technical deficiencies of Version 4:
  - Double encryption
  - Session Keys
  - Password attack
  - Mode of Encryption

New Elements in Kerberos Version 5

- Realm
  - Indicates realm of the user
- Options
- Times
  - From: the desired start time for the ticket
  - Till: the requested expiration time
  - Rtime: requested renew-till time
- Nonce
  - A random value to assure the response is fresh

Kerberos Version 5 Message Exchange:1

- <u>To obtain ticket-granting ticket:</u>

(1) C → AS : Options || IDc || Realmc || IDtgs ||Times || Nonce1

(2) AS → C : Realmc || IDc || Ticket tgs ||
EKc [Kc,tgs || IDtgs || Times || Nonce1 ||| Realm tgs]

Ticket tgs= **EKtgs** [Flags || Kc,tgs || Realm c ||
IDc || ADc || Times]

Kerberos Version 5 Message Exchange:2

- To obtain service-granting ticket :

(3) **C → TGS :** Options || IDv || Times || Nonce2 || Ticket tgs ‖ Authenticator c

(4) **TGS → C :** Realmc || IDc || Ticket v || **EK c,tgs** [Kc,v ‖ Times || Nonce2 || IDv ‖ Realm v]

Ticket tgs= **EKtgs** [Flags || Kc,tgs || Realm c || IDc || ADc || Times]

Ticket v : **EK v** [Kc,,v ‖ Realmc || IDc ‖ ADc ‖ Times]

Authenticator c : **EK c,tgs** [IDc ‖ Realmc ‖ TS1]

Kerberos Version 5 Message Exchange:3

- To obtain service

(5) C → S : Options || Ticket v|| Authenticator c

(6) S → C : **EK c,v** [TS2|| Subkey || Seq#]

- **Ticket v** : **EK v** [Flags || Kc,v || Realmc || IDc || ADc || Times]
- **Authenticator c** : **EK c,v** [IDc || Realmc || TS2 || Subkey|| Seq#]

Kerberos : Strengths

- User's passwords are never sent across the network, encrypted or in plain text
- Secret keys are *only* passed across the network in encrypted form
- Client and server systems mutually authenticate
- It limits the duration of their users' authentication.
- Authentications are **reusable** and **durable**

Conclusion

- Kerberos is an authentication service using convention encryption
- Kerberos the solution to network security is a protocol designed to provide centralized authentication whose function is to authenticate user to server and server to user.