Kerberos Quick Overview

Kerberos Quick Overview

This article is designed to give those with zero familiarity a working understanding of the nature and purpose of the Kerberos authentication protocol. For a more in-depth discussion, see

- http://web.mit.edu/kerberos/
- https://en.wikipedia.org/wiki/Kerberos_(protocol)
- http://www.kerberos.org/
- http://web.mit.edu/kerberos/dialogue.html

Most of this article is a repackaging of the key points in the "dialogue" linked to in the last bullet point above.

One of the bigger problems of having an open network environment is authentication, proving someone is who they say they are (and not letting impostors get away with pretending to be someone else). Kerberos is a tool that provides authentication as a service to other services (and users of those services) on a network.

Kerberos was built to meet a bunch of security requirements, most notably that no passwords get sent over the network. It does this by having a centralized Kerberos server (the KDC) which knows the passwords for all users and all services on the system. When a user requests access to a particular service on the network, the KDC issues a "ticket" to the user. This ticket contains information that can only be decrypted using the user's password, as well as information about how long the ticket is good for, and a session key that prevents users on other workstations from stealing and reusing the ticket. When the user receives the ticket, a program on their computer "kinit" decrypts the ticket's information using the user's password (so the password never leaves the computer).

The ticket that the user now has, along with its session key, is actually a "ticket to get tickets" or a "ticket-granting ticket" (TGT). The user can now use this TGT to request tickets for specific services from the KDC. This allows the user to only have to get one ticket per workday, and not have to personally request (and re-enter passwords for) other tickets as other services are used throughout the day.

For more information about the design choices made for this protocol, please refer to the excellent dialogue at http://web.mit.edu/kerberos/dialogue.html