IK2206 – Internet Security and Privacy

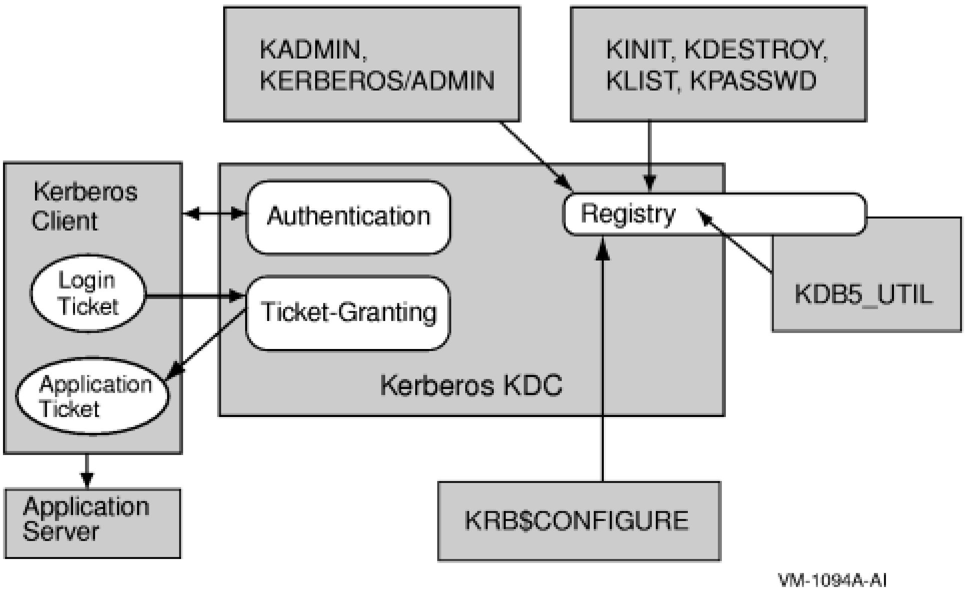
Chapter 13 – Kerberos V4

13.1 Introduction

- What were the design goals of Kerberos?

  - No cleartext passwords on the network
  - No client passwords on servers
  - Minimize password exposure on workstation
  - Compromise only impacts one client/server/user
  - Minimize the need for the password (single sign-on)

- What are the main components of Kerberos?



http://h71000.www7.hp.com/doc/83final/ba554_90008/ch01s03.html

- Kerberos comes from project Athena at MIT, which ended in 1991. Consider how Kerberos was influenced by how computers and computer systems were used in the 1980s?

  IS THIS A QUESTION???

13.2 Tickets and Ticket-Granting Tickets

- What is a ticket?

  A ticket contains authentication information such as the name and the session key that is shared between two entities. If Alice gets a ticket from KDC she won't be able to read it since the ticket is encrypted with Bob's private key K_B. The holder of a ticket can only send it to its opponent which is able to decrypt it and read its contents. The ticketing concept allows faster reauthentication processes. Each ticket has a limited validity.

- What is a ticket-granting ticket?

  A Ticket-Granting Ticket (TGT) contains the session key of the holder (e.g. S_A), the user's name and an expiration time, encrypted with KDC master key (K_{KDC}). Therefore, a holder cannot read/manipulate its contents. TGT is cached on the local machine in volatile memory space.

  A TGT contains enough information for a KDC to decide about a client's ticket request. There is no need for a KDC to save state information which offers operational advantages.

13.3 Configuration

- What keys and secrets must the participants in Kerberos keep track of?

  • K_C is **long-term** key of client C
  - Derived from user's password
  - Known to client and Key Distribution Center KDC

  • K_{TGS} is **long-term** key of ticket granting service TGS
  - Known to KDC and TGS

  • K_V is **long-term** key of network service V
  - Known to V and TGS; separate key for each service

  • K_{C-TGS} is **short-term** key between C and TGS
  - Created by KDC, known to C and TGS

  • K_{C-V} is **short-term** key between C and V
  - Created by TGS, known to C and V

13.4 Logging Into The Network

- How does the workstation acquire a TGT on behalf of the user?

  A workstation sends an authentication request (KRB_AS_REQ) to the KDC. This message is not encrypted.

  The KDC generates a session key (S_A) and a Ticket-Granting Ticket (TGT; encrypted with K_{KDC}). Both, S_A and TGT are encrypted using Alice's master key (K_A) and sent back to Alice in an authentication reply message (KRB_AS_REP).

  The workstation converts Alice's password into a DES key and encrypts KRB_AS_REP. The received session key and the TGT are enough to continue – the workstation can forget about the user's password…

- How does the user contact a remote node?

