JISC DEVELOPMENT PROGRAMMES

Project Document Cover Sheet

FINAL REPORT

WORKPACKAGE 4 : Implementation of Shibboleth for an e-Science Application Target

Project

Project Acronym	ASMIMA	**Project ID**	
Project Title	Adoption of Shibboleth for Multiple Identity Management Applications		
Start Date	April 2005	**End Date**	March 2006
Lead Institution	Cardiff University		
Project Director	Hugh Beedie		
Project Manager & contact details	Joan Wright wright@cardiff.ac.uk 029 2087 4496		
Partner Institutions	n/a		
Project Web URL	http://www.cardiff.ac.uk/insrv/shibboleth/		
Programme Name (and number)	JISC Core Middleware Programme (Shibboleth Early Adopters)		
Programme Manager	Nicole Harris		

Document

Document Title	Workpackage 4 : Implementation of Shibboleth for an e-Science Application Target		
Reporting Period	February – May 2006		
Author(s) & project role	Jaspreet Pahwa, IT Officer		
Date	9 June 2006	**Filename**	WP4 eScience Service Provider.rtf
URL			
Access	☐ Project and JISC internal	☐ General dissemination	

Document History

Version	Date	Comments
1.0		Completed report sent to JISC

A1.4 WP 4 - Implementation of Shibboleth for an e-Science Application Target

A1.4.1 Introduction

The Biodiversity World (BDW) system is a BBSRC-funded e-Science project which provides an extensible problem-solving environment (PSE) giving biodiversity researchers the ability to link distributed resources together into managed re-usable workflows.

As part of this work package we have developed a proof-of-concept implementation of accessing resources available through the BDW environment in a secured way using Shibboleth. We have developed a software system called the BDWShib system which provides a non-browser based access mechanism for accessing the BDW resources. Non-browser based access enables desktop applications such as the Triana workflow management system to access resources in a Shibboleth-protected BDW PSE. We describe the activities performed under this work package in the span of 4 months as part of the ASMIMA project.

This work package report is organised as follows. Section A1.4.2 provides background information where we introduce the BDW System and identify the importance of providing a secure access to the resources available through the BDW PSE. In this section we also identify the resources for which we have developed a secure access mechanism using Shibboleth in the BDW PSE. Section A1.4.3 defines the aims and objectives for this work package. In section A1.4.4 we describe the implementation of the BDWShib system, its system architecture and challenges addressed as part of the system development to provide a secure access to the resources in the BDW PSE. Section A1.4.5 briefly describes the activities performed as part of system testing and demonstration and section A1.4.6 provides a list of outcomes and results. In section A1.4.7 we identify the valuable lessons that were learned by the ASMIMA project members from this work package and finally in section A1.4.8 we present conclusions and further work.

A1.4.2 Background

Information sources providing access to data in the area of bioinformatics are widely distributed. Various scientists have created Problem Solving Environments (PSEs) in order to achieve an integrated access to these widely distributed data sources to perform complex experiments. For example the web based myGrid project [1] provides middleware for conducting *in silico* experiments in biology. It provides access to a range of tools, services, information repositories and remote legacy bioinformatics applications which are wrapped as Web Services. The GeneGrid [2] provides access and integration of disparate and heterogeneous applications and datasets from across the globe through the creation of a 'Virtual Bioinformatics Laboratory'. It provides access to resources and tools to biologists interested in the development of antibodies and drugs.

The Biodiversity World (BDW) system provides a framework for biodiversity problem solving by providing access to widely dispersed and disparate data sources and analytical tools. The BDW system is a biology-led project and is being actively used for biodiversity informatics research in three main exemplar study areas: (a) biodiversity richness analysis and conservation evaluation, (b) bioclimatic modelling

and global climate change, and (c) phylogenetic analysis and biogeography. In all three of these research areas existing data sources and analytical tools are widely dispersed, available on different platforms and present data in different formats. The BDW system provides scientists with tools which allow ready access to resources originally designed for use in isolation and the ability to compose these resources into complex workflows. It enables chaining of data processing operations and provides flexibility both in the choice of the kinds of resources to be used and in the sequence of operations to be performed for conducting biodiversity experiments. The system is extensible so that new resources and tools can be added to it.

