Secure the Web: OpenSSO

Sang Shin, Technology Architect
Sun Microsystems, Inc.
javapassion.com
Agenda

• Enterprise security needs
• What is OpenSSO?
• OpenSSO features
  > SSO and Access Control
  > Federated Single Sign On
  > Web Services Security
  > Identity Services
• OpenSSO Community
• Summary & Resources
Enterprise Identity-based Security Needs
Enterprise SSO Challenges

• Within an organization - We need Single Sign-On (SSO) within an organization
  > “Every application wants me to log in!”
  > “I have too many passwords – my monitor is covered in Post-its!”
  > “We're implementing Sarbanes-Oxley – we need to control access to applications!”

• Outside of an organization - We need Federated SSO across organizations
  > “We need to access outsourced functions!”
  > “Our partners need to access our applications!”
Enterprise Security Use Cases

• A customer logs into a company’s web site and looks for a product in their online catalog.
• A manager retrieves employee salary histories to determine an individual’s merit raise.
• An administrative assistant adds a new hire to the corporate database, triggering the company’s health insurance provider to add the new hire to its enrollment.
• An engineer sends an internal URL for a specification document to another engineer who works for a partner company.
• A vendor submits an invoice to the company’s accounting department.
• A corporate human resources administrator accesses an outsourced benefits application.
Enterprise Security Challenges

- Single Sign-On
- Access control
- Centralized policy management
- Provisioning and profiling
- Identity auditing
- Standards-based solution
- Easy to deploy and manage
What is OpenSSO?
What is OpenSSO?

- OpenSSO (http://openssso.org/) is a Sun Microsystems-sponsored open source project providing core identity functionality such as:
  - Single sign-on (SSO) and Access Control
  - Federated SSO
  - Web services security
  - Identity Web services
- The project was based on the code base of Sun Java System Access Manager 7.x
- Sun OpenSSO Enterprise 8.0, the currently shipping commercial product, is built from OpenSSO
Sun’s Identity Management Suite

Identity Manager
- Automated Provisioning
- Password Management
- Identity Synchronization
- Identity Auditing

Role Manager
- Role Engineering
- Role Maintenance
- Role Certification
- Identity Compliance

OpenSSO Enterprise
- Single Sign-on/Log-out
- Federation services
- Authorization policies
- Authentication modules

Directory Server
- Directory services
- Virtual directory services
- Security/failover services
- Data distribution services

OpenSSO
- Open Sourced
- Product codebase for Sun OpenSSO Enterprise

OpenDS
- Open Sourced
- Next Generation
- Product codebase for Sun OpenDS SE

3+ Billion Identities Under Management
OpenSSO Architecture
OpenSSO Architecture
OpenSSO Architectural Roles

• OpenSSO Server
  > Provides services like Authentication, Authorization, Federation etc.
  > Is contacted by the Policy Agent for these services
  > Comes in a form of a single deploy'able Web application (opensso.war)

• Policy Agent
  > Sits on the application/web server hosting the application that needs to be protected
  > Intercepts requests to protected resource and redirects them to OpenSSO server
SSO & Access Control
(Within an Enterprise)
How SSO Works (Within a Enterprise)

- “Policy agents” are installed to protect web resources (web sites or web-based applications)
- “Policy agents” interact with OpenSSO “policy server” to handle authentication, single sign-on, and authorization requests
SSO - Initial Login Process

1. Browser sends access request to a protected resource the first time - no SSO-Token is present

2. Agent intercepts the request, and redirects it to OpenSSO server for Authentication

3. OpenSSO server performs authentication and then sends back SSO-Token

4. Agent validates SSO-token and allows access
SSO - Subsequent Access

1. Page request (with SSO-token) to a 2nd protected resource

2. Agent validates the token - no login required
How SSO Works (Within a Enterprise) Again
Authentication & Access Control Schemes
Authentication (by the SSO Server)