  Let's assume that Alice wants to logon to Bob. The workstation of Alice sends the TGT, the target name (e.g. "Bob") to the KDC. (This request is called KRB_TGS_REQ).

  The KRB_TGS_REP contains a ticket to Bob T_B and K_{AB} (the session key shared by Alice and Bob). The ticket T_B does also contain K_{AB} as well as the requester's name "Alice" and an expiration time. T_B and K_{AB} are encrypted with the session key S_A (KDC extracts this key from Alice's TGT) and sent back to Alice.

  Now, Alice's workstation can send the ticket T_B to Bob. He can open T_B with his key, K_B, and discovers the shared key, K_{AB} as well as an expiration timestamp. This step is called KRB_AP_REQ (application request).

  Bob responds with KRB_AP_REP, an increased timestamp, encrypted with K_{AB} to ensure mutual authentication. Alice's workstation is now reassured that it is talking to Bob, since he obviously knows K_{AB}.

- How are the different servers authenticated to each other?

  ???

13.5 Replicated KDCs

- What is KDC replication, and why is it needed? How can the KDC be replicated?

  A KDC is a single point of failure. If it is down (or the network connection is interrupted) the authentication server and with it all depending services are not available. KDC implements a single master replication scheme. Only the master copy of KDC is writable – all replicated copies are read-only.

- What steps must be taken to protect the database?

  The KDC database consists of tuples representing the scheme <principal name, key>. The principal's master key is encrypted with the KDC master key.

  Theoretically, an attacker could rearrange the encrypted key values. This threat is avoided by transferring the KDC database as file including cryptographic hash to preserve its integrity.

13.6 Realms

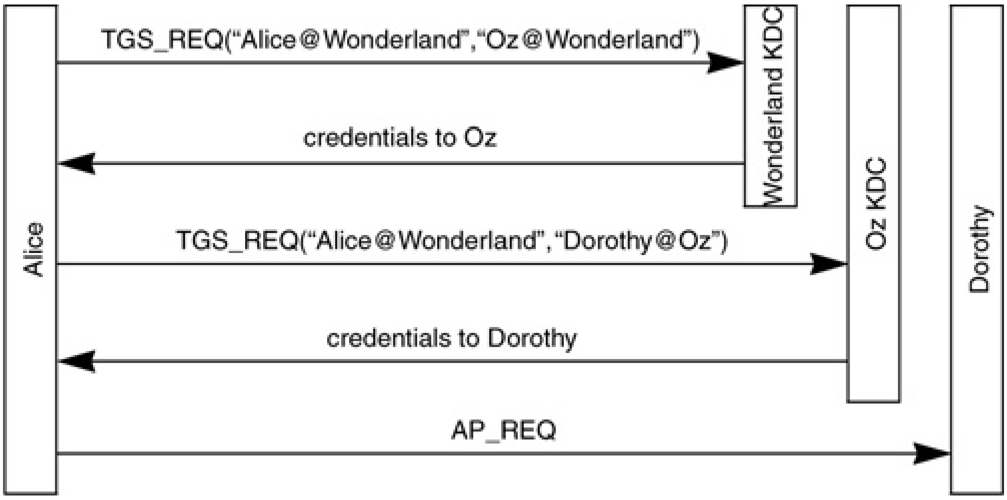
- What is a realm?

  A realm is an administrative domain of Kerberos. Each realm maintains a separate KDC database and is usually administered by different people. Realms can be hierarchical.

13.7 Interrealm Authentication

- How can a user gain access to a resource in a different realm?

  KDC in realm B can be registered as principal in realm A. In the example below, Alice gets a ticket with the credentials to access Oz KDC (within the foreign realm). Oz KDC finally provides Alices the access credentials to the principal in the foreign realm (if granted).


Alice
Wonderland KDC
Oz KDC
Dorothy
TGS_REQ("Alice@Wonderland","Oz@Wonderland")
credentials to Oz
TGS_REQ("Alice@Wonderland","Dorothy@Oz")
credentials to Dorothy
AP_REQ

- Who decides what resources the user can access?

  ??? Kerberos V4 prevents access through a chain of KDCs.

13.8 Key Version Numbers (BRIEFLY)

- Why isn't the key version number exposed to the user, and what implications does this have?

13.9 Encryption for Privacy and Integrity (BRIEFLY)

- What are the key properties of PCBC and what were the perceived benefits?

13.10 Encryption for Integrity Only (SKIP)

13.11 Network Layer Addresses in Tickets

- Why does Kerberos put the Network address in tickets?
- What are the potential disadvantages?

13.12 Message Formats (VERY BRIEFLY)

You should have an idea of what the messages are, and what goes into these messages, but you need not remember the exact formats.