The BDW system is a desktop-based system for conducting *in silico* biodiversity experiments. The system uses the Triana [3] workflow management system to provide workflow capabilities. In the BDW system we provide access to widely distributed and disparate data sources and analytical tools as part of a PSE. Access to heterogeneous resources is provided via resource wrappers. By using resource wrappers we are able to use a standardised mechanism to perform operations on heterogeneous resources and analytical tools from the Triana workflow application. Resource wrappers are deployed as web services in a web services-enabled Grid environment within the BDW system. The architecture of the BDW System, a description of its three exemplar study areas, and the manner it uses computational and BDW resources in the Grid environment is described in our recent publications [4], [5].

Security is an important area to address to provide a secure access to analytical tools and data sources available in the BDW PSE which are vital for performing complex biodiversity experiments. Resources such as data sources are owned and managed by particular organisations or communities of researchers, and some resources contain a large amount of intellectual property to which their owners wish to regulate access. Results derived from biodiversity experiments can also contain sensitive information. For example, a biodiversity experiment may use species data pertaining to a species that is endangered or under specialist conservation program and project its distribution across the world onto a map. Therefore providing restricted, secure access to biodiversity data for its use by authorised users is of significant importance.

The BDW System provides access to several remote and local heterogeneous resources in its PSE via the resource wrappers. As part of this work package we aimed to develop a proof-of-concept implementation of providing a secure access to two resources initially which are available via the BDW PSE using the Shibboleth framework. The two resources are: the database of the International Legume Database & Information Service (ILDIS) project [6], and the Species 2000 project portal. The ILDIS is a collaborative organisation involving legume experts and institutions from all over the world. The ILDIS project aims to document and catalogue the world's legume species diversity in a readily accessible form. The ILDIS database is managed locally at Cardiff by Dr. Richard White.

The Species 2000 [7] is a "federation" of database organisations which brings together an array of global species databases covering major groups of organisms of world’s species (plants, animals, fungi and microbes). It aims to produce a dynamic checklist of all known organisms by accessing an array of global species databases. A global species database contains all known species for a particular group of organisms (e.g. legumes, fish, bacteria, etc). Further it also aims to act as an index to

a virtual library of biodiversity information on the World Wide Web. In the BDW PSE access to data from Species 2000 portal is provided via the Spice resource wrapper which is deployed as a web service.

A1.4.3 Aims and Objectives

The main aims and objectives of this work package are:

- Familiarisation with the Shibboleth Security Framework.
- Installation of the Shibboleth Service Provider (SP) framework for providing a secure access to BDW resources.
- Configuring trust between the Cardiff Identity Provider (IdP) and SP (at the BDW side).
- Using Shibboleth to provide a secure access mechanism when retrieving species data from the ILDIS database via a web service interface from the Triana application in the BDW PSE.
- Using Shibboleth to provide secure access to data from the Species 2000 project portal via the Spice resource wrapper from the Triana application in the BDW PSE.
- Deploying resource wrappers as web services which allow access to remote resources in the BDW PSE and protecting them using Shibboleth.
- Implementing a software system (The BDWShib system) at the client side for enabling user authentication and providing programmatic access to Shibboleth protected BDW resources.
- Creating new workflow units in the Triana workflow system which enables usage of the BDWShib system inside the Triana system.
- Composition and execution of BDW workflows in the Triana system which interact with Shibboleth protected BDW resources.

A1.4.4 Implementation

The implementation of this work package started with familiarisation of Shibboleth components at the SP side and their installation. We installed Shibboleth SP on a BDW server locally running the Debian-based Ubuntu Linux distribution. The installation of Shibboleth SP was straight forward (however not easy). This was mainly because installation of the Shibboleth SP also required installation of other software utilities and libraries on which it is dependent. This dependency list is identified in the MATU website (http://www.matu.ac.uk/) as part of Shibboleth SP installation steps. We joined the Shibboleth user community by subscribing to the Internet2 Shibboleth-users mailing list.

