Improving Web Services Security Models

Sawsan Abu-Taleb and Hossam Mustafa
Al-Balqa' Applied University, Jordan

Abstract: *Web services are considered one of the main technologies which emerged in recent years, they provide an application integration technology that allows business applications to communicate and cooperate over the Internet. Web services encouraged existent architectures to adopt as one of the most important technologies; Portals, providing content aggregation from various web services sources for providing useful information to users. The distributed sources of web services aggregated into users' pages provide a component model architecture, which allows the plugging of components in infrastructure, which are referred to as portlets. This paper defines effective models for securing portlet contents by defining an access control list for each portlet, which will looks into the access control of web services, and authentication of web services consumers. In addition, this paper introduces a design for trusted authority that will be responsible for fair contract exchange between portlet producers and consumer; thus, defining a single sign-on model, which is responsible for authenticating remote portlets requests.*

Keywords: *Web services, WSRP, single-sign-on, security, and e-contract.*

Received July 23, 2008; accepted May 17, 2009

1. Introduction

Web Services (WS) are considered one of the main technologies which emerged in recent years, it provides an application integration technology that allows business applications to communicate and cooperate over the internet. WS encouraged existent architectures to adopt as one the most important technologies; Portals, providing content aggregation from various WS sources for providing useful information to users. The distributed sources of WS aggregated into users' pages provide a component model architecture, which allows the plugging of components in infrastructure, which are referred to as Portlets. This article defines effective models for securing portlet contents by defining an access control list for each portlet, which will looks into the access control of web services, and authentication of WS consumers.

In addition, this article introduces a design for trusted authority that will be responsible for fair contract exchange between portlet producers and consumer, thus defining a single sign-on model, which is responsible for authentication remote portlets requests. Portals offer many services providing a huge amount of information to the user, many of these portals, also they provide personalized versions, and such portals allow the user to have one or more personal pages composed of a number of personalized services. Usually, the user can personalize a lot of aspects, such as the layout of services in personal pages and page skins [3].

One of newest technologies which emerged in the recent years is the WS paradigm. It is an important mechanism for interoperation among the separately developed distributed applications in such a dynamic e-business environment. WS interacts as remote procedure calls RPC distributed over the internet, these procedures (services) will defer in functionality according to the applications requirements. On the other hand, Security is one of the major concerns to be taken into consideration when developing online business applications, this concern motivated the WS security specifications [7]. These security specifications include authorizations, authentication and integrity of information that can be sent and received by the business parties. Thus; integrating services together in a single interface will introduce an overhead processing in authenticating users and services. Security model for such applications should be portable between several technologies that may be used as a service.

Because of the lacks of standardization for security mechanisms and Interoperability of portals, this article propose a system architecture for portlets that can help to avoid several security drawbacks, and moreover, it can help in the design of portlet specifications, and merging the suitable security tokens into the portal architecture. The system architecture can provide detailed security tokens, proposed models for portlet authentication and authorizations.

2. The Problem Definition

Interaction of the services in web portals is that a user interacts with the user interface server, which maintains client proxies to the Universal Description, Discovery, and Integration (UDDI) [6] and SOAP

Service Providers (SSP). Each of these runs on a separate web server.

Simple Object Access Protocol (SOAP) allows applications to bind to other applications in order to make user of their functionality. SOAP is used to send data from one application to another applications, so it is sometimes seen as a messaging protocol as well as a means of using functionality that published by a remote application.

Web Services Description Language (WSDL) separates the operations supported by a service and the definition of their input and output messages from its mappings to available deployed implementations. The UDDI protocol [6] designed to allow WS to be easily located and subsequently invoked; also, UDDI maintains links to the service providers' WSDL files and server URLs. The client examines the UDDI for the desired service and then binds to the SSP. The SSP in turn acts as a proxy to some backend services.

This approach introduces a separation between the server that manages the user interface and the server that manages a particular service. This separation is not presented in the three-tiered portal model; it is considered the key development for breaking the portal stove pipe. The User Interface server can potentially bind to any SSP. By using SOAP and WSDL universally, the portal services can be encapsulated and invoked independently of the implementation. The significance of portal applications stems not only from being a handy way to access data but also from being the means of facilitating the integration with third party applications. This has led to the so-called portal imperative: the emergence of portal software as a universal integration mechanism [5].

