caGrid Web Single Sign On

Design Document

Version No: 0.1

Last Modified: 10/10/07

Author : Kunal Modi

Team : WebSSO

Client : National Cancer Institute - Center for Bioinformatics,

National Institutes of Health,

US Department of Health and Human Services

Document History

Document Location

The most current version of this document is located in CVS under c3pr/docs/designs.

Revision History

| **Version Number** | **Revision Date** | **Author** | **Summary of Changes** |
| --- | --- | --- | --- |
| 0.1 | 6/21/2007 | Kunal Modi | Initial Version |
| 0.5 | 7/01/2007 | Kunal Modi | Added Attribute Level Security |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Review

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Team/Role** | **Version** | **Date Reviewed** | **Reviewer Comments** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Related Documents

More information can be found in the following related CSM documents:

|  |
| --- |
| **Document Name** |
|  |
|  |
|  |
|  |
|  |
|  |
|  |

Table of Contents

[1. Introduction 4](#_Toc180873715)

[1.1 Overview 4](#_Toc180873716)

[1.2 Problem Scenario 4](#_Toc180873717)

[1.3 Scope 4](#_Toc180873718)

[2. Requirement Analysis 5](#_Toc180873719)

[2.1 Summary of Requirements 5](#_Toc180873720)

[2.1.1 Business Requirements 5](#_Toc180873721)

[2.1.2 Technical Requirements 5](#_Toc180873722)

[2.2 Assumptions 6](#_Toc180873723)

[2.3 Dependencies 6](#_Toc180873724)

[2.4 Known Issues or Future Considerations 6](#_Toc180873725)

[3. Detail Design 8](#_Toc180873726)

[3.1 Instance Level Security 8](#_Toc180873727)

[3.1.1 Overview 8](#_Toc180873728)

[3.1.2 Class Diagram 8](#_Toc180873729)

[4. Unit Testing 9](#_Toc180873730)

[4.1 HTTPUnits/JUnit Test Cases 9](#_Toc180873731)

[4.2 Test Case Scenarios 9](#_Toc180873732)

[5. Configuration/Deployment Considerations 10](#_Toc180873733)

[5.1 Property/Configuration Files 10](#_Toc180873734)

[5.2 Deployment Considerations 10](#_Toc180873735)

# Introduction

## Overview

The WebSSO is chartered to provide a comprehensive solution for a Single Sign On framework for participating web application using the caGrid’s GAARDS Framework. WebSSO framework takes care of a framework for user to navigate between different applications without being challenged to provide credentials to login when they switch application. Also this framework also will help the applications establish a grid session for the signed in user. This way the user can access the grid services without being challenged for credentials

## Problem Scenario

In its current form the CCTS provided some form of Single Sing On capabilities. This solution was developed quickly and in-house by the CCTS team which leverages few components of the caGrid Infrastructure to provide the needed Single Sign On Capabilities. However this solution has many security flaws and needed a proper implementation which would adhere to various security policies and guidelines set by the caBIG Security Working Group. Alternatively, with bundling of applications in the form of suites and distributing them collectively amongst the caBIG community, there is a requirement to provide a Single Sign On solution which these teams can leverage easily.

Hence the caGrid Infrastructure team proposes to develop a Single Sign On product which can be easily integrated into the CCTS Suite as well as by other application that wants to leverage the Single Sign On capabilities. This solution will leverage the underlying caGrid Security Infrastructure and use it to establish a user’s identity and maintain his session across applications and grid services.

## Scope

This document defines the detailed design for Web Single Sign On Project. It provides detailed design for the providing a Single Sign On solution using the caGrid’s GAARDS framework including the new Delegation Service. It also shows how target application can leverage the solution. However it doesn’t give any design details about the Delegation Service. A separate document will address that which will be prepared by caGrid Team. Also this design document currently only provides the details for the caGrid WebSSO v0.5 CCTS Release.

# Requirement Analysis

## Summary of Requirements

### Business Requirements

1. **Web based Single Sign On**

There is a requirement to allow a user to provide his credentials only once while logging into an application. However when the user navigates to another related application either by clicking a link or entering the URL in the browser then he should not be challenged to provide his credentials again. He should automatically be logged into that other application.

