



# Stronger Authentication with Biometric SSO

using  
OpenSSO Enterprise and BiObex™

**Ramesh Nagappan**  
Sun Microsystems,  
Burlington, MA  
[ramesh.nagappan@sun.com](mailto:ramesh.nagappan@sun.com)

<http://www.coresecuritypatterns.com/blogs>



# Setting Expectations

What you can take away !

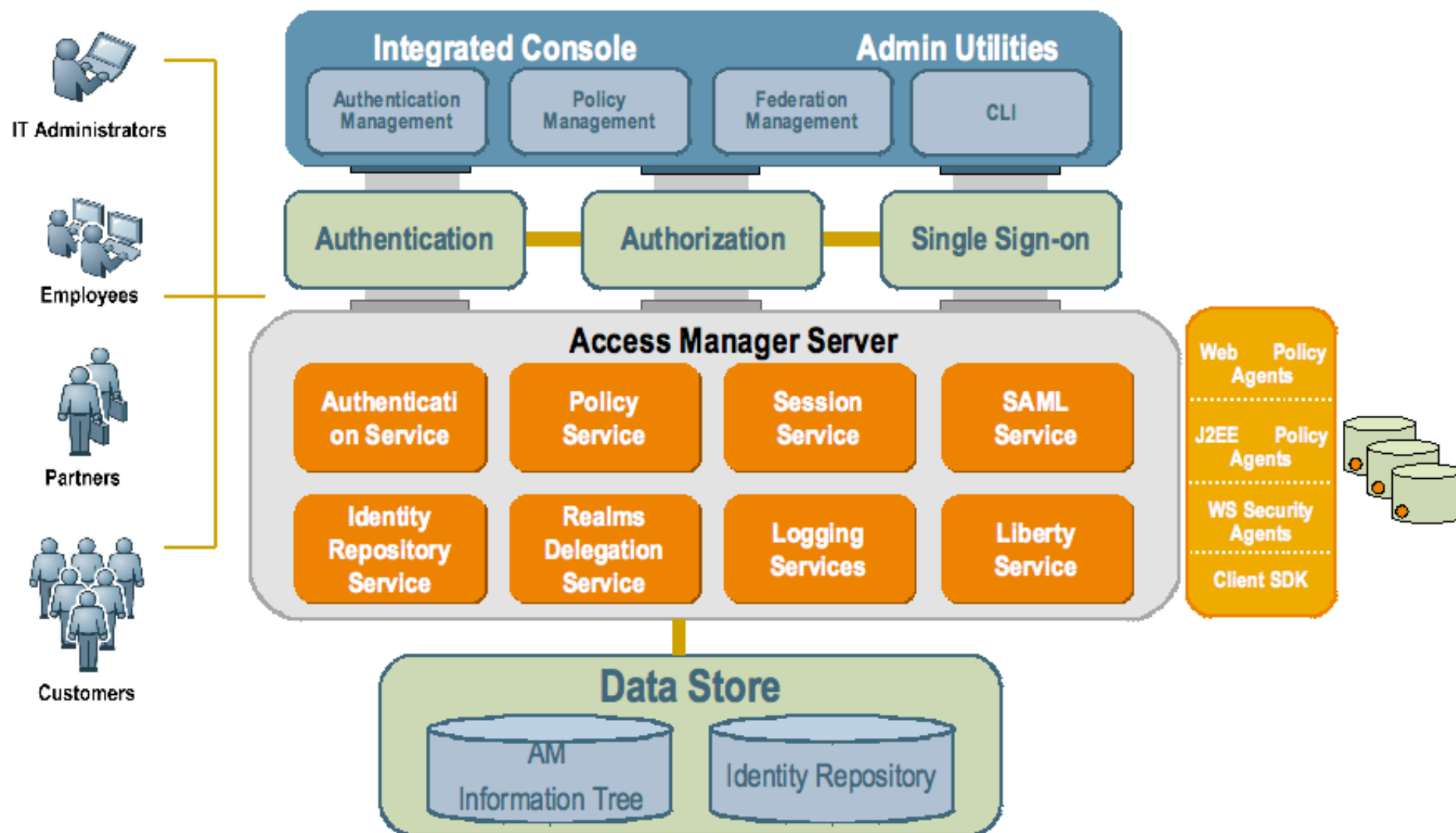
- ✓ **OpenSSO Enterprise and BiObex** - Architectural Overview
- ✓ **Pre-requisites** for enabling Biometric SSO authentication.
- ✓ **Configuration** of BiObex AMLoginModule in OpenSSO environment.
- ✓ **Deployment and Testing** of Biometrics enabled SSO.
- ✓ **Multi-factor/Biometric authentication** based SSO - Moving forward and next steps !

# OpenSSO Enterprise

- **Identity Services Infrastructure** facilitates Single Sign-On (SSO) for Web applications residing within an enterprise or across networks.
- **Open standards based framework** supports centralized authentication, authorization and auditing.
  - > JAAS based authentication services
  - > Agent-based and XACML based policy enforcement
  - > User session management
  - > Identity-enabled XML Web services for AuthN, AuthX, Audit and Provisioning
  - > Identity Federation Protocols support include SAMLv2, ID-\*, WS-Federation, WS-Policy)
  - > XML Web Services Security (WS-Security, WS-Trust, WS-I Basic Security Profile)
  - > Multi-factor authentication via chaining
  - > Centralized configuration, logging and auditing services
  - > Supports multiple Java EE application servers and Web containers
- Deployed as a Web application (single WAR file)

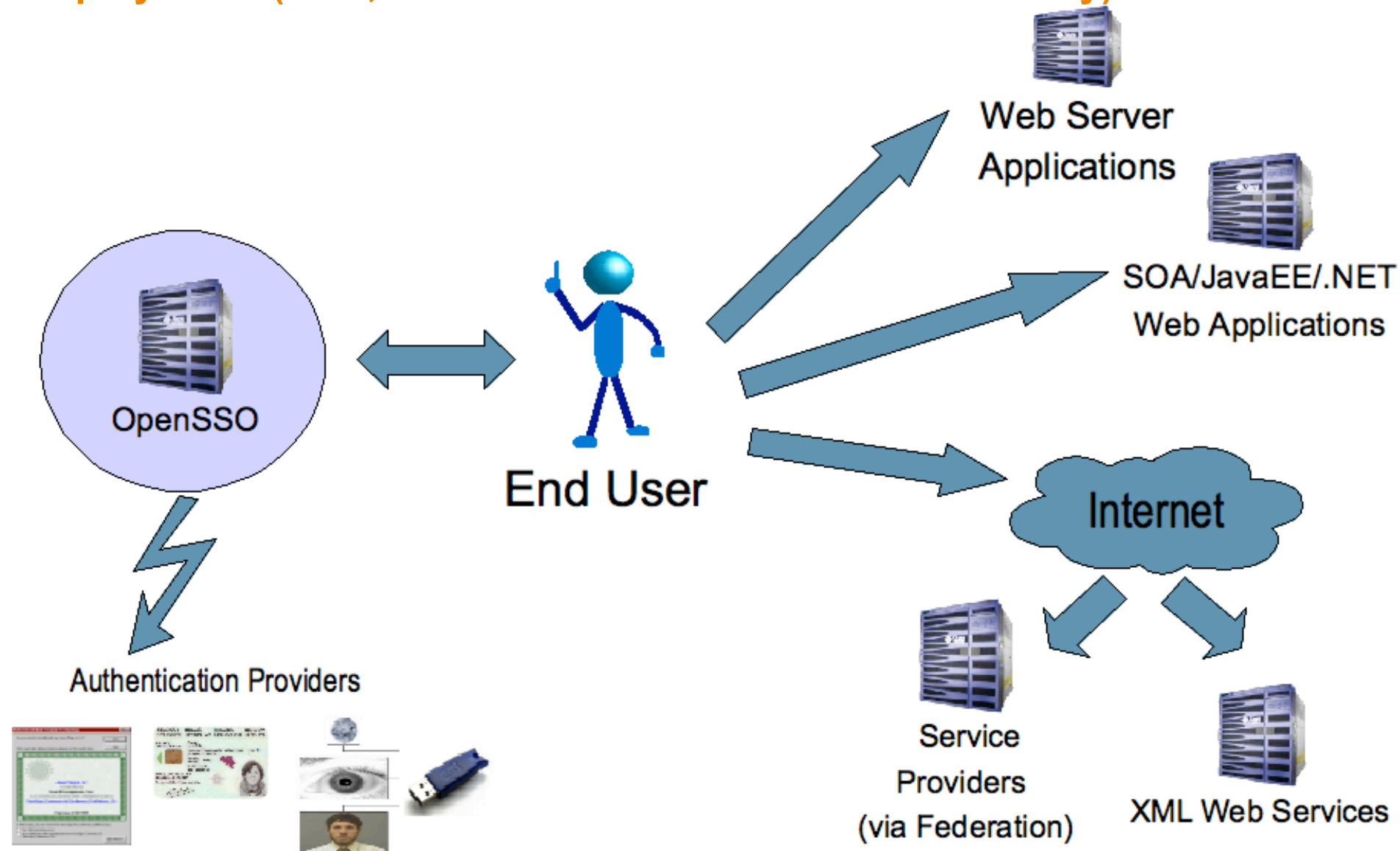
# OpenSSO Enterprise

## Architecture and Services



# OpenSSO Enterprise

Deployment (SSO, Federation and Web Services Security)