The Key to this view is the notion of portlet. Portlets are applications within a portal in much the same way as servlets are applications within a Web server. The difference stems from portlets being multi-step, user-facing applications. They are very much like Windows applications in a user desktop in the sense that a portlet renders markup fragments that are surrounded by a decoration containing controls. The portal page then, contains a number of portlets whose fragments can be arranged into columns and rows, and minimized, maximized, or arranged to suit the user needs.

Information contained in one portlet will surely be required in another, and thus forcing the individual users to manually copy data from source to target portlets leading to frustration, losing productivity, and inevitable mistakes. And this situation certainly hinders the fulfillment of the portal imperative.

Portlets which pertain to distinct producers remain isolated. On the other hand, the API-based approach facilitates a programmatic interface for portlets to communicate their state to interested parties. Unfortunately, there is not yet an agreement on how to standardize this mechanism.

A portlet application contains resources that can be accessed by many users. These resources often traverse unprotected, open networks such as the Internet. In such an environment, a substantial number of portlet applications will have security requirements.

The portlet container is responsible for informing portlets of what the user roles are when accessing them. The portlet container does not deal with user authentication. It should leverage the authentication mechanisms.

3. Related Work

There are only few approaches that could be compared in our security models, since the WSRP specification is a new technology. Java Portlet Specification JSR-168 [1] introduced portlet specification with little security concern. WSRP [8] on the other hand, provides interfaces for implementing remote portlet between several technologies, and try to enable an application designer or administrator to pick from a rich choice of compliant remote content and application providers, and integrate them with just a few mouse clicks and no programming effort.

WS security, moreover, provides huge security researches, these researches are the main concerns in role based web services, access control web services, authorization using WS. [4] Describes a formal semantics for WS-security policy, and propose a more abstract link language for specifying the security goals of WS and their clients. [11] Provide a language by which can be expressed and enforced automatically, portably and efficiently security policies.

Microsoft corporation and Sun Microsystems proposed a Web single sign-on interoperability [2] that defines an interoperability profile of the web single sign-on metadata exchange protocol. This allows using either Liberty Identity Federation or WS-Federation based Identity Providers to interact with a service.

[9] Take a radically different approach to address fair contract exchange problem, which is to apply the idea of optimistic fair contract signing recently, also shows a design of the protocol based on the latest XML and WS Security standards and discusses the benefits and limitations of this approach.

[10] Builds interoperating portal services around a WS model, and presents a comprehensive view of an interoperable portal architecture, beginning with core portal services that can be used to build application WS, which in turn may be aggregated and managed through portlet containers.

4. Proposed Security Models

4.1. Access Control List in the Remote Portlets

Current portals implementation follows the component model architecture that allows the plugging of components in infrastructure. Dynamic and real-time integration in portals will face security drawbacks including authorization of the end users. The remote portlets as defined previously include three parties: the producer, the consumer, and the end user of the consumer. The proposed model will define the data flow between the portlet parties, and in addition, will introduce the required security tokens.

The proposed model determines an access list for the remote portlets that will be provided by the producer, the consumer will manage these access control list. This model is done by the following as shown in Figure 1:

- Additional steps added to the registration interface between the consumer and producer of the portlet that include the security access control list, and policies and conditions required. These steps will control the portlet modes of the portlet.
- Markup interface will be modified, to show only the content assigned by the access control list of the remote portlet.
- Portlet management interface will contain additional configuration to modify and control the access control list of the remote portlet.

Therefore, the producer will provide a set of web service interfaces and by implementing these interfaces, and agreeing to conform to Web Service for Remote Portlets (WSRP) specification, both producer and consumer can use a standard mechanism to offer and consume portlets.

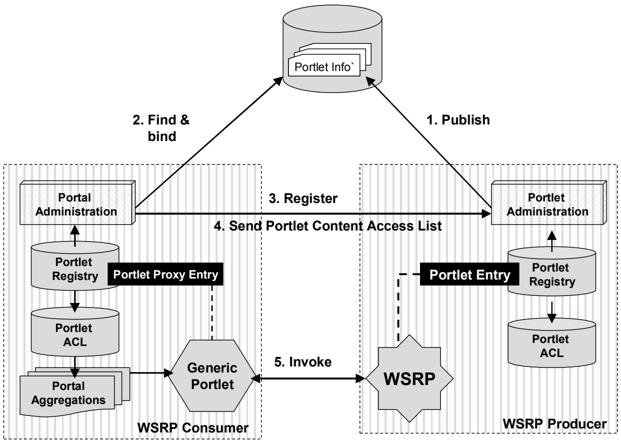

Figure 1. Proposed access control list in the remote portlets.

