



UCISA-Infrastructure Group Case Study

Edinburgh does it with EASE (a single sign on implementation for web applications)

1. Introduction

The University of Edinburgh is a research and educational institution with over 25,500 students and 7000 staff. The structure of the University is devolved into three Colleges and several support groups. A wide variety of IT services and technologies are provided across the University by Information Services and also within Colleges and their Schools. Reducing the impact on users and support staff managing multiple sets of credentials for services provided by the University was seen as a valuable strategic aim for the institution. In 2001 the University formed the Authentication and Directory Working Group, whose remit includes investigation of authentication to improve cross platform interoperability. In 2002 this group recommended that a project should be initiated to deliver a University-wide authentication service.

The project aims included building a central authentication service capable of being used as an authentication authority for the three main University platforms, which at the outset were Microsoft Windows, Novell Netware and Unix. A review of Novell Netware resulted in its replacement with Active Directory (AD), which utilises Kerberos as its authentication mechanism. However, it was decided that this could not be a total solution for non-Windows platforms, as there was a concern about how Microsoft had implemented their Kerberos services with various extensions. A further major issue was which Microsoft Client Access Licences were required, especially by non-Microsoft platforms, when using the AD KDC for authentication. There was significant unease over the resulting dependency on Microsoft technology and licensing arrangements. For these reasons the proposed solution of maintaining a second authentication source in a vanilla MIT Kerberos service hosted on the Unix platform was pursued. In April 2003 a mixed AD/MIT Kerberos based solution was delivered.

Following this deployment, the Authentication and Directory Working group recommended that the University's Computing Services embark upon a second project, Reduced Sign-on Technology (REST). The aims of REST were to evaluate available technologies that could be used to provide a reduced sign-on capability for web technologies that would integrate with the existing Kerberos service. This evaluation would be followed by the implementation of the chosen technology. This case study aims to discuss the approach taken at to carry this out and deliver the service.

2. Description of the work

Initiation

The development of the web sign-on mechanism was initiated by the University's Authentication and Directories Working Party, a cross-institutional representative technical committee, to investigate web sign-on technologies and select and develop a solution that was the most appropriate for the institution. The objectives defined by the Working Party were:

- To evaluate the relevant technologies that can be used to provide reduced sign-on capability across the IT services of the University.
- To implement an agreed solution, approved by the University committees, based on the outcome of the evaluation.

Subsidiary objectives included:

- To compare the functionality, security and ease of use between X.509 certificates and cookie based solutions.
- To determine whether a single solution is the best option or whether multiple solutions are feasible, sensible and can be used transparently to the end user.

The underlying assumptions were:

- The Kerberos authentication infrastructure established in the University would form the basis of any reduced sign-on solution.
- Reduced sign-on is an appropriate goal for access to the University IT services.
- Any solution implemented for the EASE authentication portal, which resulted from the earlier Kerberos project, will work with Active Directory once the underlying problems with exchange of tickets encountered in the earlier Kerberos project were resolved.

The Working Party acknowledged the diverse range of University applications and users and agreed that having the goal of a single sign-on architecture was 'impractical, inappropriate and in fact inadvisable'¹. A balance was to be struck between ease of use and security considerations. For example, it was acknowledged that certain types of application may always require separate or additional authentication steps if they were particularly sensitive, such as HR and Finance systems.

Planning

The Working Party defined the scope of the evaluation project. The project aim was to look at both cookie-based and certificate-based solutions to web authentication. An additional requirement from the University's Management Information Services (now part of Information Services) was to include the Yale CAS cookie-based system² in the evaluation. This requirement arose from their project to develop a University portal. The initial development system for the project used CAS as the authentication mechanism.

1 http://www.intercom.ucs.ed.ac.uk/ucsinfo/cttees/citc/work/authdirwg/rest/rest_defn.pdf

2 Yale CAS: <http://www.yale.edu/tp/auth/cas20.html>

The REST project was required to evaluate each technology on the basis of the following criteria:

- The ease of use of the product for the end user.
- The ease of implementation and support for the service provider.
- The theoretical security of the model being implemented.
- The likely security of the actual implementation.
- The quality of the product code and documentation.
- The route to ongoing development and support for the product by the supplier.