# BiObex™

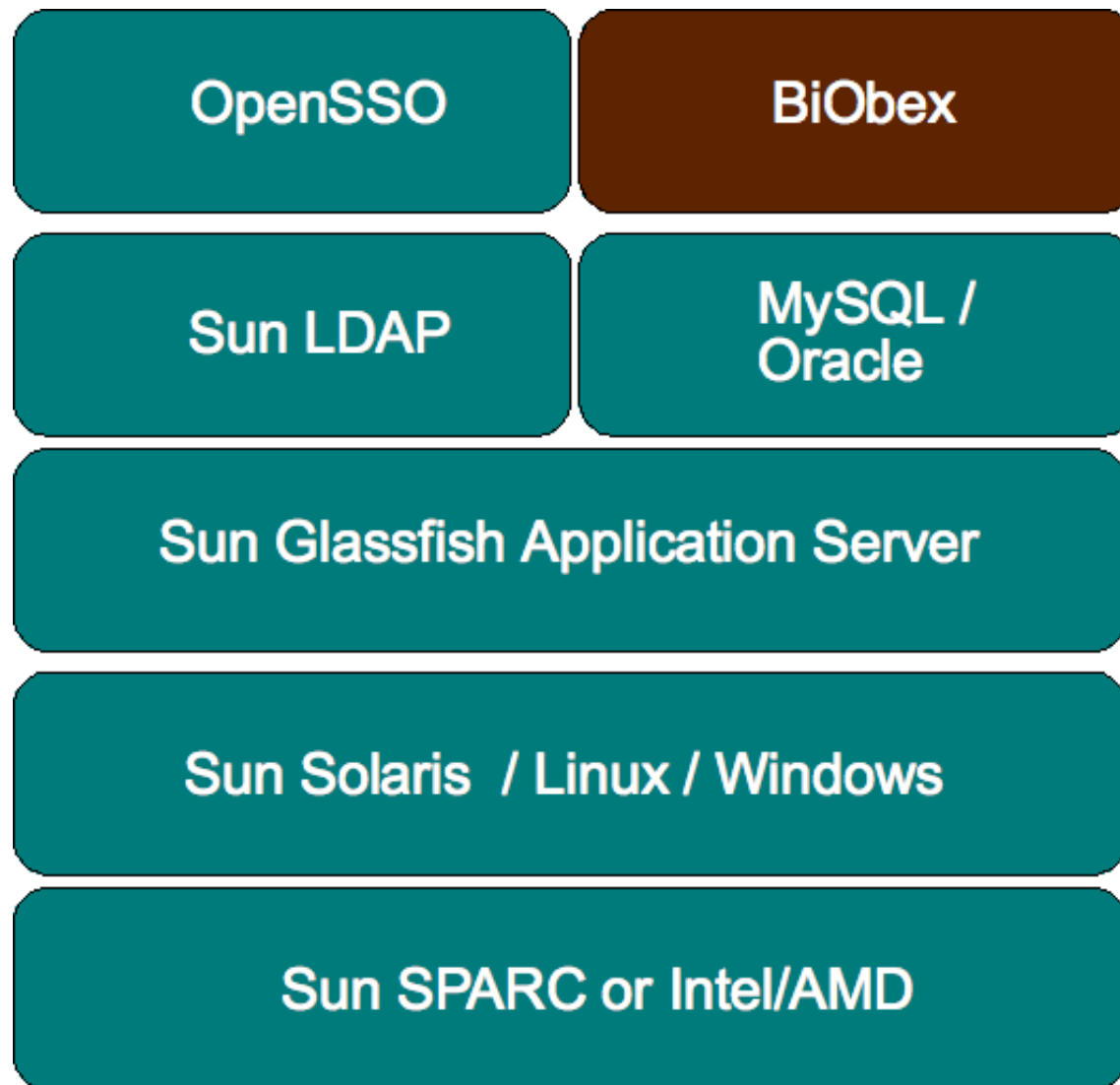
## Interoperable Biometric Middleware

- **Biometric Assurance Infrastructure** facilitates User enrollment and Physical/Logical access control using Biometric credentials.
  - > Fingerprints, Iris, Facial geometry and Hand geometry
  - > Biometric enrollment and device management.
  - > Biometrics based Logical access control enables Web Single Sign-On and Desktop authentication (Windows, Solaris/Linux and Sun Rays).
  - > Biometrics based Physical access control restricts personnel access to doors, buildings, locations and restricted areas.
  - > Standards support include CBEFF, BioAPI, MINEX/INCITS-378 and FIPS-201.
- **Integrates with Sun OpenSSO** for addressing Web SSO and Federation scenarios.
  - > Enables stronger/multi-factor authentication by chaining of Biometric authentication with other credentials such as Smartcard/Tokens, PKI/Digital certificate and Password.
- **Integrates with Sun Identity Manager** for provisioning and de-provisioning of Biometric credentials for Credential issuance and authentication.



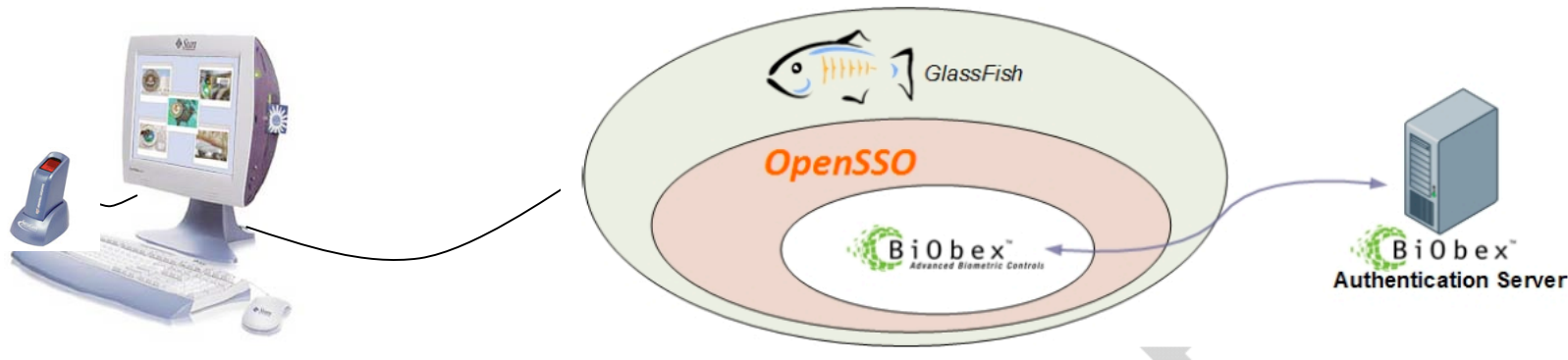
# Biometric SSO - Logical Architecture

## Architecture and its core building blocks



# Tools of the Trade

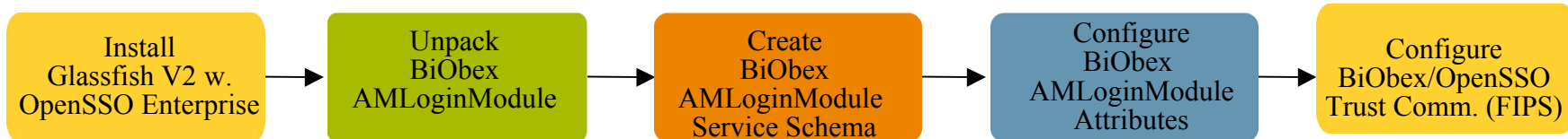
## OpenSSO/BiObex Integration Pre-Requisites



- **OpenSSO Enterprise 8.x**
  - > Deployed on Glassfish Enterprise V2.x
  - > Configured with NSS Keystore (FIPS mode) or JKS (Non-FIPS mode).
- **BiObex 2.8.x Authentication and Enrollment Middleware**
  - > OpenSSO BiObex LoginModule artifacts (Available as part of BiObex 2.8 and above).
  - > BiObex enrollment client for user enrollment.
- **SecuGen Hamster Plus/IV** (preferred) or CrossMatch Verifier-E Fingerprint scanners.
- **Solaris 10 (preferred)** or Solaris Trusted Extensions, Sun Ray, RHEL/SUSE Linux and Microsoft Windows environments.