4.2. Registration Model

Registration can be made by a registration interface on the producer services. This mechanism will be replaced by an e-contract, and fair exchange protocol. The contract between the consumer and producer consists of the following parts: contract initiator: consists of references to contract assertion, and signature of the contract consumer and producer. The contract assertions contain SAML assertions for the consumer and producer, which reference the remote portlet that wants to register on. SAML provides XML-based framework for exchanging security information. On the other hand, producer will provide an assertion for remote portlet details (i.e., registration price, duration, *etc.,*). The contract initiator presents a commitment for the contract. Later, the contract parties will sign this pre-contract by series of request/response operations.

Commitment requests for the consumer and producer: this part consists of the parties requests to sign the contract. This part consists of two main requests: the request from the consumer to the producer to sign the contract, and the request from the consumer to e-contract authority. The final contract will include commitment from the two sides of the remote portlet.

4.3. Single Sign-on Communication Model

This model examines the exchange of authentication between producers as shown in Figure 2 and consumers. Authentication mechanism is based on the single sign-on protocols which can mutually solve the overhead time on the producers and there for will make the response time in the end user portal page dependent on each of the producers service time. That authentication process will be on a partner network.

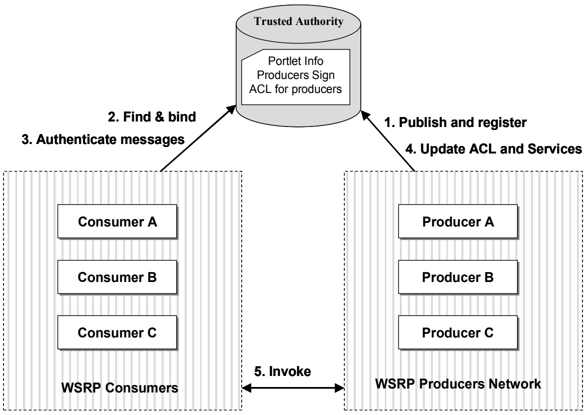

Figure 2. Proposed architecture for SSO between WSRP parties.

The SSO model will implement the existing WS-security specifications, in addition to SSO protocols. The SSO model uses SAML assertion to authenticate the producers. Additional tokens will be added for security strength of the SSO model an example of such security token is the signature of the trusted authority on the authenticated messages.

5. Results

5.1. Performance Analysis

In order to compare between the performance of the proposed single sign-on model with the current authentication and authorization model, the total time from the portal application to the client has been taken as the basis of comparison. The analysis assumes the following assumptions:

- N: number of portlets in the consumer (number of the producers).
- CR: Client Request of portlet page from the portal (consumer) in bits.
- CS: portal response to the client in bits.
- CTS: client connection speed (bits/second).
- AMCL: authentication and access list message in bits.
- SMCL: authentication and authorization message in bits.
- PTS: producer connection speed (bits/second).
- GM: get markup request message in bits.
- RM: response to get markup message in bits.
- TTS: trusted authority connection speed (bits/second).
- SRAT: percentage of the required information on the current model SMCL message.

The assumed network diagram of the current authentication and authorization model of consumers of portlets is shown in Figure 3.

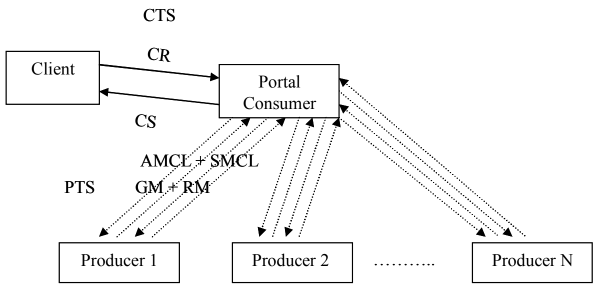

Figure 3. Current authentication and authorization model of consumers of portlets.

The assumed network diagram of the proposed SSO model of consumers is shown in Figure 4.

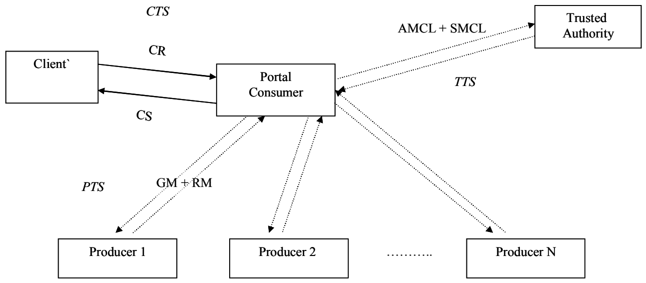

Figure 4. Proposed SSO model of consumers.