The next step in the process of setting up the Shibboleth SP was to link the SP with an IdP for user authentication. We established links between our SP which provided access to BDW resources with the CU's IdP setup by the INSRV with the help of Rhys Smith at INSRV. The links were established (as part of the bilateral setup of the Shibboleth security framework) by configuring trust between the SP and IdP and by using a digital certificate signed by the INSRV for secure exchange of authentication statements between the IdP and the SP. Once this step was completed we were able to test our Shibboleth SP installation which provided a secure access to BDW resources to the users of CU. This step was achieved with the support of members of the ASMIMA project team at INSRV.

At this stage we were able to begin our implementation work of providing access to BDW resources using Shibboleth from the Triana system. The BDW system provides

a web service based approach for accessing resources in the BDW PSE. In the BDW PSE the resource wrappers corresponding to resources are deployed as web services and each web service defines a list of operations that can be invoked on the corresponding resources. The Triana workflow units invoke these web service operations for performing operations on remote resources and retrieving intermediate/final result data when a workflow is executed whilst conducting a biodiversity experiment. Therefore in the BDW PSE, we required access to Shibboleth protected web services from the Triana application. Using Shibboleth for providing a secure access to resources in the BDW PSE required addressing the following technical challenges.

A1.4.4.1 Access to Shibboleth protected resources from desktop-based applications

The Shibboleth framework allows inter-institutional sharing of web accessible resources with users who are authorised to access them. The framework is well suited for application environments where resources or applications are accessible via the web and require direct interaction with a user using a web browser. However in an application environment such as the BDW, users cannot interact with web accessible resources directly. The resources and analytical tools in the BDW PSE are available as web services which do not possess GUI capabilities and can only be invoked from a web service client application or through the desktop-based Triana workflow application in our case. Hence, using a web browser to perform user authentication for accessing resources protected using the Shibboleth framework is not conducive for the BDW PSE.

Furthermore, the web browser-based user authentication and subsequent resource access is not appropriate to the BDW PSE because the functionality of Shibboleth authentication is required from within the Triana application. Triana invokes a resource wrapper web service when it executes a unit representing the resource wrapper whilst a workflow is being enacted. In the BDW PSE, we require the ability to perform user authentication using a workflow unit which is part of a larger workflow. This design will allow us to link workflow units corresponding to tasks such as user authentication; resource access and invocation of operations on resource; results display; etc. when creating a workflow and execute these units in a sequence. Therefore we required a mechanism whereby a user is able to perform user authentication as part of executing a workflow inside the Triana application as opposed to using an external application such as a web browser.

A1.4.4.2 Sharing of Shibboleth sessions with other applications or toolkits

When a user is authenticated, the Shibboleth SP establishes and maintains a session with the user's web browser on behalf of an application or resource which it is protecting. This session consists of cookies which are exchanged between the web browser and the web server. The cookies are associated with a security context at the SP which holds user's authentication information and a set of attributes describing user's identity.

In the BDW PSE, we not only required access to a Shibboleth protected resource, but also the ability to perform operations on the resource from a client application on behalf of a user. By using a web browser, a user can successfully access the resource wrapper web service page after going through the Shibboleth authentication process. However at this stage, the user is also stuck and cannot proceed any further because the user can only view the page. Because of the distributed nature of the

BDW system, the user requires a separate web service client application in order to access the web page and then invoke the web service operation available on the web page. In this scenario, even if the web service client tries to access the protected page after the user has authenticated using a web browser; it is denied access because the web service client is not a part of the same session with which the user initially authenticated. This session is maintained inside the web browser in the form of cookies. The web browser and the web service client are two different applications and there is no legitimate means of sharing session information between these two applications when accessing a Shibboleth protected resource. Therefore in our system we required a mechanism of sharing Shibboleth session information so that resource access operation could be carried out smoothly after user authentication.