1. **Grid Single Sign On**

There is a requirement to allow a user to access a Grid service once he has established a single sign on session without providing his credentials again. Also when the user navigates to another web application under the single sign on umbrella, then his grid identity should be effectively delegated to that other application such as he should be able to access a grid service seamlessly as he did from the first application.

1. **Parameter Passing**

There is a requirement for the WebSSO framework to allow application pass user’s specific data elements between each other as part of the URL which links various applications. Currently CCTS applications uses only GET method to pass parameters.

1. **Single Sign Out**

There is a requirement to log the user out of all the applications which he has visited under the single sign on session whenever the user presses logout on any one of the application. Also at the same time his grid session should also be terminated. Due to time restriction the out of box capability provided by the selected product will be sufficient.

### Technical Requirements

1. **Single Identity Provider**

As the entire CCTS Suite is to be deployed at a single Institution the 0.5 release of WebSSO should support a framework where all the applications that are part of the WebSSO Trust Fabric use a Single Identity Provider (IdP) to validate the user’s credentials against it. Also all these applications will use only a single Dorian server to obtain the grid identity for the authenticated user.

1. **Look and Feel**

The Web UI component of the WebSSO project should be configurable to allow different applications to plug in their look and feel templates. CCTS has a commonly developed look and feel the WebSSO application should be able to adapt to this look and feel.

1. **Session Timeouts**

WebSSO Component should provide a configuration parameter to configure the SSO session time out for the user. Also this session timeout should be linked with the user’s grid session.

1. **Browser Support**

WebSSO project should support at least Internet Explorer v6.0 and above as the end user browser. However if possible the WebSSO project should also try to support Mozilla (Firefox) v1.5 and above

## Assumptions

1. **Applications to use Grid Identity**

All the participating applications of the CCTS will use the Grid Identity to identify a user. This is would ensure a common identity across the entire suite and aid in the single sign on capabilities.

1. **Single Identity Providers for the WebSSO v0.5 (CCTS Release)**

For the CCTS Release of WebSSO project, this design assumes that all the applications in the Suite will use a single Identity Provider for user credentials.

1. **Interface Stability**

Since WebSSO CCTS Release v0.5 is an interim release just to support the immediate release of the CCTS v1.0 there is not guarantee for interface stability between it and the final WebSSO caGrid Release v1.0.

1. **User Attributes**

The WebSSO project assumes that the following user attribute obtained from the SAML Assertion are okay to be returned to a target web application along with the user’s grid identity.

* User’s First Name
* User’s Last Name
* User’s Grid Identity

## Dependencies

1. **Delegation Service**

All the participating applications of the CCTS will use the Grid Identity to identify a user. This is would ensure a common identity across the entire suite and aid in the single sign on capabilities.

1. **CCTS Look and Feel**

All the participating applications of the CCTS will use the Grid Identity to identify a user. This is would ensure a common identity across the entire suite and aid in the single sign on capabilities.

## Known Issues or Future Considerations

1. **Sign Out across all application**

Current solution provides an out of box implementation for Single Sign Out capabilities provided by the product used. However as a long term solution a more elaborate Single Sign Out Capability is desired which signs out the user from all the applications as well as destroy his grid session.

1. **Supporting Multiple Identity Provider**

Currently WebSSO solution is designed to support only Single Identity Provider which would be used for CCTS implementation. However in future there would be a need to support multiple identity provider. In this case there should a mechanism devised which would obtain the list of available Identity Providers and provide users an option to select the Identity Provider it wants to use to authenticate against.

# Detail Design

## Out of Box SSO Component

The WebSSO Team along with the caGrid Architecture team evaluated various open source technologies for selecting an out of box standard Web Single Sign On Product. Following are some of their findings

1. The Team looked at the following open source products in details –
   1. Internet 2’s Shibboleth
   2. JA-SIG CAS (Central Authentication Service)
   3. JOSSO (Java Open Single Sign On)
   4. Sun’s OpenSSO (Open Single Sign On)
2. The team decided to drop both Shibboleth and OpenSSO as they are too complex and overkill for our simple requirement of providing a simple SSO solution. Also the deployment of both Shibboleth and OpenSSO was elaborate needing various components to be installed and configured even if they weren’t used.
3. Both JOSSO and CAS are simple framework which met all out business needs. JOSSO uses web service as its underlying protocol for communication between the web agent and the center sign on server. However this caused an issue with the Axis version used by Globus. As a result of which there were compilation errors in the original JOSSO source code which would have needed elaborate recoding of the JOSSO product itself. Due to this technical difficulty JOSSO was dropped
4. Finally the team evaluated JA-SIG’s CAS. It seems to be a perfect fit for our requirement. Following are some of the important features which helped us finalized on it:
   1. Provides a simple spring based pluggable approach to extend and customize CAS to add functionality
   2. Simple non intrusive filter based integration with the target webapps (other ways possible too)
   3. Extendable Client Server Protocol. Provides capability of returning signed SAML assertions back to client from the server, hence providing a higher level of security
   4. Extensive documentation, user base and a bigger development team.