5.1.1. Current Authentication and Authorization Model

In the current model, the client will request the page from the portal, and will be receive a response to the request containing the page contents.

The size of the data that sent and received is (CS+CR). The time required in order to send and receive this amount of data is:

$$(CS+CR)/CTS \qquad (1)$$

The portal in the other side of the communication will send an authentication and authorization request from the producers of the portlets contained in the client requested page. The size of the data sent and received is N*(AMCL+SMCL+GM+RM). So the time required in to order to send and receive this amount of data is:

$$(AMCL+SMCL+GM+RM)*(N / PTS) \qquad (2)$$

Then the total time required to send and receive from the client side will be:

$$(CS+CR)/CTS+ (AMCL+SMCL+GM+RM) *(N / PTS) \qquad (3)$$

5.1.2. Proposed SSO Model

In the proposed model, the client will request the page from the portal, and will be receive a response to the request containing the page contents. The size of the data sent and received is (CS+CR). The time required in order to send and receive this amount of data is:

$$(CS+CR)/CTS \qquad (4)$$

The portal in the other side of the communication will send an authentication and authorization request from the trusted authority. The size of the data sent and received is (AMCL+SMCL). So the time required in to order to send and receive this amount of data is:

$$(AMCL+SMCL)/(TTS) \qquad (5)$$

The portal will send a request to retrieve the portlet content from the producers; the size of the data sent and received is N*(GM+RM). So the time required to send and receive this amount of data is:

$$(GM+RM)*(N/PTS \qquad (6)$$

Then the total time required to send and receive from the client side will be:

$$(CS+CR)/CTS + (AMCL+SMCL)/TTS + (GM+RM)*(N/PTS) \qquad (7)$$

The proposed SMCL (response message from the trusted authority) will be increased depending on the number of producers. The size of the message will be calculated according to the current model, so the size of the message will be:

$$SMCL_{proposed} = (SMCL_{current} * SRAT * N) + SMCL_{curren} * (1 - SRAT) \qquad (8)$$

where the $SMCL_{proposed}$ is the size of the SMCL message in the proposed model and the $SMCL_{current}$ is the size of SMCL in the current message. The SRAT parameter defines the percentage of the required information (useful) in the current model SMCL message.

5.2. Experimental Analysis

Three experiments have been made as described bellow in order to compare between the existing and proposed models using the mentioned formulas.

5.2.1. Experiment I

Consider the following values of the common variables:
AMCL = 1 Kb, SMCL = 8 Kb, GM = 1 Kb, RM= 8 Kb, PTS = 10 Mpbs.
CR =1 Kb, CS =10 Kb, CTS = 56 Kbps, SRAT = 0.1, TTS = PTS

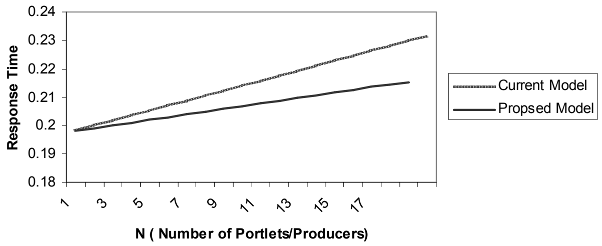

Figure 5. Response time graphs experiment I.

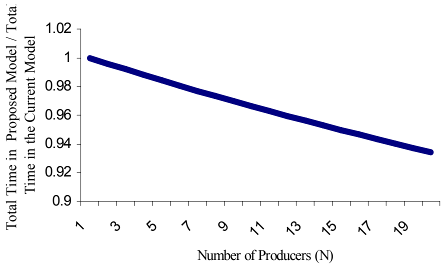

Figure 6. Division graph experiment I.

5.2.2 Experiment II

AMCL = 1Kb, SMCL = 8Kb, GM = 1Kb, RM= 8Kb, PTS = 10 Mbps. CR =1Kb, CS =10Kb, CTS = 56 Kbps, SRAT = 0.1, TTS = 100 Mbps

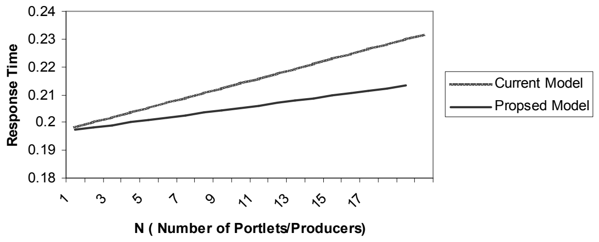

Figure 7. Response time graphs experiment II.