# Configuration/Deployment Steps



1. Install OpenSSO v8.x Enterprise on Glassfish v2.x Enterprise.
  - Ensure that **NSS Keystore** is available to support **FIPS-mode** communication.
  - Ensure **OpenSSO access to BiObex authentication server** (up and running).
2. Unpack BiObex-LoginModule bundle and deploy the LoginModule artifacts to OpenSSO.
3. Install the BiObex AMLoginModule Service Schema in OpenSSO.
4. Configure the BiObex AMLoginModule Global attributes.
5. Configure the BiObex/OpenSSO truststore to support SSL with FIPS-mode communication.

# BiObex AMLoginModule Installation

1. Unpack the BiObex module archive:

```
jar -xf BiobexAMLoginModuleWebDevices-  
unix-glassfish.zip
```

```
cd biobex-am-loginmodule
```

2. Use GlassFish's ant tool to deploy into OpenSSO:

```
/opt/SUNWappserver/lib/ant/bin/ant
```

# Configure BiObex/OpenSSO Service Schema

## Create a new OpenSSO BiObexLoginModule service

### 3. Configure the BiObexLoginModule service schema via SSOadm console.

- <http://<glassfish>/opensso/ssoadm.jsp>
- Click 'create-svc'.
- Copy 'BiObexService.xml'
- Paste to 'create-svc' entry box.



OpenSSO

[Back to main page.](#)

Sub Command, create-svc  
Create a new service in server.

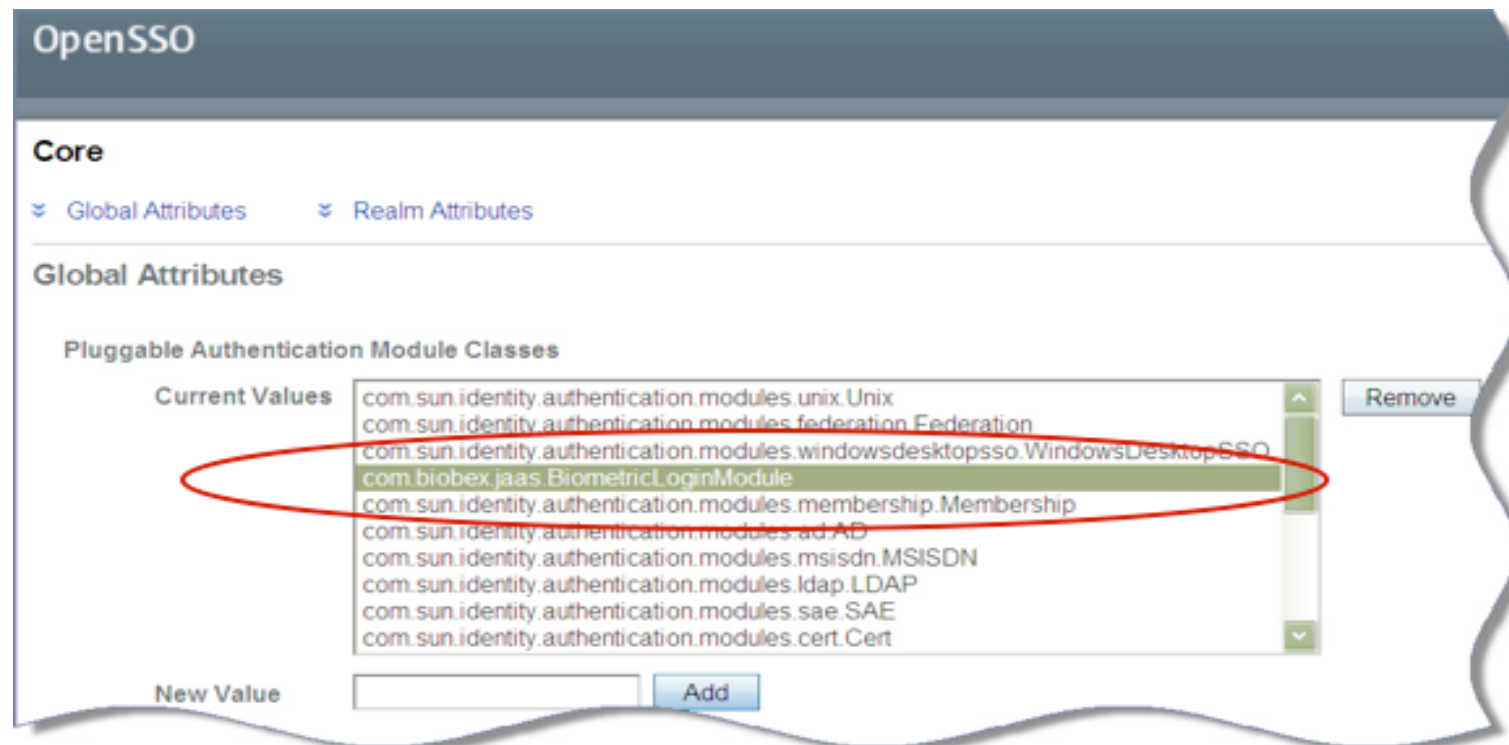
xml\* `<?xml version='1.0' encoding="UTF-8"?>  
<!DOCTYPE ServicesConfiguration  
PUBLIC "-//iPlanet//Service Management Services (SMS) 1.0 DTD//EN"  
"jar://com/sun/identity/sm/sms.dtd">  
  
<!-- Note: the order that configurable values appear in the rendered web page  
is based on the sort order of the i18nKey (like "a101", "a102", etc) -->  
  
<ServicesConfiguration>  
 <Service name="sunAMAuthBiometricLoginModuleService" version="1.0">  
  
 <!-- Biobex Device 2001 = BioKEY BSP -->  
 <Value>2001=29C74AEC69B9466EBD2F56B6055E18F8</Value>  
 </DefaultValues>  
</AttributeSchema>`

XML file(s) that contains schema.

# Configure BiObex AMLoginModule

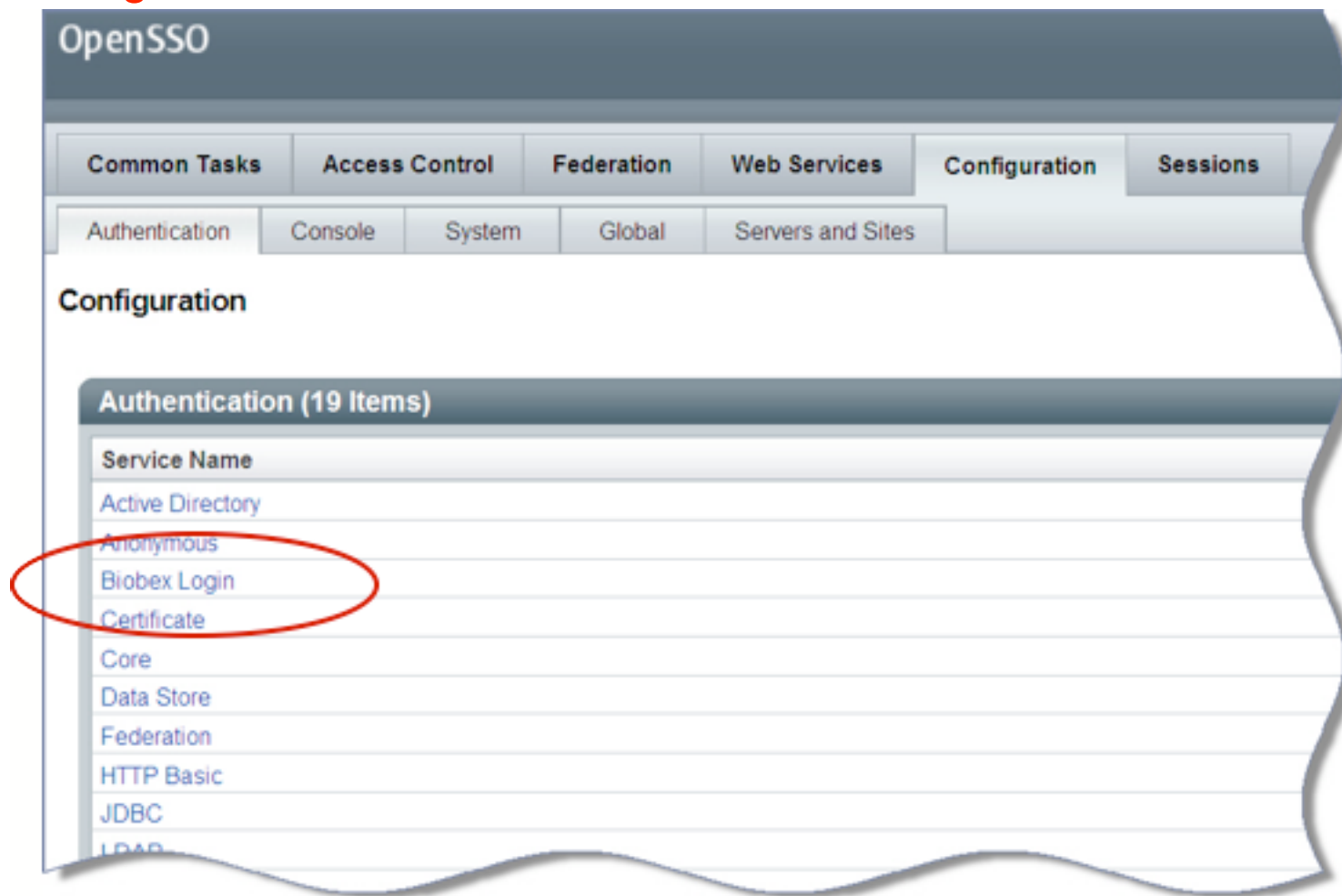
## Configure the BiometricLoginModule in OpenSSO