- Standard-based, extensible authentication framework (JAAS based)
- Supports multiple pluggable Authentication mechanisms
  - LDAP/AD, RADIUS, Certificate, SafeWord, RSA SecureID, Unix, Windows NT, JDBC, MSISDN, WindowsDesktopSSO (Kerberos), Anonymous, Membership (self-enrollment), Radius, Safeword, HTTP Basic
  - Custom authentication mechanisms using the SPI
- Multi-factor Authentication (Chained Authentication Mechanisms)
- Multi-Level Authentication
- Resource-based Authentication
Authorization (by the SSO Server)

- Policy = Rules + Subjects + Conditions
  - Rules – The resource to be protected (e.g. URL)
  - Subjects – Who is allowed to access (User/Role/Group etc.)
  - Condition – Extra Constraints (IP Address mask, authN level/scheme, time/day etc.)
Federated Single Sign-On
Service Outsourcing Without Federation (Multi-Login problem)
Service Outsourcing With Federation (Single Sign-On)
Important Concepts in Federation

• Identity Provider (IDP)
  > Performs authentication, access control

• Service Provider (SP)
  > Provides services, resources

• Circle of Trust (CoT)
  > A trust relationship exists between its members (IDP's, SP's)
  > Must include at least one IDP

• Metadata
  > SAML specifications describing the entities in a standard way
Use Case #1 of Federation

• University now uses Google gmail as their primary mail system
  > Students don't have to carry two email accounts
  > University saves time and resource

• University still maintains the identity information, performs authentication, authorization
  > It plays the role of IDP
  > Google plays the role of SP

• University might use external student loan processing service for their students/alumni
  > Forms a CoT
Use Cases of Federation

- Business organization let its employees to use Google App, SalesForce.com
- Business organization let its employees to manage their 401K through 3rd-party management company
- Business organization let its employees to manage their healthcare through 3rd-party HMO's
Service provider sends SAML authentication request to identity provider via HTTP redirect.

2 User is redirected to identity provider. User logs in.

3 User is authenticated.
Federated SSO Example (2 of 2)

4. HTML page with a form containing a SAML Assertion and Javascript to post it to the service provider

5. Form with SAML Assertion is posted to the service provider
Federating identities

- **Account Linking**
  - Allows existing accounts at IDP and SP to be linked
  - Persistent opaque identifiers preserve privacy

- **User linked accounts**
  - Login at both IDP and SP to establish link

- **Auto Federation**
  - Matches some common unique attribute (e.g. email address) and links accounts automatically without principal interaction.

- **Bulk Federation**
  - Exchange LDIF files or XML files.
Fedlet
What is Fedlet?

• A lightweight Service Provider (SP) implementation which provide quick enablement of service providers

• Support minimal SSO-related needs in business scenarios without the need for a full fledged Federation product deployment

  > Two guys working in a garage “Two-guy-ringtone” providing ring tones to the Telecom company

• Admin at IDP (Identity Provider) can use the OpenSSO console to create a Fedlet zip file

  > Telecom company as a IDP create a fedlet and give it to the “Two-guy-ringtone” company
Fedlet: SP-Initiated SSO
Fedlet: IDP-Initiated SSO
Demo: Fedlet

www.javapassion.com/handsonlabs/opensso_basics/
(Demo Scenario in the Next Slide!)
Demo Setup

• Installation and configuration of OpenSSO server
  > Single war file - opensso.war
  > Simple configuration - only thing you have to provide is admin and agent passwords
  > Embedded DS (Directory Server) is used - no need to configure DS

• Creation of IDP (Identity Provider) in a new CoT
  > IDP performs authentication and access control policy check
  > IDP maintains the user credentials in the embedded DS

• Creation of Fedlet
  > Functions as a front-end SP (Service Provider)
Demo Scenario

• A user access a resource in a SP (Service Provider)
• The SP redirects the request to the IDP for authentication
• A user logs into IDP
• IDP authentications and redirects to SP
• SP allows access
Web Services
Security
Requirements for Web Service Identity

• Identify the end user and web service participant

• Preserve identity
  > Across multiple 'hops' - end to end
  > Across domain boundaries - beyond company boundary
  > Across vendors' products - standards based

• Using existing standards and technologies

• Container plug-ins for runtime injection and validation of Identity Tokens
  > Glassfish, WebSphere, WebLogic; possibly Tomcat, JBOSS
Web Services Security
Secure Token Services

