02230: Computer Security
Lab4: Authentication and Access Control

Autumn 2005

The goal of this lab is to provide hands-on experience with authentication and access control in distributed systems. In particular, using the Kerberos authentication protocol to boot strap an access control mechanism.

1 Java Kerberos System

The Java Kerberos System is a distributed system, which means that it could be distributed across several machines or hosts connected to the network. The architecture of the Java Kerberos System is presented on figure 1.

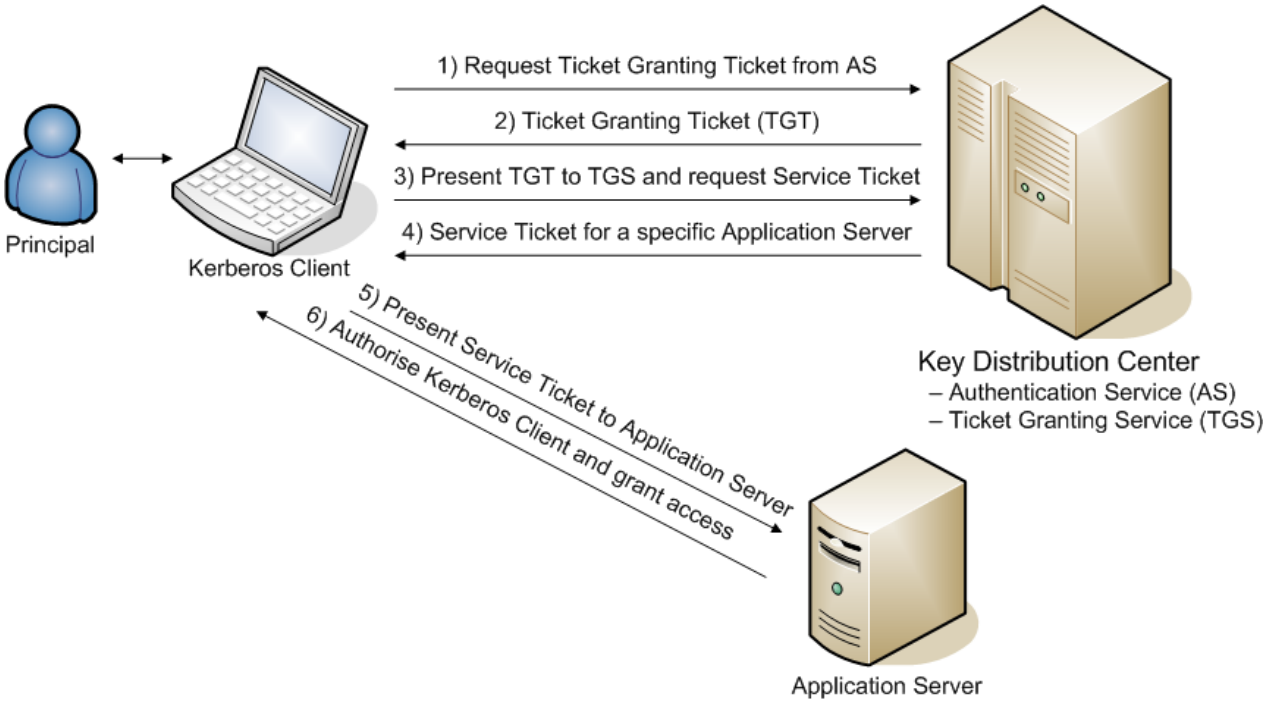

Figure 1: Java Kerberos System Architecture

The Java Kerberos System consists of three different components: a Key Distribution Center (KDC), Kerberos Client(s) and Application Server(s). The KDC represents the heart of the Kerberos system. It maintains a database of Kerberos Clients and Application Servers within its realm, as well as issuing Kerberos Ticket Granting Tickets (TGT) to authenticated clients. It consists of an Authentication Server (AS) and a Ticket Granting Server (TGS), which are both running on the same machine, whereas the Kerberos Client(s) and the Application Server(s) may be running on any host in the network.

2 Task

This lab uses the Java Kerberos System, which provides a userlevel implementation of the Kerberos Services in Java. Initially, you have to download the Java Kerberos System User Guide version 1.1 from
http://www.imm.dtu.dk/courses/02230/labs/Kerberos/krb5-UserGuide-1.1.pdf. This user guide describes a step-by-step installation of the Java Kerberos System and shows how the Java Kerberos System may be used.

The first task is to configure and run the Java Kerberos System, so that the Kerberos Client, the KDC and the Application Server runs on different nodes in the network. This scenario should allow any authenticated client should be able to invoke the application server.

The second task requires you to select an access control model and specify an access control policy that will be enforced by the system (the actual implementation of the access control mechanism that enforces the policy is developed in task 3). The access control policy should distinguish between different principals in the system, e.g., different users, different roles, etc. You could also try to specify a security policy from one of the formal models of access control, such as the Bell-LaPadula confidentiality model or the Biba integrity model.

The third and final task is to implement a mechanism that enforces the access control policy specified above. You are free to implement any access control mechanism, but it must allow you to distinguish between at least two different principals and the evaluation of the mechanism must include an application scenario, where one principal is granted access and another principal is denied access to the protected resource.

3 Evaluation

This lab is a mandatory part of the course, which means that you have to hand in a small report which will be evaluated and count towards your final grade. You are expected to work in groups of 2 participants. Your report should document your solution to the problem and explain how you have chosen to implement and test the access control policy enforced in the system. The report should be limited to a maximum of 5 pages, excluding the source code. **NOTE:** You should not include all source files in the report, instead, only the ones that have been modified.

The reports must be handed in by end of business (i.e., no later than 16:00) on Tuesday, November 1, 2005. Reports should be left in the "mailbox" marked 02230 in the west entrance to Building 322.