4. Configure the BiometricLoginModule in OpenSSO pluggable authentication classes.
  - Login to OpenSSO admin console as 'amadmin'
  - Goto 'Configuration' and click on 'Core'.
  - In pluggable authentication, add **com.biobex.jaas.BiometricLoginModule**



# Configure BiObex Global Attributes

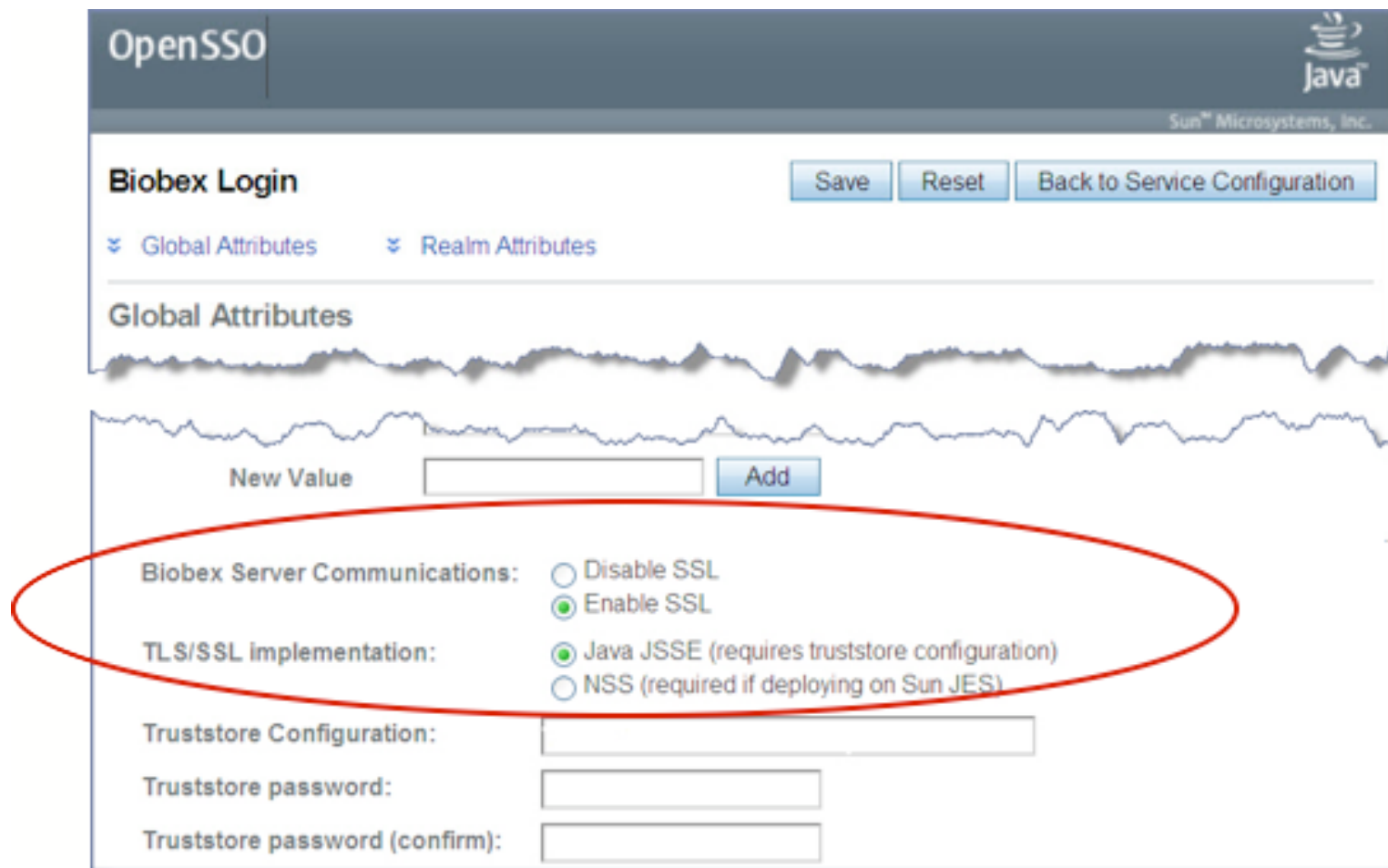
5. Specify the BiObex Global attributes.
  - Goto 'Authentication' and click "Biobex Login" to configure **global attributes**.



# Configure BiObex Global Attributes

## 5. Specify the BiObex Global Attributes

- Enable SSL/TLS communication to BiObex by choosing NSS (FIPS) or Java SE (Non-FIPS).



OpenSSO

Java™

Sun™ Microsystems, Inc.

**Biobex Login** [Save](#) [Reset](#) [Back to Service Configuration](#)

Global Attributes [Realm Attributes](#)

**Global Attributes**

New Value  [Add](#)

Biobex Server Communications: ☐ Disable SSL ☒ Enable SSL

TLS/SSL implementation: ☒ Java JSSE (requires truststore configuration) ☐ NSS (required if deploying on Sun JES)

Truststore Configuration:

Truststore password:

Truststore password (confirm):



# Configure BiObex/Glassfish SSL Truststore

## SSL communication between BiObex and OpenSSO

6. BiObex requires the Glassfish's SSL implementation to enable trusted communication with the BiObex Authentication Server.

- In case of NSS (FIPS-Mode), use the NSS *certutil* tool to import the CA certificate for the BiObex Authentication Server.

1. Note the “-t C” option restricts trust in the Biobex CA to issuing SSL certificates, NOT client certificates.

```
cd /opt/SUNWappserver/domains/domain1/config
certutil -A -d . -t C \
-i ~bioauth/biobex2/certs/bootstrapCA.cer \
-n biobex-authserver
```

