Reducing SOA Identity Fatigue through Automated Identity Testing

By Crosscheck Networks

I. Introduction

Identity Management plays a pivotal role in securing Web Services-based applications in a Service Oriented Architecture (SOA). Enterprises are under unrelenting pressure to streamline business processes by integrating internal systems and external trading partners. As enterprises integrate applications internally and across corporate boundaries, identity management systems are being tasked to extend identities, roles and access controls seamlessly across internal and external applications.

The need for flexible identity management is primarily driven by rapid Web Services proliferation and security standards adoption. WS-Security 1.1 (WSS-2004) has introduced message-based identity representation and use. Such message-based identities can easily traverse corporate and system boundaries. These Web Services Security standards that promise relief for SOA applications developers concerned about security, result in grief for IT personnel concerned about testing identities. Without guidance and tools that test WS-Security 1.1 identity profiles, this grief in turn results in *Identity Fatigue Syndrome*.

II. Symptoms of Identity Fatigue Syndrome

Identity Management solutions already exist in the User-to-Application space and provide robust Single Sign On Solutions (SSO) that enable users to access various business applications with a single identity token [1]. The following are key characteristics of identities in the User-to-Application domain:

- *Transport-level credentials:* Typical user credentials in SSO include HTTP Basic Auth, SSL Auth, Two Factor Auth, and Kerberos.
- *Authentication*: Prove user identity based on supplied credentials.
- *Authorization*: Control access to resources, typically URIs, based on user credentials.

Key Points

Increased complexity in testing Web Service identities

No standard methodology to test Web Services identities, roles and access Controls

Automated Regression key to reducing fatigue associated with testing of identities

Diagnostics Software will play a pivotal role in successful deployment of SOA applications

These characteristics of the identity management create a burden in successfully integrating browser-based clients securely with business applications since there are no standard tools that test all aspect of user-based identities for access control permissions. Instead of providing comprehensive testing tools for deploying Identity management systems, identity product vendors provide rudimentary trouble-shooting guides on how to diagnose access characteristics. For example, if an authorized user is denied access to a URI, detailed access-log and configuration audit-log analysis reveals improper memberships or resource restrictions. At best, identity vendors supply the customer with ad-hoc tools and manual methodologies to test pre-deployment access control functionality. The identity-testing crisis is further exacerbated by testing tools and testing methodologies lagging behind new identity product releases [2].

Today, the SOA paradigm based on Web Services has ushered in an era of loosely coupled and chained Web Services-enabled applications [1]. To integrate these applications securely and with minimum friction, new Web Services Security (WSS) standards have been introduced. These new standards enable existing transport-based identities (HTTP Basic Auth, SSL X509s, Kerberos) to traverse application and corporate boundaries. The WSS standards have increased the identity testing challenge because of several factors:

- ➢ WSS standards support enveloping of exiting and new identities such as, Username tokens, X.509 certificates, Kerberos tickets and SAML assertions using flexible message-based formats.
- ➢ The above identities can appear in XML messages over diverse protocols such as HTTP, HTTPS, JMS, SMTP, and IIOP [1].
- ➢ Critical business functions exposed by Web Services require fine-grained access control (authentication, authorization) based on these identities.
- ➢ Compliance and regulatory pressures mandate that all proper access controls are in place across the enterprise applications.

The above factors increase the workload on developers and QA personnel, already inundated with developing, testing and deploying Web Services-based applications. There is no standard testing methodology defined by a standards committee for the various identity standards in a Web Services framework. Recently, there has been an attempt by Web Services-Interoperability Organization (WS-I) to provide guidelines – through Basic Security Profile (BSP) – for interoperability among various implementations of WSS that utilize identity tokens [3]. The WS-I BSP is a step in the right direction. By promoting interoperability, automated testing tools can be developed for comprehensive identity token testing across diverse applications.

III. Remedy through Identity Testing

There is no silver bullet to eliminate the challenge of testing the numerous scenarios across protocol- and message-based identities within a Web Services-based SOA. However, certain steps can be taken to alleviate the fatigue that can result in dealing with various identity mechanisms associated with Web Services. A successful identity-testing plan should:

- ➢ Include an identity diagnostics tool that mimics every Web Services client in the Web Services chain. This Web Services tool should be fully capable to generate all kinds of identities for positive and negative testing. The negative testing can be achieved through mutation of identities and conditional tests.

- Implement Identity Management tasks, such as identity generation and validation. Such tasks should be a part of functional regression testing, performance testing and compliance assessment.

- Deploy strong reporting functionality for rapidly isolating trouble spots during various stages of identity testing.

- Include a diagnostics tool that mimics every Web Services client and acts as an independent auditor. The richer the reporting, the easier it is for compliance officer in an enterprise to audit the access controls in place for Web Services.

Figure 1 is an example of how a SOA tester could utilize automated diagnostics software from a desktop for complex identity management testing. Figure 1 shows a typical system that involves authentication of a user whose identity traverses across the enterprise applications.

- In step 1, a user authenticates into a browser-based portal with a built-in Web Services client.

- In step 2, after the user is validated, based on his or her credentials, a subsequent SAML assertion is generated that is then signed by the Web Services client. SAML (Security Assertion Markup Language) is an XML based security token standard that carries authentication and authorization information [4]. SAML is used to envelope other identities and project them across application and enterprise boundaries.

- The Web Services client then takes two actions. It sends the requests to the back end Web Services provider, as shown in step 2, while also logging the SOAP message to a central repository as shown by step 3.