A1.4.4.3 The BDWShib System

In order to address the challenges identified in sections A1.4.4.1 and A1.4.4.2 we designed and implemented the BDWShib system which enables user authentication and provides a mechanism of subsequent resource access for a resource protected by the Shibboleth framework by sharing session information. The system in its present version is implemented to be used from within the Triana application. At the client side, the BDWShib system provides a non-browser based solution for performing user authentication. This scenario is applicable in environments where direct interaction of a user with the Shibboleth protected resource is not possible or is not required. For example, in the case of web services based resources where the user interacts with resources via software agents or a web service client application. This scenario is also applicable in environments where a set of tasks are required to be batch processed at certain times during the day or at regular intervals and involve interactions with Shibboleth protected resources. The BDWShib system is implemented using the Java programming language. The system uses Jakarta Commons *HttpClient* [8] library for requesting web pages from the Shibboleth protected web services in the BDW PSE over the HTTPS protocol. Fig. 1 illustrates the architecture of the BDWShib system and the sequence of operations which are performed in order to authenticate a user and allow access to Shibboleth protected BDW resources. The architecture of the system can be summarised as a series of steps which the system performs.

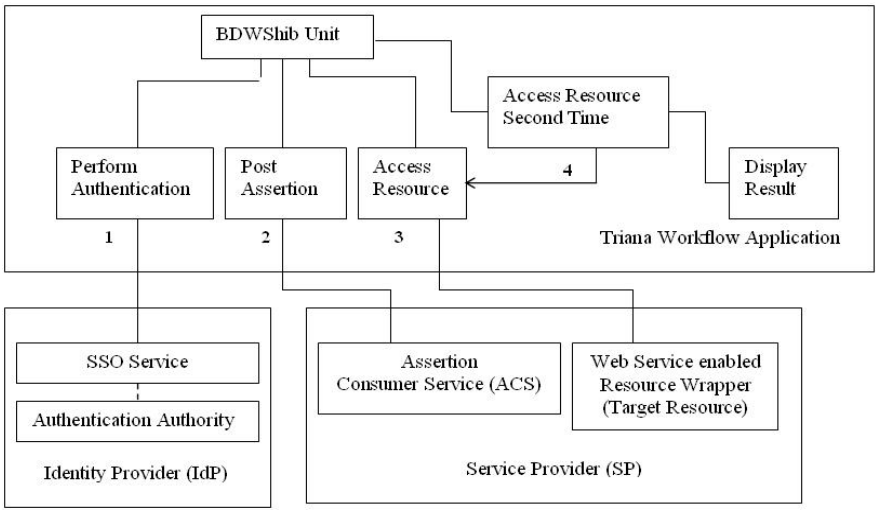

Fig. 1 The BDWShib System Architecture (inside the Triana Workflow Application)

Step 1. When the BDWShib workflow unit is executed by the Triana workflow manager it pops up a window (Fig. 2) inside the Triana application which performs the Single Sign-On (SSO) on user's behalf. The user provides URL of the target resource and username and password for authentication. The 'Perform Authentication' component first accesses the target resource for identifying details such as URL of the IdP, Assertion Consumer Service (ACS) URL, etc. as shown in Fig. 3. The component then performs user authentication by sending the username and password via the HTTPS protocol to the SSO Service. The SSO service processes the authentication request and obtains authentication statements for the user. If authentication is successful the SSO service responds with a digitally signed SAML response containing an authentication assertion which is to be delivered to the SP. The 'Perform Authentication' component gets the authentication assertion from the SSO service on behalf of the user. The other components of the Shibboleth framework at the IdP side such as the Attribute Authority and the Artefact Resolution Service are not shown in the figure.