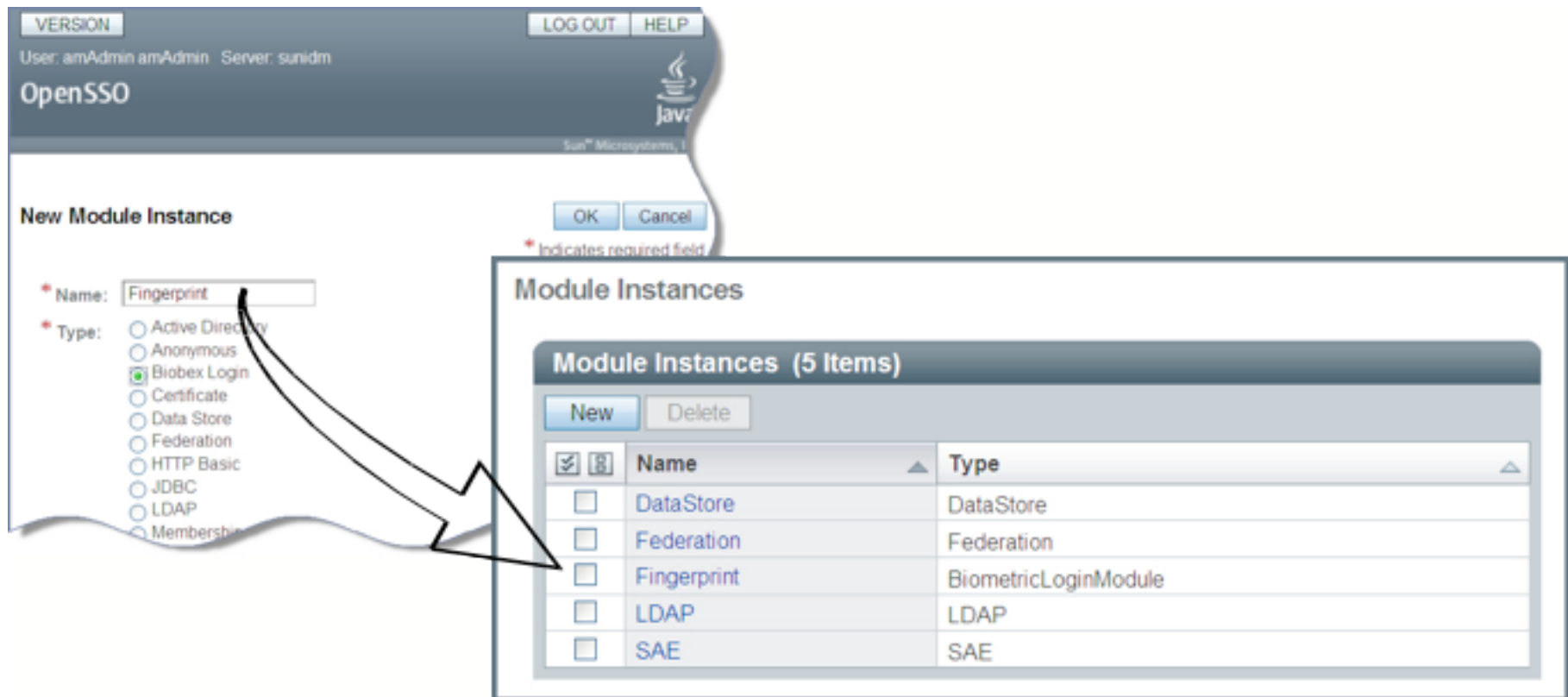
- In case of Java Key Store, use the Java Keytool to import the CA certificate.

```
keytool -import -keystore cacerts.jks -file \
~bioauth/biobex2/certs/bootstrapCA.cer
```

# Configure Biometric Authentication

## Setting up a Fingerprint authentication module instance.

- Configure the BiObex Login Module Instance
  - Goto 'Authentication', select 'Module instances' and click "New".
  - Add a module instance named 'Fingerprint' and choose "BiObex Login".
  - The new module named "Fingerprint" will showup in Module instance list.



The screenshot shows the OpenSSO administration console. The 'New Module Instance' dialog is open, with the 'Name' field set to 'Fingerprint' and the 'Type' set to 'BiObex Login'. An arrow points from the 'Fingerprint' entry in the 'Module Instances' list to the 'Name' field in the dialog. The 'Module Instances' list shows 5 items: DataStore, Federation, Fingerprint, LDAP, and SAE.

**New Module Instance**

VERSION LOG OUT HELP  
User: amAdmin amAdmin Server: sunidm  
OpenSSO  
Sun Microsystems, Inc.  
OK Cancel  
\* Indicates required field

\* Name:

\* Type:

- ☐ Active Directory
- ☐ Anonymous
- ☒ BiObex Login
- ☐ Certificate
- ☐ Data Store
- ☐ Federation
- ☐ HTTP Basic
- ☐ JDBC
- ☐ LDAP
- ☐ Membership

**Module Instances**

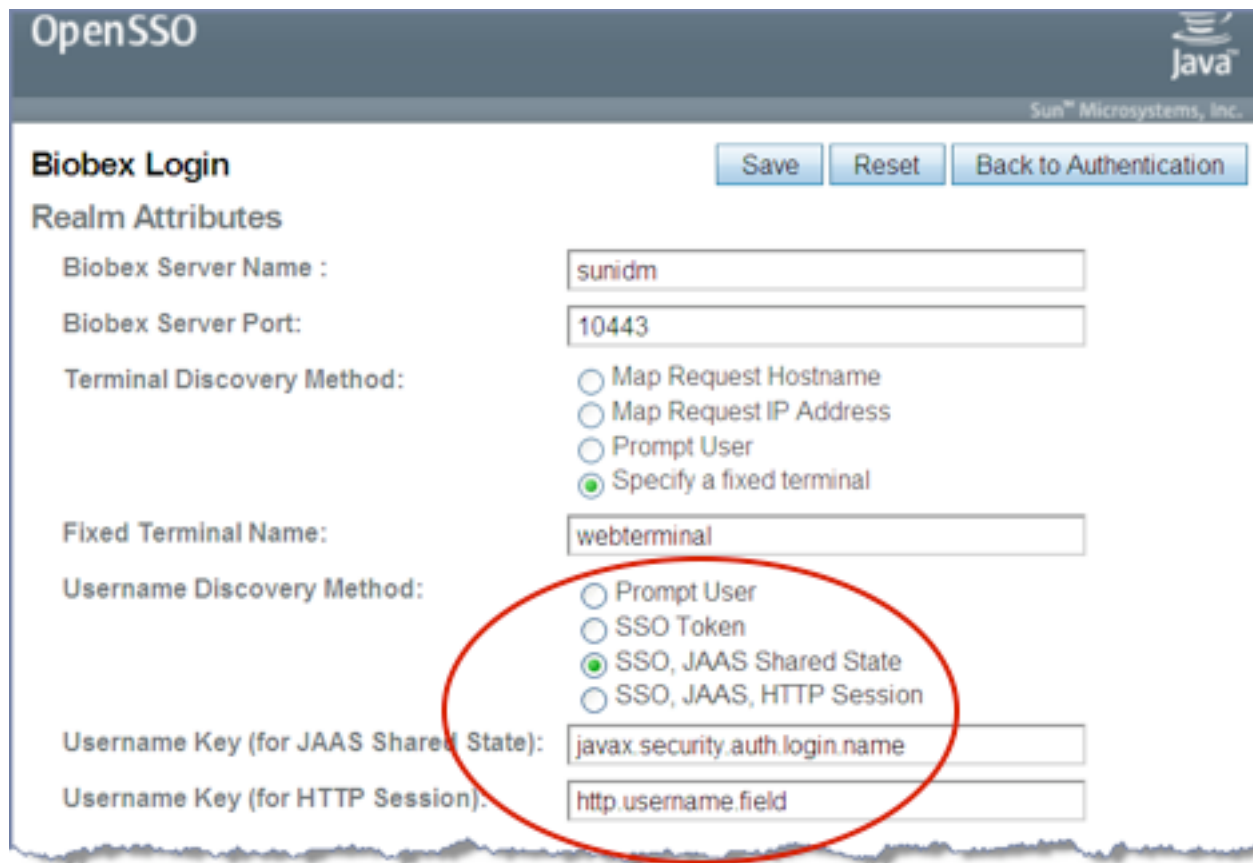
Module Instances (5 Items)

New Delete

<input checked="" type="checkbox"/>	Name	Type
<input type="checkbox"/>	DataStore	DataStore
<input type="checkbox"/>	Federation	Federation
<input type="checkbox"/>	Fingerprint	BiometricLoginModule
<input type="checkbox"/>	LDAP	LDAP
<input type="checkbox"/>	SAE	SAE

# Configure BiObex Login Realm Attributes

- Specify the realm attributes of BiObex AuthN server.
  - Enter the BiObex Authentication server hostname and port (ex. 10443)
  - Set “Terminal Discovery Method” to “Specify a fixed terminal”.
  - Set “User Discovery Method” to “SSO, JAAS Shared State” and then **“Save”**.



The screenshot shows the 'OpenSSO' web interface for configuring 'Biobex Login' realm attributes. The page has a header with the 'OpenSSO' title and a 'Java' logo. Below the header, there are three buttons: 'Save', 'Reset', and 'Back to Authentication'. The main section is titled 'Biobex Login' and contains several configuration fields:

- Biobex Server Name :** A text input field containing 'sunidm'.
- Biobex Server Port:** A text input field containing '10443'.
- Terminal Discovery Method:** A group of radio buttons with the following options:
  - ☐ Map Request Hostname
  - ☐ Map Request IP Address
  - ☐ Prompt User
  - ☒ Specify a fixed terminal
- Fixed Terminal Name:** A text input field containing 'webterminal'.
- Username Discovery Method:** A group of radio buttons with the following options:
  - ☐ Prompt User
  - ☐ SSO Token
  - ☒ SSO, JAAS Shared State
  - ☐ SSO, JAAS, HTTP Session
- Username Key (for JAAS Shared State):** A text input field containing 'javax.security.auth.login.name'.
- Username Key (for HTTP Session):** A text input field containing 'http.username.field'.