- In step 4, the software diagnostics running on the desktop pulls all the SOAP messages from the repository through a batch process. It then applies its two regression policies, Automated Positive Regression and Automated Negative Regression on the batched SOAP requests. Both these policies are invoked on the batched SOAP messages through automated tasks (validate SAML, transform, send, analyze). The positive and negative test suites are launched against the target Web Service as shown in step 5.

- The resultant requests are collected for analysis as shown in step 6. Such tests ensure the reliability and robustness of a backend Web Service Provider as to how it handles Web Services access controls based on complex identities. This is an example of how diagnostic software should handle identity testing and alleviate the tester from manually testing various identity combinations.

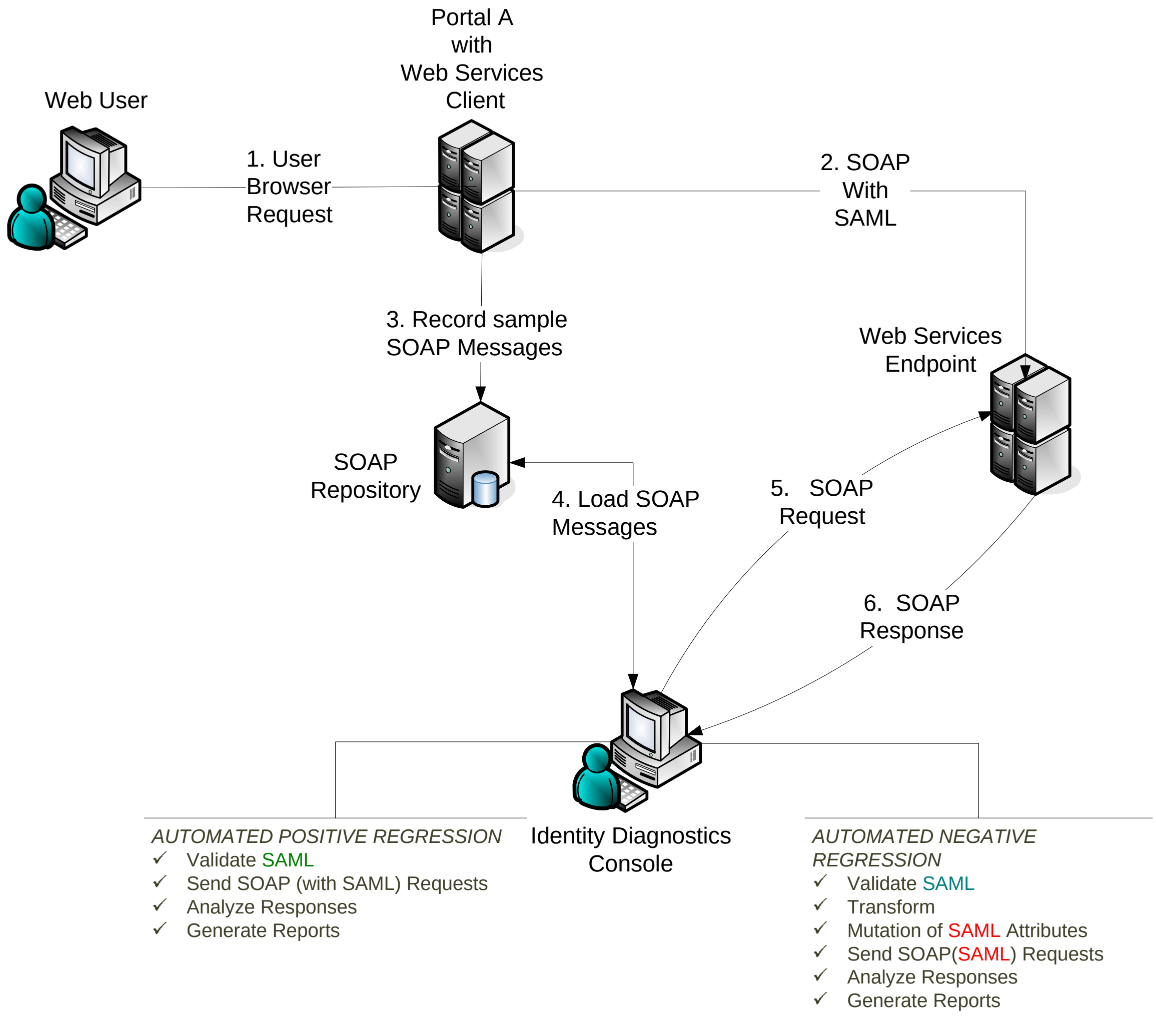

Figure 1: Automated positive and negative regression testing for Web Services Identity.

IV. Conclusion

Identity management systems remain tightly coupled with Web Services deployment to ensure tight access control based on multiple identities. As Web Services applications move towards external partner integration, testing of identities becomes even a more complex task. Robust and reliable access control is only possible through automated regression of identity testing at every point in the Web Services chain. SOA managers are responsible for providing comprehensive software diagnostics tools and methodologies early in a SOA projects development life cycle. By deploying such tools and techniques, SOA Managers ensure that their Web Services-enabled applications are less prone to information theft through access control breaches.

References

[1] Mamoon Yunus and Rizwan Mallal, “Identity Bridging Techniques across SOA-based Business Service Networks”, March 2006 (IEEE publication pending).

[2] Jerry Gao, PhD, “Testing Component-Based Software”, San Jose State University

[3] WS-I, Basic Security Profile Version 1.0, Working Group Draft, Dec 2004

[4] John Hughes, Eve Maler, et., Security Assertion Markup Language (SAML) v2.0 Technical Overview, Working Draft, September 2005, section 2.

Contact Information
General: info@crosschecknet.com
Sales: sales@crosschecknet.com
Phone: 888.276.7725 Fax: 707.988.3840
25 Thurston Road · Newton, MA 02464 · USA