Phase one of the REST project was to carry out a selection process of appropriate technologies to be evaluated. An initial search of Internet resources started with the Internet2 Web Initial Sign-on website³. Additional searches were made but no better list of resources was found. The list was then pared down to the following for evaluation:

- Yale CAS (a mandatory requirement from the project requirements).
- Washington Pubcookie⁴ because it was widely deployed.
- Stanford WebAuth⁵ because it was being deployed by Oxford University for the uPortal technology used by the University of Edinburgh for the portal development project and also its ability to encapsulate Kerberos ticket credentials in cookies for onward authentication.
- Michigan KX.509 certificate system⁶ and Michigan Cosign⁷ cookie-based system for its close links to KX.509 and its support for Kerberos ticket credential forwarding.

Business Analysis

Following a University decision based on the presentation of this evaluation work, the IT policy body for the University requested that the Michigan Cosign technology was used to implement a reduced sign-on framework providing a centralised authentication portal and password changing service which other portals/web-based services would refer users to for authentication. Specific Business Requirements included:

- Determining how WebCT, the University's VLE, would integrate with this scheme.
- Ensuring that the University's proposed portal technology would work with the selected platform and that interfaces to existing important web services such as staff and student web-mail could be seamlessly integrated.
- Integrate the authentication portal with Athens DA using the current single University-wide permission set. (A directory service would be required if multiple permission sets were required⁸).
- Create system administrator's documentation outlining a security policy that should be followed when creating services requiring secure authentication

3 <http://middleware.internet2.edu/webiso/>

4 Pubcookie: <http://www.pubcookie.org/docs/how-pubcookie-works.html>

5 Stanford WebAuth: <http://webauth.stanford.edu/protocol.html>

6 KX.509: <http://www.umich.edu/~x509/>

7 Cosign link: <http://www.umich.edu/~umweb/software/cosign/>

8 The development of an authorisation directory service was regarded as out of the scope of both the REST evaluation project and the project to deploy the chosen solution.

- Create administrator's portal for creation/deletion/update of host and service principals in the Unix KDCs for those not using AD. (This functionality may not be realisable as it will probably require some form of directory service for authorisation).

Computing Services was tasked with developing and building the service, drawing on their expertise with Kerberos authentication services.

System Analysis and Design

It was essential that the service be highly resilient and scalable to support authentication requests from a very large number of services across the University. The design incorporated the requirement for multiple web login servers across multiple locations. Underpinning this was the Kerberos Key Distribution Centre (KDC) infrastructure, which was also required to be resilient and replicated across multiple sites. The initial design was to have two Kerberos KDCs, one at each of the main University campuses, and two EASE web login servers, one on each campus. The Cosign technology provided a mechanism for each server to replicate their cookie stores to provide a resilient service, so that any web service utilising the service could pick any Cosign server and interrogate it to check cookie credentials for a user. Thus the loss of any one site should not prevent the loss of the central authentications service.

The initial design also incorporated mechanisms to allow the registration of the University's staff and student body to set up their passwords. To provide a simple mechanism for people to register, we provided a leapfrog registration process, allowing people to authenticate using existing credentials on existing services to provide an initial authentication before setting up their new EASE account.

From the outset the aim was to reduce the number of support requests for people who forget their passwords and need to have them reset. The registration process was designed to require users to provide a number of shared secrets from a selection of questions and answers that they provided⁹. These would be used to check a user's identity if they needed to reset their password at a later date. To protect this information and to ensure the security of this personal information, the shared secret application does not store the actual response the user gives but an MD5 hash of the response. Even in the unlikely event of a compromise of the shared secret database the data could not be used by anyone.

The design of this system was widely distributed amongst Computing Support Officers in the University for discussion and refinement before being signed off for implementation by the Authentication and Directories Working Party.