A red circle is drawn around the 'Username Discovery Method' radio buttons, highlighting the 'SSO, JAAS Shared State' option.

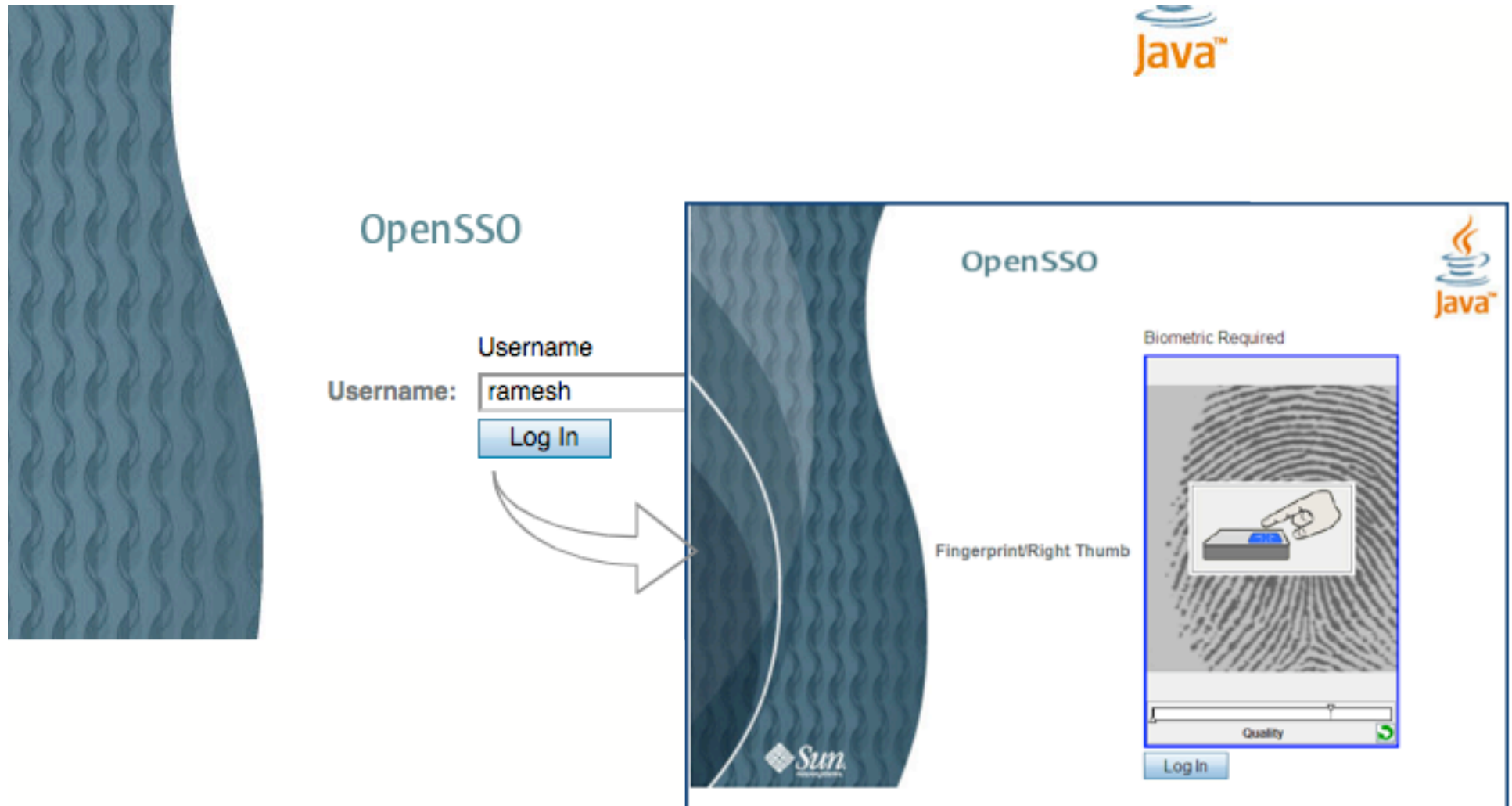
**Important:** Circled option enables 'username' discovery through various methods and facilitates multi-factor authentication and/or session-upgrade scenarios.

# Verifying and Testing Biometric AuthN

## Quick sanity check

1. Install and verify the Fingerprint scanner drivers and test the scanner by capturing sample fingerprints.
  - Make sure the **USB or Ethernet based scanner** is connected and **working properly**.
2. Make sure the **user has already enrolled his/her fingerprints** in BioBex.
  - Verify the user account exist in both BiObex and OpenSSO.
3. Now you are ready to test Biometric authentication...
  - Goto: **<http://<GlassFish>/opensso/UI/Login?module=Fingerprint>**

# Testing the Biometric Login...



OpenSSO login will prompt for random fingerprints as enrolled in BiObex

# Multi-factor AuthN and Session Upgrade

## OpenSSO Authentication Chain and Session upgrade thru' AuthN

- OpenSSO facilitates **stronger/ multi-factor authentication** through authentication chain including multiple authentication providers.
  - > Enables an authentication process where an user must pass credentials to one or more authentication modules before session validation.
  - > Session validation is determined based on the control flag (Required, Requisite, Sufficient, Optional) configured to the authentication module instance chain.
  - > The overall authentication success or failure is determined based on the control flag assigned to each module in the authentication stack.
  - > OpenSSO is tested and verified to provide multi-factor authentication chain that include BiObex Login, Smartcard/PKI and other OpenSSO supported authentication providers.
- **Session Upgrade** allows upgrading a valid session based on a successful “second-factor authentication” performed by the same user.
  - > Allows user authenticate to access second resource under the same or different realm
  - > If authentication is successful - OpenSSO updates the session based on the second-level authentication. If authentication fails, the current session will be maintained.



# Configuring Authentication Chain

Goto: "Authentication" Select "Authentication Chaining" .....

## Authentication Chaining

Authentication Chaining (2 Items)

New
Delete



**New Authentication Chain**

\* Name : PasswordFingerprint

OpenSSO



**PasswordFingerprint - Properties**

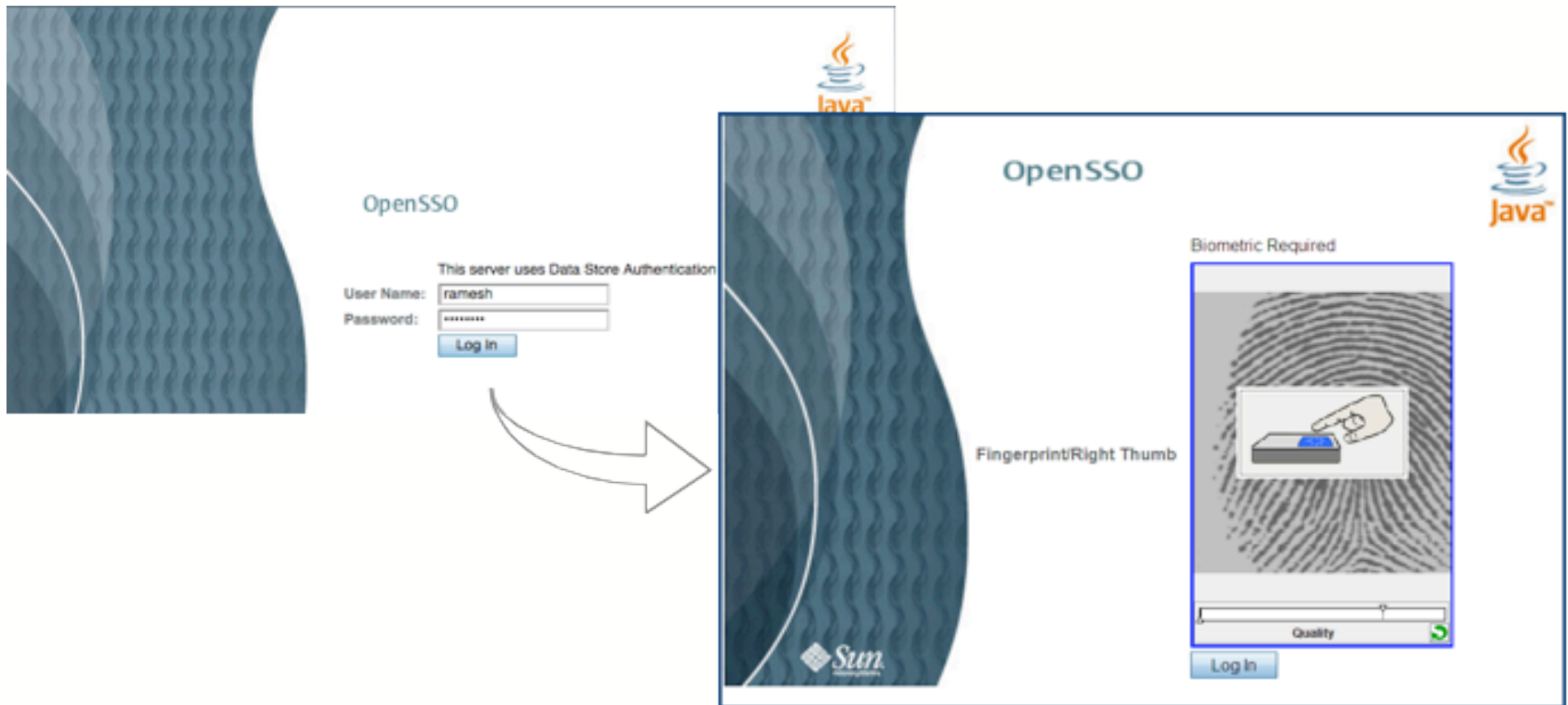
(2 Items)