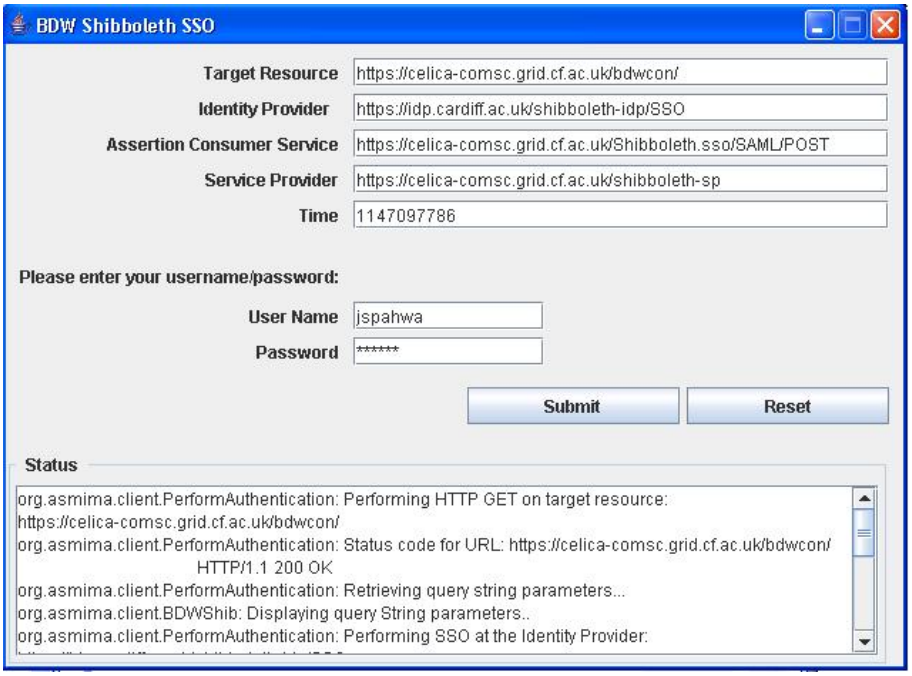

Fig. 2 The BDWShib SSO Window

Step 2. As part of this step, the BDWShib unit, via the 'Post Assertion' component issues a HTTP POST request to the ACS at the SP. The ACS is presented with the authentication assertion returned by the SSO service in the previous step. The ACS processes the authentication assertion and establishes a security context at the SP. The ACS service finally redirects the user to the target resource. The other Shibboleth framework component at the SP called the Attribute Requester is not shown in the figure.

Step 3. Once the security context is established at the SP in step 2, the 'Access Resource' component is allowed to invoke operations at the target resource. At this stage, we can invoke operations on resource wrapper web services by exchanging SOAP messages with the web services.

As part of this step we share session information with the web service client stubs which uses Apache Axis [9] for invoking operations on web services for resource access by sending SOAP messages over the HTTPS protocol. The Shibboleth session information which is maintained by the BDWShib modules using cookies is provided to the web service client stubs by setting request headers of SOAP messages with cookie information. Therefore when an operation is invoked on a Shibboleth protected web service the web service client is allowed access to the resource because the web service client presents the Shibboleth SP with the valid session information. Hence in the BDWShib system we can share session information which is originally created by using Jakarta Commons *HttpClient* API with the Apache Axis web service toolkit. This sharing of session is not possible using web browsers because of the limitations identified in section A1.4.4.2.

The prototype BDWShib system, in its present version provides access to species data from the ILDIS database. The ILDIS website provides access to the database of legumes where a user can search the database by providing the scientific name of the legume species and retrieve all the data pertaining to the species. The BDWShib module inside the Triana workflow tool provides programmatic access to the ILDIS database for its use when conducting a biodiversity experiment. In the BDWShib system we have also provided support for one operation from the Spice Wrapper web service which retrieves species data from the Species 2000 portal.

Step 4. By using the same security context established in Step 2 more than one operation can be invoked at the target resource without the user having to go through the authentication process again when running a single workflow. This is possible using the 'Access Resource' component which keeps java objects pertaining to resource access alive in the memory until the workflow is terminated. The operations can belong to the same web service or other web services pertaining to different resources which are deployed in the same machine at the SP. After data is retrieved from the web service it can be used for further analysis in the BDW PSE, or it can be displayed to the user.

A1.4.4.4 Implementation of other System Components

The Triana Workflow Units: We implemented a number of workflow units inside Triana which enabled the usage of the BDWShib System inside Triana. Each workflow unit corresponds to a BDWShib system class which provides the functionality of the class inside the Triana application when the unit is executed. These units can be linked together to quickly assemble a workflow to perform a biodiversity experiment which enables user authentication and provides subsequent access to Shibboleth protected BDW web services (ILDIS and Spice Wrapper).