9 Shared secrets: <http://homepages.ed.ac.uk/jaw/papers/shared-secrets.txt>

Build

Two mid-range servers were purchased to implement the web login service. The two KDCs were smaller systems that had already been purchased, built and installed as part of the earlier Kerberos project. We also utilised an existing development server to do the initial build and prototyping.

The service was built using Apache 1.x and the Cosign 1.5.x release from Michigan, later upgraded to Cosign 1.6.x releases to support n-tier authentication. Additionally, a MySQL service was provided to support the registration and the shared secret processes. The CGI scripting of password change and registration functions was written in Perl. The number of supporting applications and software infrastructure was deliberately kept to a minimum to limit the possibility of compromise through vulnerabilities in the software.

Integration

The first major task was to integrate the uPortal infrastructure into the new EASE web login service. The initial implementation of the Cosign software for Tomcat did not include support for n-tier authentication using proxied cookies. Since the portal required not only to be protected by EASE but also to provide onward authenticated access to other services protected by EASE, a mechanism needed to be developed to allow a Cosign protected service to request proxy login cookies on behalf of a user to allow it to do onward authentication. Java developers within the University's Unix Section developed the Java classes to allow this to be supported in conjunction with the Cosign developers at Michigan. This work was carried out during April and May 2004 and by the end of June 2004 a production quality service was ready.

The web-front ends to staff and student mail run IMP¹⁰, which at the time, provided access to a University of Washington IMAP mail service. It was straightforward to implement the Apache Cosign module in the IMP web service and to configure IMP to trust the authentication provided to it by the module. Because Cosign provided a mechanism to distribute Kerberos tickets as part of the authentication process, it was also trivial to allow IMP to authenticate to the back-end IMAP service using GSSAPI.

Acceptance

User acceptance testing was conducted to ensure that the University portal was fully integrated with the new EASE web login service. The functionality of the technology was confirmed but feedback suggested that some of the documentation regarding the registration process was not sufficiently clear and that the registration process was too lengthy and complicated. In light of the comments a revision of the design and documentation of the EASE registration process was undertaken, signed off and published ready for the 2004/2005 academic session in September.

¹⁰ <http://www.horde.org/imp/>

Deployment

The initial deployment was on the MyEd University portal service, the staff and student mail services, the University's IT Call Management Service and the Computing Services intranet. This short list underestimates the true number of services making use of the infrastructure as lying behind the portal were a number of other services such as the web-base student record, Library lending records, access to Athens resources through AthensDA and others. Soon after initial deployment other parts of the University started to deploy the EASE Cosign service on their intranets and web-based applications, for example, the Student Association web-forum and student e-voting elections.

After initial deployment, issues arose regarding the performance of the central portal service, with serious load problems being experienced during the busy start of session period. This was exacerbated by the loss of one of the sites due to building works disruption to power supplies. As the web login service was closely associated with the portal service, investigation into the performance of the Cosign service was undertaken alongside the investigation of the portal. After extensive testing and performance monitoring it was ascertained that most of the performance problems were due to issues within the load balanced network infrastructure hosting the portal service. However, whilst normal load with both resilient sites being available showed no problems in the web login service, there was a potential performance issue if one of the resilient sites was lost. To resolve this each site received an additional system making a total of four web login servers. Thus the loss of any one site should still provide at least two servers capable of sustaining the peak load.

Closure

All documentation for the use of the service is hosted on the web login service itself. This includes end-user documentation and documentation for web site owners who wish to deploy the technology within their service. The web login service documentation is available at:

<https://www.ease.ed.ac.uk/userdocs/whatisit.html>

Project Roles

- **Project Manager:** Keith Farvis ; Responsible for overall project management communications and coordination
- **Technical Architect:** Graeme Wood; Responsible for interpreting business requirements and developing solution design
- **Implementation staff:** Graeme Wood, Gavin Gray, Rosanna McInerney; Responsible for the implementation of the services
- **Testing staff:** Rosanna McInerney, Gavin Gray, Martin Campbell, Richard Good; Responsible for proving the prototype and ensuring that integration with the University Portal was feasible.