Add
Remove
Reorder

<input checked="" type="checkbox"/>	<input type="checkbox"/>	Instance	Criteria
<input type="checkbox"/>		DataStore	REQUIRED
<input type="checkbox"/>		Fingerprint	REQUIRED

# Testing Multi-factor/Biometric SSO

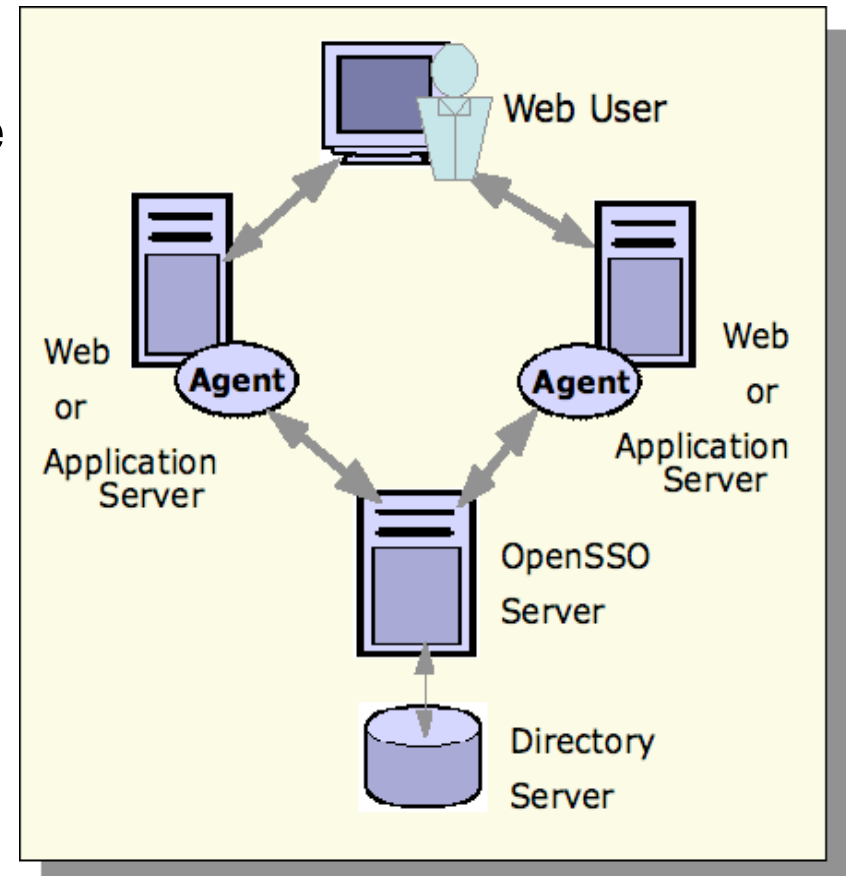
1. Goto: Authentication - Configure “**PasswordFinger**” as **Default Authentication Chain**.
  - Make sure the **IdapService** remains as **Administrator AuthN chain**.
  - Goto: <http://<GlassFish>/opensso>



# Role of OpenSSO Policy Agents

## Authorization and Policy enforcement

- **Policies** are managed by Policy configuration service in OpenSSO.
  - > Policy service authorizes a user based on the policies stored in OpenSSO.
  - > Policies consists of Rules, Subjects, Conditions and Response providers..
- **OpenSSO Policy Agents** enforce policies and policy decisions on protected resources .
  - > Intercepts requests from clients/applications and redirects the requests to OpenSSO for authentication - if no valid session token is present.
  - > Once authenticated, the policy agent communicates with OpenSSO Policy service to grant/deny access to the user based on policy evaluation.



# Attribute Retrieval for Application Use

- **User Profile Attributes**

- > J2EE Agent allows retrieving LDAP Attributes and sets them as HTTP Headers or Cookies.
  - > `com.sun.identity.agents.config.profile.attribute.fetch.mode` (Possible values are HTTP\_HEADER, HTTP\_COOKIE, REQUEST\_ATTRIBUTE, NONE)
  - > Attribute mapping can be done using `com.sun.identity.agents.config.profile.attribute.map`

- **Response Attributes**

- > J2EE Agent allows retrieving Response Attributes and sets them as HTTP Headers or Cookies.
  - > `com.sun.identity.agents.config.response.attribute.fetch.mode` (Possible values are HTTP\_HEADER, HTTP\_COOKIE, REQUEST\_ATTRIBUTE, NONE).
  - > Attribute mapping can be done using `com.sun.identity.agents.config.response.attribute.map`

- **Session Attributes**

- > J2EE Agent allows retrieving Session Attributes and sets them as HTTP Headers or Cookies.
  - > `com.sun.identity.agents.config.session.attribute.fetch.mode` (Possible values are HTTP\_HEADER, HTTP\_COOKIE, REQUEST\_ATTRIBUTE, NONE)
  - > Attribute mapping can be done using `com.sun.identity.agents.config.session.attribute.map`

- **Privileged Attributes**

# OpenSSO / BiObex Troubleshooting

- Enable **Message-level Debugging** in OpenSSO
  - > Goto Administration page, select 'Configuration' tab.
    - > Select your OpenSSO server, In Debug section, select Debug Level to "Message".
    - > Restart your Web container.
- View **BiObex** AMLoginModule logs
  - > Goto ~/opensso/debug/ and view the following files.
    - > "Biobex": Contains tracing when the login module in action
    - > "BiobexSSL": Contains OpenSSO-Biobex server communication related messages, SSL related configuration errors.
    - > "amAuth": Contains message related to LoginModule instance and issues related to configuration of BiObex AMLoginModule.
- View **BiObex Authentication server logs** for issues related to user authentication failure.
- Make sure user has an account in OpenSSO and also enrolled his/her fingerprints in BiObex Enrollment server.