We also created a workflow unit for invoking a Java based web browser application within the Triana application. The Java based web browser application is developed by the open source JDIC Project [10]. The JDIC web browser is used for displaying workflow results inside the Triana application after the workflow operation is complete. For this purpose the corresponding workflow unit automatically invokes and pops up the browser window inside the Triana for displaying results to the user.

Sever side development: In addition to setting up the Shibboleth SP at the server side we also performed the following development activities. We created a resource

access mechanism for the ILDIS database using the Java Database Connectivity (JDBC) API and deployed this access mechanism as a web service using the Apache Axis toolkit inside the Apache Tomcat [11] servlet container. Presently the operation of retrieval of species information from the ILDIS database is supported.

The Spice Wrapper web service which retrieves species data from the portal of Species 2000 project was also set up to be accessed from the Triana in a secured manner using Shibboleth. In addition to the Spice Wrapper and the ILDIS database resource wrapper web services we also deployed all other resource wrapper web services presently available in the BDW PSE to be accessed securely using Shibboleth. Presently there are more than 20 resource wrapper web services which provide access to different types of resources in the BDW PSE. However because of time constraints the access mechanism for accessing resources via resource wrappers and performing operations on them from the Triana system was only developed for two resources (ILDIS database and Spice Wrapper). The access mechanism developed for the ILDIS database and the Spice Wrapper using the BDWShib system provides a proof-of-concept that resources in the BDW PSE can be protected using the Shibboleth framework. Therefore as part of the future work, in a manner we have developed the BDWShib system to access the two resources, the system can be extended to support other resources also. In the server side the resource wrappers are deployed as web services in the Apache Tomcat servlet container.

A1.4.5 System Testing and Demonstration

The individual components of the BDWShib System were tested thoroughly whilst the system was in development and tests were also carried out on the system as a whole. The Shibboleth SP protecting the BDW PSE was also tested by Richard Annett, MATU Service Analyst by using the CU's IdP. We originally planned to test the Shibboleth SP by using the IdP of MATU but we were unable to do so because of firewall restrictions at the MATU side. Although the MATU tested the BDW Shibboleth SP at the server side, it could not run BDW workflow as it required installation of the Triana workflow system and the BDWShib system at their side. Since the Shibboleth SP tests were successful we can assume with a greater degree of confidence that full tests would also have been successful if the Triana workflow application and the BDWShib system were installed at their side.

We gave the demonstration of the BDWShib system on four different occasions. Demonstrations were given to the members of the ASMIMA project team at INSRV, to project investigators and to the lead developer of the Triana project team.

A1.4.6 Outcomes and Results

The outcomes and results of this work package can be summarised as follows:

- The BDWShib System, which provides a non-browser based mechanism of performing user authentication and subsequent resource access from the Triana workflow application.
- Two fully functional workflows which demonstrate a proof-of-concept implementation that the Shibboleth security framework can be used for protecting BDW resources.
- Triana workflow units which allow usage of the BDWShib system inside the Triana workflow application.

- The server side infrastructure including the Shibboleth SP installed for protecting BDW resources and web services for resource wrappers.
- Based on the work we did as part of this work package we also wrote a paper [12] which is presently submitted to organising committee of the UK e-Science All Hands Conference 2006 for review. In the paper we stressed the importance of providing access to Shibboleth enabled web accessible resources from desktop-based applications and identified how the BDWShib system provides such access from the desktop-based Triana workflow application.

A1.4.7 Lessons Learned

As part of this work package we learned several valuable lessons.

- We learned that that setting up Shibboleth SP infrastructure and addressing all the security issues associated with it such as creating digital certificates and configuring trust relationship between SP and IdP can take between 6-8 weeks for a full time employee who is new to this area.

- We have learned that the standard implementation of Shibboleth SP is straight forward (however not easy) and help is always available from Shibboleth engineers at INSRV and Shibboleth users' community.