Project Costs

The initial development of the proxy support code involved one person working with the Cosign developers at Michigan for two months to extend the protocol, deploy new server code and test the infrastructure to support the authentication services within the portal service. Additionally, one person was employed for two months to deploy the web login service itself and prepare documentation. This was carried out by a summer student.

Hardware costs involved the purchase of four dual core AMD web servers each with 8GB of memory. The total purchase cost was less than £10K.

Ongoing and recurrent Costs

- **Implementation costs per site**

For a system administrator accustomed to the single sign-on (SSO) technology, setting up a web site to make use of the University's SSO solution requires approximately one hour of hands on time, but due to the requirement of the University's auditors to ensure that each website is audited and separately registered by Information Services this can take up to a day or two. Nominally however, websites can be brought online within a day. For a new web site, whose administrator has not implemented SSO or executed the procedure before, there can be additional time delays whilst they familiarise themselves and compile the necessary filter software. Where problems have occurred it has on occasion to diagnose, especially using the IIS and Java versions of the filter.

- **Recurrent costs per site**

There are currently no recurrent costs for the use of the EASE web login service. Once enrolled the service is provided free of charge.

Operational issues and observations following service deployment

The benefits

- Common authentication framework therefore allowing developers and customers to focus on delivering business requirements.
- Better user experience and ease of use.
- Ability to deliver services seamlessly through a University portal.
- University institutional branding and shared user experience.
- Confidence in the security of the authentication solution.
- Framework for emerging technologies such as Shibboleth.
- Reduced user access control management for participating applications.
- Reduced user support overheads with end user self management of credentials.
- Simplification of the mechanisms used to offer authorisation to web applications to a consistent and universally applied set.
- The ability to deploy large scale web applications that require authentication to the University population with little or no overhead.
- No consumption of the user's real password as the authentication system is based on cookies granted by the central EASE service and verified by each application independently.

How have the benefits been realised

- A great many web applications have been migrated to authenticate using EASE but SSO for all services has not been delivered. Vendor package integration is a significant challenge. For those applications that have been adapted virtually no support requests are received relating to passwords and access to these services. Any application that does wish to adopt EASE is immediately able to authenticate all of the registered users of the service. This is a very attractive feature.
- For some applications that might be regarded as more sensitive a second independent authentication challenge has been requested and provided.
- Users do find the benefits to be very positive with a reduced set of credentials to remember. Feedback from users is generally very supportive and positive.

- Executing an enterprise deployment such as the University Portal would have presented considerable user management challenges without an established and trusted authentication mechanism.
- Systems that have not adopted the SSO solution continue to be plagued by problems relating to forgotten passwords. This is in all cases frustrating for the user, a barrier to them executing their role and a deterrent to using services that would otherwise be helpful. The costs associated with supporting this are considerable and can be overcome with a SSO solution.

The SSO deployment challenges

- Creation of a Single Sign on Service has created the potential for a new single point of failure. Disruption of the SSO service can and does have very serious implications for dependent services.
- Delivering required resilience and performance has been a challenge with a great deal of effort being applied to ensure that the infrastructure can deal with very heavy user demands placed on the service that are exhibited in a very “peaky” manner throughout the year.
- Ensuring individual services don’t compromise overall security through misuse of SSO.
- Vendor package integration can be challenging and without cooperation or an established mechanism to “trust” an external authentication authority potentially impossible to realise.
- Developing a shared secrets mechanism that was sufficiently comprehensive yet user friendly to avoid alienating the user base.
- Delivering and maintaining an appropriate Identity Management solution to support SSO is essential to allow efficient provisioning of new users in the SSO system, granting access to the services in a timely manner.
- A number of differing views on the nature of what constitutes a secure authentication service have been expressed. This has made the roll out of the service more challenging than would have been the case had all stakeholders a common understanding and agreed priorities.

