

Yale University



Authentication and Single Sign-on Using Java Technologies

Key highlights

Industry/Market
Higher Education

Institution
Yale University

Products

- Sun Fire™ 280R
- Sun Ultra™ 10 workstation
- UltraSPARC® microprocessors
- Solaris™ Operating Environment
- Java™ technologies

Key Business Challenges

- Provide secure single sign-on for system users
- Enable “untrusted” applications to authenticate users without jeopardizing overall system security

Key Business Solutions

- Yale develops the Central Authentication Service (CAS) using the Java programming language

Results

- Approximately 20,000 system users at Yale have single sign-on access to mission-critical applications that provide services like network registration, student grades, financial information, and static content protection
- Yale, and other institutions, have a proxy authentication service that supports portals and other complex, multi-tier systems
- CAS can interoperate with industry-standard operating platforms and authentication mechanisms
- Organizations can download the Java technology-based CAS server and free CAS clients in the Java, ASP, PERL, and PL/SQL programming languages

“Java™ technologies help us to keep applications like Central Authentication Service (CAS) small and simple. It would have been possible to develop CAS using another platform, but it would not have been as easy to be sure CAS was secure and correct.”

Shawn Bayern
Research Programmer
Yale University

Overview

In 1701, Yale University’s first students and faculty assembled in Abraham Pierson’s Connecticut home, drawing upon life experience and a small library to further knowledge and enhance the community. Over 300 years later, the university is still working toward the same goals. However, today’s students connect with people and data resources from around the world, often through secure Web sites.

Deploying secure, Web-based services can prove challenging, but Yale University recently developed a robust Java™ technology-based, single sign-on framework that can authenticate users for Web applications, portals, and complex multi-tier systems such as Web-based e-mail. Available to the public as an open-source product, Yale’s own “Central Authentication Service (CAS)” helps provide 20,000 system users with secure, single sign-on to the university’s mission-critical applications. Yale is also using CAS to support an incipient portal built on the uPortal framework and Sun™ technologies.

Making Single Sign-on Possible without Caching Passwords

Several years ago, Yale decided to offer single sign-on access to applications. However, the school was not satisfied with simple, insecure authentication solutions. For instance, systems that store passwords in a cache circulate a user’s password to every application they access. If one application is compromised, the entire system is.

Yale’s Technology and Planning department carefully evaluated user authentication. They wanted to develop a standalone, middle-tier authentication service that would allow Web applications to authenticate users without having access to users’ passwords. By isolating applications, Yale could more effectively contain any security breach. Additionally, authentication services could be made available to “untrusted” third-party applications developed by students or other organizations. Furthermore, the new system could hide the details of the back-end authentication mechanism, such as Kerberos or LDAP, allowing greater flexibility and maintainability. Given its centralized, independent design, the team aptly named their product the Central Authentication Service (CAS).

The team designed CAS to work as follows: when a user logs into Yale's site, the Web service redirects them to CAS. CAS then authenticates the user, typically by asking for a password and verifying it with a back-end authentication service like Kerberos. CAS then returns the user to the application that originally requested authentication. Thereafter, whenever the user requests access to an application, CAS confirms the user's identity without requiring the user to log in again.

Java and Kerberos Technologies Enable Secure, Flexible Authentication

With CAS's design established, Shawn Bayern, research programmer at Yale University, implemented the source code using the Java programming language. Bayern explains, "We chose the Java language because of the many advantages it offers, coupled with the fact that Java technologies are based on industry standards." Introduced to Java technologies at Yale years ago, Bayern now actively participates in the Java Community Process[™], contributes to the Java in Administration Special Interest Group (JA-SIG), and publishes books on Java technologies. For more information on JA-SIG, visit www.ja-sig.org.

A UNIX systems programmer by nature, Bayern wrote the CAS source code in one week using a text editor. He then tested the code manually on a Sun Ultra[™] 10 workstation, in an environment that used the Kerberos authentication mechanism. "Java technologies help us to keep applications like CAS small and simple. It would have been possible to develop CAS using another platform, but it would not have been as easy to be sure CAS was secure

and correct. We also wanted to ensure that Yale and others could flexibly choose a platform on which to run CAS, and Java technologies helped us achieve this goal." Commenting on the use of Kerberos, Bayern adds, "Yale adopted Kerberos authentication a long time ago, because it is a full-fledged authentication system, not merely a mechanism for validating passwords."

