

[< prev](#)   [contents](#)   [next >](#)

---



# Using Kerberos Authentication In Java

## Draft Version

### Fermilab Document #

Beams-doc-1502

### Revision:

November 24, 2004.

### Author:

Andrey Petrov <[apetrov@fnal.gov](mailto:apetrov@fnal.gov)>, [Accelerator Controls Department](#)

© 2004 Universities Research Association. All rights reserved.

---

**This document is obsolete. See the new version, [Kerberos Module fo Java Reference Guide](#).**

## Introduction

A while ago Fermilab started using [Kerberos V5](#) protocol for strong authentication, and by this time a vast number of computers, networking services, and users can work with it in some way or another. However, for most of us Kerberos is an annoying black box, tolerated only for the sake of easy access to telnet and rlogin.

Meanwhile, many Java applications demand some sort of a common and not very complicated tool to authenticate the users. One day, every programmer needs to figure out who is running the program. The most common and least reliable way to do so is to get the user name from the system properties:

```
| String userName = System.getProperty( "user.name" );
```

You do not have to be a *malicious hacker* to break this way of authentication. And, unfortunately, there are too many reasons why people use different account names on different computers.

Using Kerberos authentication in Java looks very attractive. Sun included implementation of Kerberos and GSS-API in Java SDK ver. 1.4, so we do not have to depend on third-party libraries. Existing implementation is pretty comprehensive and quite good (except credential caches). The problem, however, is that it is too complex. An average developer who writes some data acquisition application will likely be confused with non-obvious API and hazy meaning of settings. And the chance that an additional month will be spent for building up Kerberos security is very slim.

## Abstract

This document describes the reusable set of components (known as *Kerberos Module*), that provides a uniform and simple procedure of strong user authentication for Java applications in Accelerator Controls Department. The Module is configured to work with existing Kerberos infrastructure, without changes in user

environments.

## Table of Contents

1. [Basic Concepts](#)
  1. [Kerberos](#)
  2. [GSS-API](#)
  3. [Java Implementation](#)
2. [Kerberos Module](#)
  1. [General Description](#)
  2. [Usage](#)
    1. [Getting Credentials](#)
    2. [How Authentication Works](#)
    3. [Passing Credentials Over](#)
    4. [Exchanging Messages](#)
  3. [Configuration](#)
    1. [Keytab](#)
  4. [Testing & Examples](#)
    1. [Server](#)
    2. [Client](#)
3. [Authentication in Web Browsers](#)
  1. [Task Definition](#)
  2. [Server](#)
    1. [Security Realm](#)
    2. [Auto-Authenticator](#)
    3. [Configuration](#)
  3. [Client](#)

---

[< prev](#)   [contents](#)   [next >](#)  
[security, privacy, legal](#)