Operational issues that have come to light

- Integration with load balancing technologies and managing session.
- Consistent performance delivery
- Capacity planning and testing.
- Loss of confidence in portal and SSO when problems do occur
- Logging out of a single application correctly
- User confusion resistance to register for the service
- The technology does not support all web servers deployed on all operating systems. Apache running on windows is not supported for example.
- Due to the dependency of SSL and open standards, vendors who have provided solution based on bespoke libraries may not be obliged to support the technology once integrated.
- Any loss of such a central service can have very significant impact on the dependant systems. Resilience and scaling is essential.
- Expiry of X509 certificates will disrupt the service rather than warn the user.

Have we found it impossible to SSO any services?

- Certain externally hosted services are not incorporated due to restriction that the vendor has imposed.
- A number of package software solutions from major vendors would require considerable re-development to incorporate due to the vendor specific ways in which authentication is handled.
- Any service that aims to reach an audience that cannot be authorised centrally cannot use this service. This may change in the future with services like Cosign Friend being offered that allow a self registration facility.

What changes have we been forced to make to the protocol in order to achieve Enterprise SSO?

- We required to extend the Cosign SSO protocol such that proxy capability was included. This was necessary for services like the University Portal to function correctly, this was achieved in collaboration with Michigan University.

Is user experience important?

- Yes vitally so but education is also essential so that key features such as security are not lost sight of in a bid to make SSO easier to use.
- Maintaining customer confidence in solution is also essential
- Ensuring that the registration process and the subsequent self management is as transparent as possible to very important.

What have we learned about running the service and integrating our services?

- User requirements are primary to the service being accepted
- Resilience and scalability must be measured assessed and managed in line with uptake of the service
- A strong partnership between institution IT applications and infrastructure support staff is required to ensure that the technology penetrates
- It is possible to achieve wide applicability of SSO, basing the solution on open source and/or standards
- A portal framework makes most sense when the applications (channels) within it are delivered using SSO
- Deployment of Enterprise SSO can present challenges and opportunities that can be quite unexpected.

What have we disagreed about and how have we reconciled this?

- Choice of SSO solution initially chose Yale CAS now Michigan. In the end went with option which we believed was most acceptable to widest group of service providers at the University.
- Philosophical views regarding the use of a single password and username for multiple services and the risks associated with the loss of this information to multiple services. This has been mitigated to some degree by offering multiple challenges to services that have been designated as

attractive to combine into a portal environment but not one that would immediately allow a user access to sensitive data without a further verification of identity.

What do we still need to do to improve the service?

- Integrate more services with SSO
- Take advantage of latest Cosign functionality
- Fully leverage Shibboleth
- Replace our home grown Identity Management System with a sustainable open source or commercial solution.
- Implement the Cosign Friend functionality to allow greater access to University Services to student family members etc.

Is there anything else that we should consider or recommend for the future?

- Improve the proxy cookie functionality such that services are offered as they are required rather than all at once on login.
- For other institutions approaching SSO seriously for the first time the most important advice is to approach SSO from the standpoint of the entire enterprise not that of a single technical group or supplier. Avoid taking an overly simplistic view of what you think the customer needs or will accept. By understanding your business and technical requirements, your stakeholder view and your customer needs you will be in a great position to really deliver sustainable benefits with SSO

3 Conclusion

An effective SSO solution is more than simply an authentication system for users, it requires:

- Consideration to be given to user needs in terms of documentation, ease of use, integration.
- A robust resilient and scalable infrastructure to cope with the range of demands placed on an enterprise authentication service
- A strong partnership between institution IT applications and infrastructure support staff
- Widest possible applicability of SSO, e.g. base solution on open source and/or standards
- A platform such as portal framework to deliver services to users thereby maximising customer benefits of SSO
- Consideration of deeper technical issues such as the incorporation of technologies like load balancers, ssl accelerators.
- A good security model that can be accepted and trusted by services providers and customers
- A good service measurement and monitoring framework with regular publication of performance indicators to manage service growth and predict peak periods of demand.

4 For further information contact

Iain Fiddes iain.fiddes@ed.ac.uk

Graeme Wood graeme.wood@ed.ac.uk