At Yale, CAS was first adopted by a few applications in the Academic Media and Technology department. The service was so successful that other departments, such as Administrative Systems, began using it as well. Within months of its release, CAS had become a university-wide convention and recommended best-practice, supporting applications running on Solaris 2.8, Linux, AIX, and Microsoft Windows. CAS was later migrated on to a single-processor Sun Fire[™] 280R server to support increasing volumes. Yale also began distributing CAS publicly as an open-source product, offering platform-independent clients in the Java, ASP, PERL, and PL/SQL programming languages.

Secure Authentication Streamlines Access and Supports uPortal

"CAS has become a one-stop-shop for authentication at Yale. We no longer have to think about how an application is going to authenticate users. They can just use existing libraries, or implement the protocol manually. Many of our mission-critical applications use CAS to authenticate users. For instance, applications that handle student network registration, student grades, financial information, and software downloads—as well as applications

run by the Yale Library and the Yale School of Medicine—all use CAS. Through a module for the Apache Web Server, CAS can even protect static content without requiring new application programming."

"CAS has made things possible that wouldn't have otherwise been possible," continues Bayern. "For instance, it can be used by a student newspaper application to survey students. Student government organizations can conduct secure elections." CAS also provides for proxy authentication, which allows a portal or middleware service to represent a user to a back-end service, thus improving the security and flexibility of a portal. Yale designed CAS to support proxy authentication, especially since the school is an active participant in JA-SIG's uPortal project.

uPortal is a reusable portal framework developed for higher learning institutions. Already in use by schools such as Villanova, Princeton, and Iowa State, Yale will soon deploy a uPortal-driven Web presence supported by Sun servers and Java technologies such as JavaServer Pages[™] (JSP), the JavaServer Pages Standard Tag Library[™] (JSTL), and servlets.

"One of the reasons we got involved in uPortal was because it is a great avenue for communication with other schools," Bayern explains. "CAS serves that end as well. We receive inquiries from schools around the world about CAS." As technologies continue to connect people and information in new ways, schools like Yale will open new doors for the community—eventually through a single, secure portal.

Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 USA Phone 800 786-7638 or +1 512 434-1577 Web sun.com



Sun Worldwide Sales Offices: Argentina +5411-4317-5600, Australia +61-2-9844-5000, Austria +43-1-60563-0, Belgium +32-2-704-8000, Brazil +55-11-5187-2100, Canada +905-477-6745, Chile +56-2-3724500, Colombia +571-629-2323, Commonwealth of Independent States +7-502-935-8411, Czech Republic +420-2-3300-9311, Denmark +45 4556 5000, Egypt +202-570-9442, Estonia +372-6-308-900, Finland +358-9-525-561, France +33-134-03-00-00, Germany +49-89-46008-0, Greece +30-1-618-8111, Hungary +36-1-489-8900, Iceland +354-563-3010, India, Bangalore +91-80-2298989/2295454, New Delhi +91-11-6106000, Mumbai +91-22-697-8111, Ireland +353-1-8055-666, Israel +972-9-9710500, Italy +39-02-641511, Japan +81-3-5717-5000, Kazakhstan +7-3272-466774, Korea +82-2-2193-5114, Latvia +371-750-3700, Lithuania +370-729-8468, Luxembourg +352-49 11 33 1, Malaysia +603-21161888, Mexico +52-5-258-6100, The Netherlands +00-31-33-45-15-000, New Zealand, Auckland +64-9-976-6800, Wellington +64-4-462-0780, Norway +47 23 36 96 00, People's Republic of China, Beijing +86-10-6803-5588, Chengdu +86-28-619-9333, Guangzhou +86-20-8755-5900, Shanghai +86-21-6466-1228, Hong Kong +852-2202-6688, Poland +48-22-8747800, Portugal +351-21-4134000, Russia +7-502-935-8411, Saudi Arabia +9661 273 4567, Singapore +65 6438-1888, Slovak Republic +421-2-4342-94-85, South Africa +27 11 256-6300, Spain +34-91-596-9900, Sweden +46-8-631-10-00, Switzerland, German 41-1-908-90-00, French 41-22-999-0444, Taiwan +886-2-8732-9933, Thailand +662-344-6888, Turkey +90-212-335-22-00, United Arab Emirates +9714-3366333, United Kingdom +44 0 1252 420000, United States +1-800-555-95UN OR +1-650-960-1300, Venezuela +58-2-905-3800, Or Online at sun.com/store

SUN™ THE NETWORK IS THE COMPUTER © 2002 Sun Microsystems, Inc. All rights reserved. Sun, Sun Microsystems, the Sun Logo, The Network is the Computer, Java, Solaris, Sun Fire, Ultra are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the United States and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc. LFC 3.2 Printed in USA 11/02 FE1907-0/2K