- We learned that using Shibboleth for providing a secure access to resources is not always a straight forward exercise particularly when the secure access is provided from a desktop-based application. Developing the BDWShib system had been a challenging exercise as in the system we created a mechanism of accessing Shibboleth protected resources from the Triana which is a desktop-based application and overcame challenges identified in sections A1.4.4.1 and A1.4.4.2. This required advanced level of understanding various applications and toolkits (and in addition to the Shibboleth framework) used in the system such as the Triana system, Jakarta Commons *HttpClient* libraries, Apache Axis web services toolkit, Apache HTTP server, Apacht Tomcat servlet container, JDBC and Java programming language.

- We learned that sharing of Shibboleth session information is possible between different applications. For example by sharing the session information created using Jakarta Commons *HttpClient* libraries with the Apache Axis web services toolkit at the client side we created Shibboleth-aware web services client which could interact with Shibboleth protected resources over HTTPS protocol without any problems. This sharing of session information provided us with a mechanism of invoking Shibboleth protected BDW resources more than once (from different application toolkits) by different workflow units without having to re-authenticate the same user again as part of running a single workflow.

- We learned that for application domain such as the BDW whose users may be located anywhere in the world, to benefit fully from the Shibboleth framework requires international take-up of the Shibboleth. We believe that this requires a two-way effort. On one hand we need the adoption of Shibboleth by a greater number of users/organisations worldwide and particularly those who are interested in using the BDW System.

On the other hand this also requires extension to the current BDWShib system so that it can handle authentication requests from those new users and then allow access to the BDW resources. This will require linking the BDW SP to a federation. Presently the BDWShib system is linked (as part of a bilateral setup) to the CU's IdP and authenticates CU's users only. As part of this work package we also aimed to test the BDWShib system by linking the BDW SP with the IdP from the Reading University who are our project partners in the BDW project. Unfortunately we could no do so because we could not find Reading University as early adopters of the Shibboleth technology in a list of early adopters provide by the MATU in their website.

- We learned that a wide variety of expertise, collaboration and input from different people is required for running a project of this nature. We collaborated with members from the Welsh e-Science Centre (and particularly the manager of the centre Alex Hardisty) who were extremely helpful during the brainstorming sessions when finalising the design of the BDWShib system. We took advice from developers of the Triana team when integrating the BDWShib system with the Triana system. Shibboleth engineers at INSRV helped us to set up the link between our Shibboleth SP and CU's IdP. With the help of Richard Annett from MTAU and Rhys Smith at INSRV we were able to test our Shibboleth SP installation from an external source. Finally the ASMIMA project members were helpful in giving their valuable advice on many aspects of the system during the fortnightly project meetings.

A1.4.8 Conclusions and Further Work

The aim of this work-package was to develop a proof-of-concept system which allowed a secure access to resource wrappers available in the BDW PSE via the Shibboleth security framework. As part of this system, at the server side we have deployed the BDW resource wrapper web services which are protected by the BDW Shibboleth SP. At the client side we have created the BDWShib system which is integrated with the Triana application and provides a mechanism of user authentication and subsequent resource access from inside the Triana application. We also did development work in areas identified in section A1.4.4.4 to make the system fully functional. We believe that this work packaged completed successfully and we achieved our objectives. At the same time we also believe that further work is required to be done in this area. Although we have developed a proof-of-concept system but because of time constraints we could not provide access to all the operations available in the Spice Wrapper and ILDIS database from the Triana system. We deployed all the resource wrappers available in the BDW PSE for its secure access via the Shibboleth framework but we are yet to develop its secure access mechanism from the Triana system. This requires further development to the BDWShib system to support all the resource wrappers in the BDW PSE. We aim to do this work as part of further activities in this area given the time and resources.

Presently the BDW resource wrappers are hosted locally in a BDW server at Cardiff where we have set up the Shibboleth SP. As part of future activity we also aim to take advantage of storage and computing facilities available in the Welsh e-Science Centre by setting up the Shibboleth SP and hosting the BDW resource wrappers in the Welsh e-Science Centre machines.