## JA-SIG’s Central Authentication Service (CAS)

The caGrid’s WebSSO Project uses CAS as the core component for providing the Single Sign On Framework. CAS is an authentication system originally created by Yale University to provide a trusted way for an application to authenticate a user. CAS became a JA-SIG project in December 2004.

CAS will be enhanced to use the caGrid’s GAARDS framework in backend to authenticate the user. Also this framework will be enhanced to obtain’s user’s grid credentials on the client side and provide it to the Target Web Application.

More details on CAS can be obtained from their website at: <http://www.ja-sig.org/products/cas/>.

It provides details about its inner architecture, protocols and available features and how to use them.

For caGrid WebSSO Project we are going to the following components from the JA-SIG CAS.

CAS Server v3.1 Stable Release: This is the server component of the SSO which facilitates authentication of user’s credentials and establishment of the Single Sign On Session

JA-SIG’s CAS Client for Java v3.0.

## Overview

The following diagram gives the over view of the WebSSO Project. It showcases all the components that are involved in the WebSSO framework. It also shows how these components interact amongst themselves. These components and their interactions are explained below.

Figure Overview of WebSSO Framework

### Component of WebSSO Framework

1. **Client Browser**

All the participating applications of the CCTS will use the Grid Identity to identify a user. This is would ensure a common identity across the entire suite and aid in the single sign on capabilities.

### Single Sign in work flow

## Interaction between SSO Server and GAARDS

## Publication of Delegation Policy

## Interaction between SSO Agent and Delegation Service

## Interaction between SSO Agent and Target Web Application

## Integration of Sync GTS

Server

Client

## Single Sign Out design

kill delegation property

## Web / Grid Session Sign out

# Unit Testing

## HTTPUnits/ JUnit Test Cases

This solution requires integration with an end user application. Also the testing of these is dependant on many components to be deployed up and running. As a result it is not possible to formulate automated HTTPUnits/ JUnits to test these features.

Hence the team is planning to perform integration tests.

## Test Case Scenarios

The test case scenarios will be developed in conjunction with the QA Team. Based on the initial design the overall test scenarios are as mentioned below. Note that based on details each of these scenarios can have multiple test cases

* 1. Testing Web Single Sign On
     1. Signing into One application and then in the same browser session accessing another application without being challenged for credentials
     2. Signing into One application and then clicking a link on that application which forwards your to another application accessing another application without being challenged for credentials
  2. Testing calling a Grid Service once the Single Sign On Session is established
  3. Testing Parameter Passing when a hot link is clicked to another application
  4. Testing the Sign Out feature
  5. Testing the Web Session / Grid Session Time Out
  6. Testing the common look and feel
  7. Testing for browser compatibility

# Configuration/Deployment Considerations

## Property/Configuration Files

1. WebSSO Sign Server would need the following configuration which needs to be provided at the time of deployment
2. URL for the Authentication Service which fronts the single IdP which will be used by the CCTS Suite
3. URL of the Dorian Server which would provide the Grid Credentials
4. URL of the Delegation Service which would be used to publish and store the delegation policy
5. List of Host Identities of all the applications in the CCTS Suite to which the user’s credential will be delegated
6. Single Sign On Session time out entry
7. WebSSO Client Agent would need the following configuration which needs to be provided at the time of deployment
8. URL of the Delegation Service from where the credentials are needed to be obtained.

## Deployment Considerations

1. The WebSSO Sign On Server needs to be deployed with Host credentials so as to be able to post to the Delegation Service.
2. The WebSSO Client Agent will be deployed along with the target application in its container. It would use the target application’s Host Credentials to access the Delegation Service.
3. The WebSSO Sign On Server and the WebSSO Client Agent + Target Application will have to install and run SyncGTS to establish the Grid trust fabric.