

Cams NASA Customer Case Study

The International Space Station (ISS) is the largest and most complex international scientific project in history. The *station* represents a move of unprecedented scale off the home planet that began in 1998 with the launch of the first two components, the Unity and Zarya modules. Led by the United States, the International Space Station draws upon the scientific and technological resources of 16 nations: Canada, Japan, Russia, 11 nations of the European Space Agency and Brazil. NASA Johnson Space Center provides ground support services for ISS, including computing services to a worldwide community of over 5,000 scientists, engineers, management and other users.

The Requirements

Because the ISS user community is spread around the globe all enterprise applications are deployed and accessed through the ISS web infrastructure. The web applications are loosely coupled resources that were usually created and sometimes managed by separate teams using various technologies, web application servers and operating systems. For example, ColdFusion, Oracle Forms, PERL and Java-based web applications run on redundant web and J2EE application servers including:

- Apache 1.3 on Solaris 8
- Apache 2.0 and Microsoft IIS on Windows 2000 Server
- JBoss 3 and Tomcat 4
- Oracle 9iAS

A partial sample of the ISS web applications and associated technologies includes:

- Application Support Database (ASDB) uses Cold Fusion with IIS 5 on Windows 2000 Server
- Correspondence Tracking System (CTS) uses Oracle Forms with Oracle 9iAS on Solaris 9
- Integrated Risk Management Application (IRMA) uses ASP on IIS on Windows 2000 Server
- Mission Integration Database Application System (MIDAS) uses Oracle Forms with Oracle 9iAS on Solaris
- Vehicle Master Database (VMDB) uses ASP with IIS 5 on Windows 2000 Server

As a result of the somewhat organic evolution, authentication and single sign-on had become an overwhelming issue. Users encountered a multiple sign-on environment and NASA management did not have a way to analyze who was accessing what resources and when. To address these issues, ISS management created the *named-user* initiative with the following goals:

- Require centralized user authentication and access controls to access all web applications
- Provide single sign-on to all ISS web resources
- Log all authentication and access control events centrally
- Enforce a strong password policy
- Require users to periodically change their passwords
- Warn users when passwords are about to expire
- Maintain a history of previous passwords that cannot be reused
- Enable users to reset their own passwords before and after they expire, or when forgotten

The deployment of multiple user directories further complicated the picture. ISS information technology staff managed redundant Sun ONE LDAP servers for its internal user community. However, a top-level Active Directory at the Johnson Space Center and a child Active Directory within ISS were managed by Johnson Space Center information technology staff outside of ISS. Users in these directories represented most of the external users who would also need access to ISS web applications.

The Solution

ISS deployed Cams for its mission critical web applications to provide users with web single sign-on, centralize a web access control policy, enforce security policy and audit user authentications and access requests. A redundant cluster of two Cams policy servers was deployed on dedicated Sun servers. These servers were also used to host Apache Tomcat as dedicated authentication proxy servers. Cams web agents were integrated with ISS web and application servers, which were deployed behind redundant Ingridian load balancers. Three days of Cafésoft integration assistance, training and mentoring were provided on-site to jump start the project including mentoring various web application development teams about web application integration requirements. Key solution components included:

Cams Authentication – Cams login modules were easily customized for a non-standard Active Directory schema and then stacked to support two hierarchical Active Directory nodes and a Sun ONE Directory Server. A user is successfully authenticated if found in one or more user directories. Group information is aggregated and assigned to roles for directories where authentication succeeds. With a single valid authentication users can access web resources on any web or application server protected by the Cams for which they have been authorized.



Oracle Forms Adapter – Oracle Forms is a self-contained client/server application runtime environment that is hosted within the Oracle Application Server. ISS had deployed over 700 Oracle Forms pages, which needed to be included under the Cams single sign-on umbrella. With assistance from Cafésoft customer support, NASA developers were able to create an adapter that transparently passed Cams user session information to the Oracle Forms client for automatic user authentication.

Cams Session Event Handlers – Cams security middleware provides APIs to tap into authentication, access control and session events. Using the standard Cams LDAP session event handler example, ISS was able to add to the Cams user session information regarding the user's password state. Web applications were then able to securely use these values to prompt users in advance of their password expiring.

Cams Identity for LDAP – Cams Identity provides ISS with user account self-registration, password reset, account maintenance and administrator management of user accounts and groups. A customized implementation was deployed for user management with the Sun ONE Directory Server, which included the administrator management of a user's UNIX Posix attributes. Additionally, a customized implementation of password reset was supplied that adhered to NASA's password policy restrictions.

After four months of development and integration testing, Cams went into production at ISS in June of 2004 without a hitch and has provided 100 percent up-time since.

Summary

NASA ISS engaged Cafésoft with the requirement to implement web single sign-on to track *named users* to its mission critical web applications. Working with minimal assistance from Cafésoft professional services, ISS staff was able to quickly integrate, test and deploy Cams as the security middleware for their site. The deployment required use of standard Cams features such as web single sign-on to Apache, IIS, Oracle 9iAS, JBoss and Tomcat servers using role-based access controls. Additionally, customized and stacked Cams login modules and Cams Identity for LDAP were essential components to the overall solution.

With the Cams web application security infrastructure in place, NASA ISS has a secure, powerful and flexible web application security platform to incorporate business policy changes and securely move into the future.



AMERICA

phone: +1.858.268.5100
fax: +1.858.384.3330
e-mail: info@cafesoft.com

EMEA

phone: +07531.36598 00
fax: +07531.36598 11
e-mail: info@sientenbau.com

Cams Platform Support

Cams Policy Server

- Mac OS X and OS X Server
- Red Hat 9 and Red Hat Enterprise 3.4
- Solaris 8/9/10
- SuSE 9/10
- Windows 2003/2000 Server, XP Pro, 2000 Workstation
- Other operating systems with Java JRE 1.4 or 1.5

Cams Web Agents

- Apache 1.3 on Red Hat 9 and Enterprise 3
- Apache 2.0.40 and 2.0.49-55 on Red Hat 9 and Enterprise 3
- Apache 2.0.49-55 on Windows 2003/2000 Server, XP Pro, 2000 Workstation
- J2EE Application Servers via Servlet Filter:
 - BEA WebLogic 8.1
 - IBM WebSphere 5.x
 - JBoss 3.2.x using Jetty or Tomcat
 - Jetty 4.2.x
 - JRun 4.x
 - Oracle 9iAS and 10g
 - Tomcat 4/5/5.5
 - Other J2EE servers that support Servlet API 2.3/2.4
- Microsoft IIS 5 and 6
- Sun ONE 6.x on Solaris 8/9/10
- Tomcat 4/5

User Directories

LDAP

- IBM Directory Server
- Microsoft Active Directory and ADAM
- Novell eDirectory
- OpenLDAP
- SunOne/iPlanet/Netscape Directory Server
- Other LDAP v3 compliant directories

Databases

- IBM DB2
- Microsoft SQL Server
- MySQL
- Oracle 9i/10g
- Sybase
- Other databases with a JDBC 2.0 compatible driver

Cams is secure, flexible and affordable web application security middleware that provides cross-platform web single sign-on, access control and session management services. The Cams policy server deploys on Linux, Solaris and Windows to secure files and web applications on Apache, IIS and Sun ONE web servers, and J2EE application servers such as BEA WebLogic, IBM WebSphere, JBoss, Tomcat and Oracle 9iAS. Cams Identity is a companion product to Cams that provides user provisioning and management features like user self-registration, password reset and administrator management of users and groups.