Glossary

1. **Service Provider (SP):** A Shibboleth SP (formerly called a target) is a part of Shibboleth security framework which manages secured resources.

2. **Shibboleth Identity Provider (IdP)**: A Shibboleth IdP (formerly called an origin) is a part of Shibboleth security framework which maintains user credentials and attributes. Upon request the IdP asserts authentication statements or attribute statements to service providers.

3. **The Assertion Consumer Service (ACS):** The ACS (formerly called a SHIRE) is the service provider endpoint of the SSO exchange. It processes authentication statements returned by the SSO service at the IdP. If the authentication is successful it redirects the client to the desired target resource.

4. **The Security Assertion Markup Language (SAML):** SAML which is developed by the Security Services Technical Committee of OASIS, is an XML-based framework for communicating user authentication, entitlement, and attribute information.

References:

[1] C. Goble, C. Wroe, and R. Stevens, "The myGrid project: services, architecture and demonstrator," in Proc. UK e-Science All Hands Meeting 2003 (AHM'03), Nottingham, UK, Sept. 2003. [Online]. Available: http://www.nesc.ac.uk/events/ahm2003/AHMCD/pdf/128.pdf

[2] P. Jithesh, N. Kelly, S. Wasnik, P. Donachy, T. Harmer, R. Perrott, M. McCurley, M. Townsley, J. Johnston, and S. McKee, "Bioinformatics Application Integration in GeneGrid," in Proc. UK e-Science All Hands Meeting 2005 (AHM'05), Nottingham, UK, Sept. 2005. [Online]. Available: http://www.allhands.org.uk/2005/proceedings/papers/335.pdf

[3] I. Taylor, M. Shields, I. Wang, and R. Philp, "Grid Enabling Applications using Triana," in Workshop on Grid Applications and Programming Tools, Seattle, USA, 2003. [Online]. Available: http://users.cs.cardiff.ac.uk/Ian.J.Taylor/CV/Papers/GAPT 2003.pdf

[4] J. S. Pahwa, P. Brewer, T. Sutton, C. Yesson, M. Burgess, X. Xu, A. C. Jones, R. J. White, W. A. Gray, N. J. Fiddian, F. A. Bisby, A. Culham, N. Caithness, M. Scoble, P. Williams and S. Bhagwat, "Biodiversity World: A Problem-Solving Environment for Analysing Biodiversity Patterns," in Proc. 6th IEEE International Symposium on Cluster Computing and the Grid (CCGRID 2006), Singapore, May 16–19, 2006.

[5] J. S. Pahwa, R. J. White, A. C. Jones, M. Burgess, W. A. Gray, N. J. Fiddian, T. Sutton, P. Brewer, C. Yesson, N. Caithness, A. Culham, F. A. Bisby, M. Scoble, P. Williams, and S. Bhagwat, "Accessing Biodiversity Resources in Computational Environments from Workflow Applications," Workshop on Workflows in Support of Large-Scale Science (in conjunction with the 15th IEEE International Symposium on High Performance Distributed Computing), Paris, France, June 20, 2006.

[6] (2006) The International Legume Database & Information Service (ILDIS) web site. [Online]. Available: http://www.ildis.org/

[7] (2006) The Species 2000 Portal. [Online]. Available: http://www.sp2000.org/

[8] (2006) The Jakarta Commons *HttpClient.* [Online]. Available: http://jakarta.apache.org/commons/httpclient/

[9] (2006) The Apache Axis website. [Online]. Available: http://ws.apache.org/axis/

[10] (2006) The The JDesktop Integration Components (JDIC) project website. [Onlile]. Available: https://jdic.dev.java.net/

[11] (2006) The Apache Tomcat website. [Online]. Available: http://tomcat.apache.org/

[12] J. S. Pahwa, R. J. White, A. C. Jones, W. A. Gray, A. R. Hardisty, R. O. Smith and N. J. Fiddian, “A Secure Model of Accessing Shibboleth Protected Biodiversity Resources from Desktop-Based Environments”, in Proc. UK e-Science All Hands Meeting 2006 (AHM’06), Nottingham, UK, Sept. 2006 (presently submitted for review).