# Environment Requirements

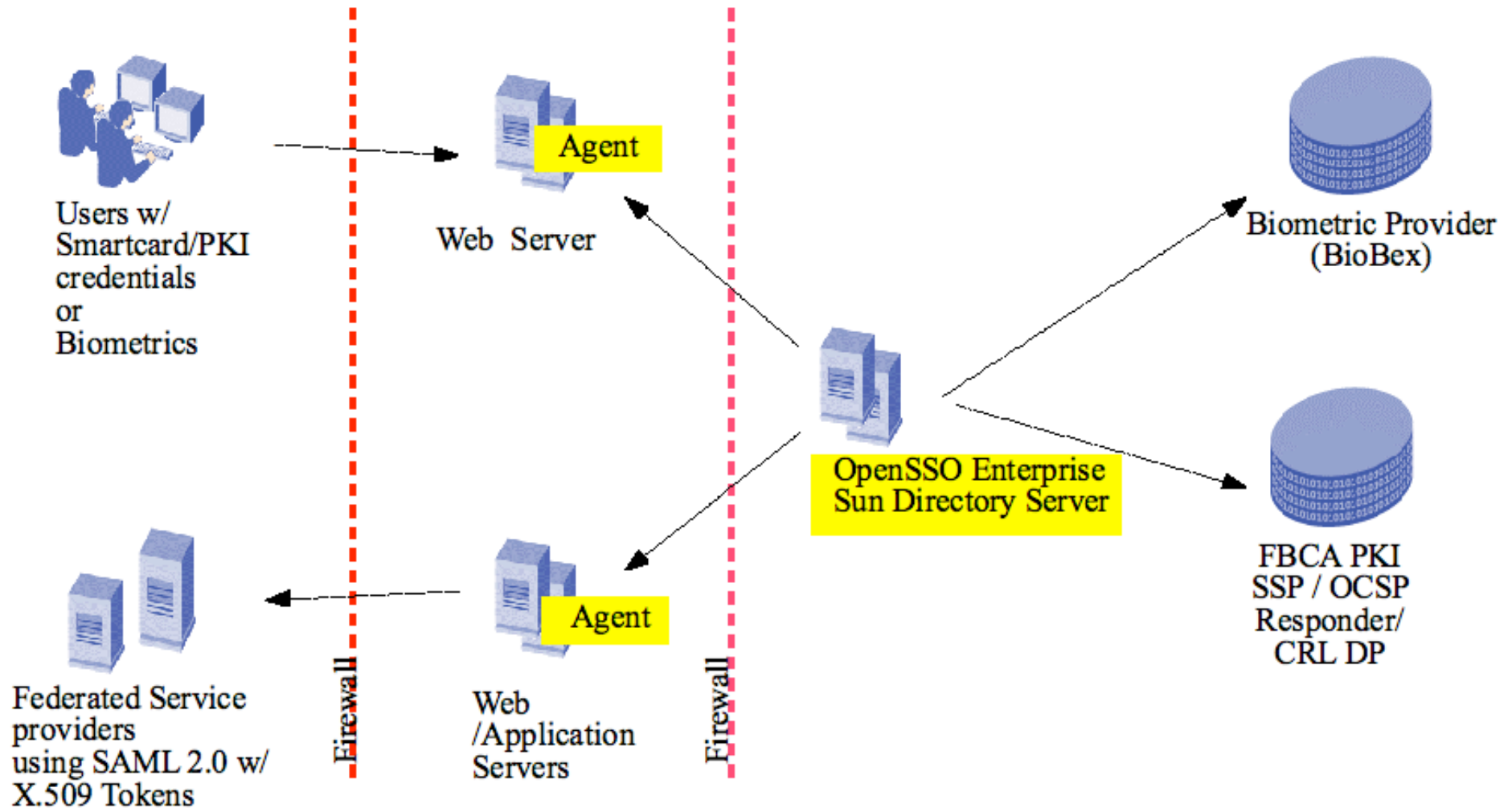
## Supported/verified BiObex environment

Components	Software Products (Provided)	Supported Environment
Application Server		GlassFish V2 Sun Application Server 7.1
Authentication and Authorization Server	OpenSSO/BiObex AMLoginModule	OpenSSO Enterprise Sun Access Manager 7.1
Database Server		Oracle 10g DB2 PostgreSQL 7.3 + MySQL 5.x
User Enrollment Workstation	BiObex Enrollment Client SecuGen Hamster Plus/IV Sensors	JRE 1.5.12 + Windows XP/2003/Vista RHEL/SUSE Linux Solaris / Sun Ray / Solaris TX
Client Workstation (Microsoft Windows / SunRay)	BiObex 2.8 Client BioGINA SecuGen Hamster Plus/IV Sensors	Windows XP/2003 RHEL/SUSE Linux Solaris / Sun Ray / Solaris TX
BiObex Server	BiObex Authentication Server BiObex Enrollment Server	JRE 1.5.12 + Windows XP/2003 RHEL/SUSE Linux Solaris / Sun Ray / Solaris TX



# Real-world Deployment

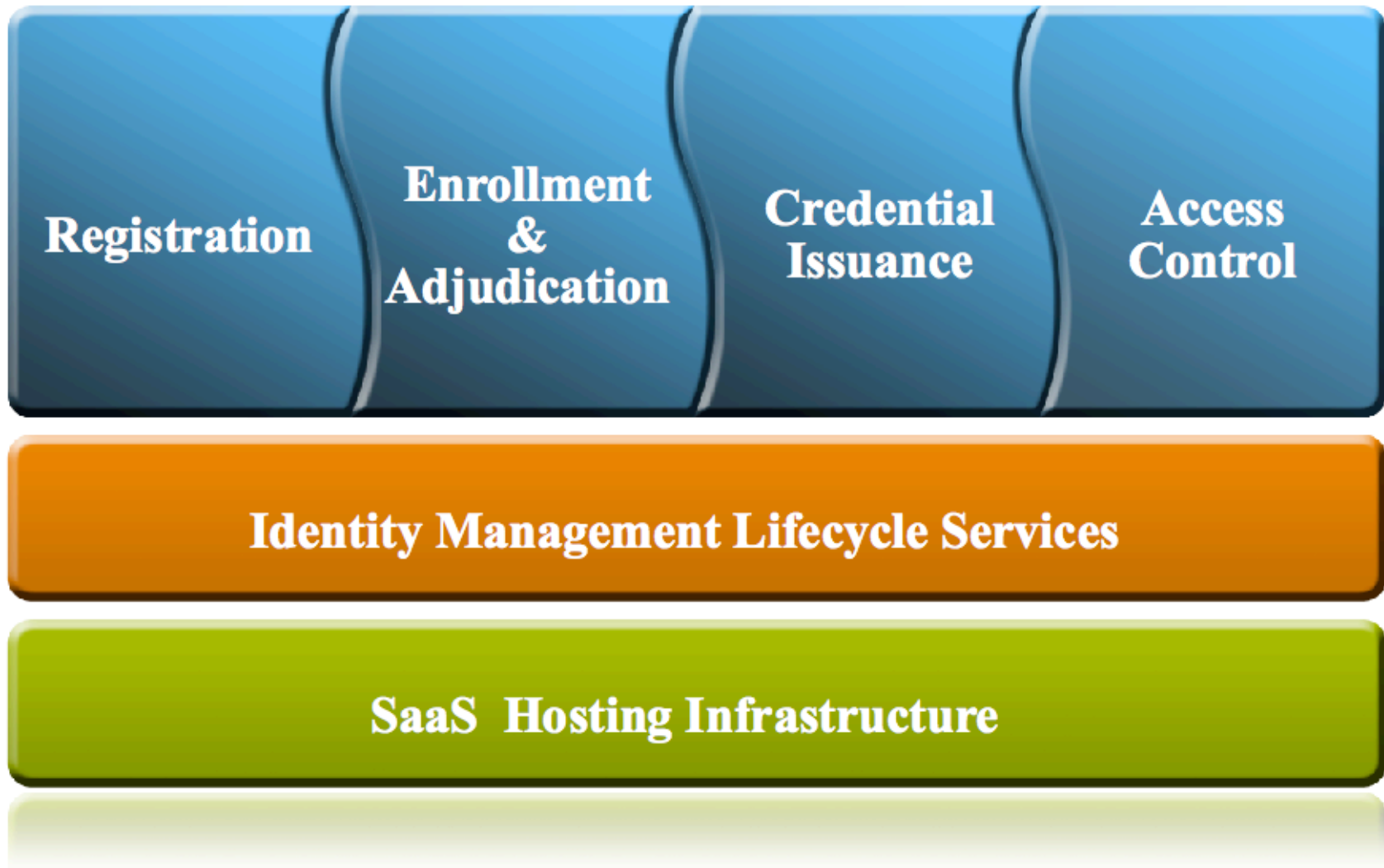
## Biometrics and Smartcard/PKI based Logical Access Control



\* User Desktops/Browsers configured to use Biometric Scanners and Smartcard Readers.

# SaaS/Cloud Deployment

Deploying Biometric Assurance as “SaaS” over Web



# Acquiring BiObex Software

Contact/Support information



Advanced Biometric Controls, LLC  
11501 Sunset Hills Rd., Suite 200  
Reston, Virginia 20190-4731  
Toll-free: 1-877-4 BIOBEX  
877-424-6239  
571-313-0969 Main  
571-313-0962 Fax

Internet: [www.biobex.com](http://www.biobex.com)  
E-mail: [support@biobex.com](mailto:support@biobex.com)



## Q & A

**Ramesh Nagappan**  
Sun Microsystems,  
Burlington, MA  
[ramesh.nagappan@sun.com](mailto:ramesh.nagappan@sun.com)

<http://www.coresecuritypatterns.com/blogs>