Validate, issue and translate standards-based tokens and proprietary tokens including Oracle Access Manager & CA Siteminder tokens.
Security Token Service
How does it work?
Identity Services
Identity Services through OpenSSO

> Expose authentication, authorization, and audit capabilities as simple Web services, that is, WSDL or REST.
Identity Services

- Authentication, Authorization, Audit, and Provisioning (AAAP) exposed as Services
- Focused on enabling developers, simplifying security
- Reusable AAAP services as building blocks for Business Integration and Composite Applications
- Supported on developers IDEs of choice
  > NetBeans, Eclipse, Visual Studio
Why Identity Services?

• AAAP are core services in any identity-enabled application whether for security or personalization

• Injecting and consuming identity in applications must get easier
  > Runtime configuration for container as opposed to building into application

• Essential elements for building a Secure Service Oriented Architecture (SOA)
Why Identity Services?

• Developers:
  > Aren’t focused on identity, not a core competency
  > Want to focus on business logic, not the identity implementation
  > Need Identity Services exposed as basic building blocks
  > Prefer building secure applications over security code
Available Identity Services

**Authentication**
Verification of User Credentials

\[ authenticate (\text{username}, \text{password}, \text{uri}) \Rightarrow \text{Token} \]

**Authorization**
Permission for authenticated users to access secured resources.

\[ authorize (\text{Resource}, \text{Action}, \text{Token}) \Rightarrow \text{boolean} \]

**Attributes**
Collection of the profiles of authenticated users

\[ attributes (\text{List attrNames}, \text{Token}) \Rightarrow \text{UserDetails} \]

**Audit Log**
Ability to audit and record operations

\[ log (\text{AppToken}, \text{Token}, \text{Logname}, \text{Message}) \]
OpenSSO Community

In three years...

- 950+ project members at opensso.org
- ~20 external committers

Production deployments

- Audi UK
  250,000 customer profiles
- Telenet
  Foundation for fine-grained authorization
- CPqD
  3000 users, 75 apps, 4 months!
OpenSSO Enterprise Model

- Purchase an OpenSSO Enterprise perpetual license (formerly Access Manager), Sun Identity Management Suite subscription or Java Enterprise System subscription.
- Receive Support and indemnification on OpenSSO commercial builds and Express builds.
- Customers choose whichever builds works best for them!
OpenSSO Enterprise Options

• OpenSSO Express Build
  > A community build that has undergone extensive automated testing and moderate manual testing by Sun Quality Assurance Engineering Team.
  > Delivered every 3 months

• OpenSSO Commercial Build
  > A community build that has undergone extensive manual and automated testing by Sun Quality Assurance Engineering Team.
  > Delivered every 12 – 15 months
OpenSSO: Latest Innovation

- Presto-Change-O Install
  > Embedded Glassfish
  > JavaWebstart Installation
  > Pre-configured
  > One Click

- [http://tinyurl.com/opensssonow](http://tinyurl.com/opensssonow)
Summary & Resources
OpenSSO Enterprise

One solution to solve ALL of your SSO problems
Web access management, Federation, and Secure Web services
Sun Identity: How We're Different?

**Simple**
- Easiest identity Portfolio to deploy, configure and use in the market
- Highest Adoption Rate

**Open**
- Only Supported Open Source Identity Suite in the world
- Implement all Identity Relevant Standards (SAML, XACML, ..)

**Scalable**
- Most Scalable Identity Platform
- Can manage billions of users, roles, partners
- Internal and External
More Information

• OpenSSO Wiki
  http://wiki.opensso.org/

• OpenSSO Project
  http://www.opensso.org

• Sun Identity Management
  http://www.sun.com/identity
Free Training Labs

- Five downloadable, self-paced labs
  - deploy two Apache Tomcat servers
  - SSL-enable them
  - install a software load balancer
  - install OpenSSO into the environment
  - configure for session failover

- Includes virtual image containing OpenSolaris, Glassfish, OpenSSO and OpenDS
  - Fast forward or rewind image using ZFS

- Go to OpenSSO.org and click on Training (left sidebar)
Thank You!