5.3. Experimental Results

From the mentioned experiments and graphs, it is clear that the proposed model provides faster response time when the number of producers increases. Despite the size of the authentication and authorization request message, it will still be faster. This result is valid when the trusted authority interconnections speed is more than or equal to the producers interconnection speed which is a real assumption.

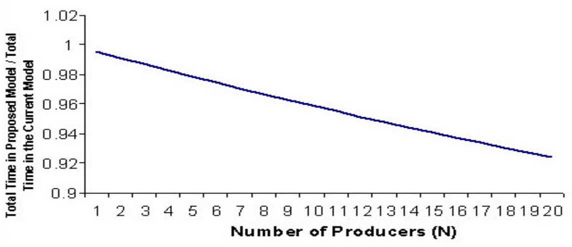

Figure 8. Division graph experiment II.

6. Conclusions and Future Work

6.1. Implications for Remote Portlets Security

WS provide an application integration technology that can be successfully used over the Internet, it allows business applications to communicate and cooperate over the Internet. WS allow objects to be distributed across Web sites where clients communicate and cooperate over the Internet. Registry standards enhance this by defining how the WS may be published, found, and bound with minimal human interaction. Securing these services face many standardized mechanisms. Remote portlets in particular, is ready made content (fragment) distributed all over the web. Controlling these contents will satisfy the security needs of the consumers that provide these services for their end users, and moreover, will transfer the processing time for validating and formatting the content into the producer interface, since it is an originally responsible of viewing, editing, and configuration of the content of these remote portlets.

Authenticating consumers of portlets, in the other hand, makes an overhead response time from the producers, since there are drawbacks in the response time from the remote portlet producers due to communication problems and processing time for each portlet. The designed model takes an advantage of unclear standards of WSRP, and provides the following new features:

- Efficient model for authorization of internal content of the portlets, that will make it possible for consumers to determine such policies and roles, to overcome content filtering according of end user privileges.
- Comprehensive model for centralized authority, which is responsible for all operations regarding remote portlet. These operations were taken on producers, and made producer's web server architecture more complex, adding new security features. This will make producers lack the awareness newest update abandon security standards. Security model designed to assist

producers to leave all operations like authentication, authorization, and registration to be responsible from a trusted third party.

6.2. Common Security Problems

Threats to Web services, pertains to the host system, the application, and the entire network infrastructure. Centralized authority has many disadvantages concerning reliability and on-line profiles, and faces many problems with the digital signature validation of the communicating parties in our model. Consumers and producers will be always worried about updating their security profiles and security policies.

6.3. Problem with Huge Access Control List

Enterprise Consumers will be aware that their ACL profiles will be so huge, depending of how they can divide their end users. Normally, the creation of such profiles will be made once, and updated when consumer policies change.

6.4. The Requirements of the Trusted Authority

The trusted authority that will authenticate and authorize the consumers must the following requirements in order to accomplish the high performance needs:

- High Speed Servers: these servers should be speed enough to handle such large requests from the consumers.
- High availability: the trusted authority should be available online all over the time.
- Combines the functionalities of a CA if a X.509 is used.

7. Conclusions and Future Work

We have presented security models that exploit existing security specifications to introduce such mechanisms for producers of WSRP and the consumers; model for ACL satisfied the consumer needs to filter the fragment of the remote portlet, and provide the suitable roles of each end user. A mechanism for registration for remote portlets is drawn as an e-contract authority. Moreover, this authority will be responsible for authentication of consumers registered on a producer remote portlets. These models assist all parties to reduce the responsibility and time consuming communication.

Our next step will be to design a model for a comprehensive authority, including existing web service security, to act as a certification authority, implementing all XKMS tokens. In the other hand, more specific tokens for Single sign-on will be designed to work for authenticating end users through the producers portals.

References

[1] Abdelnur A. and Hepper S., "Java Portlet Specification," http://www.jcp.org/en/jsr/detail ?id=168, 2010.

[2] Angal R., Kaler C., and Gong H., "Web Single Sign on Interoperability Profile," tp://www. microsoft.com/presspass/press/2005/may05/0513 MSSunFS.mspx, 2010.

[3] Bellas F., Fernández D., and Muiño A., "A Flexible Framework for Engineering," *in Proceedings of the 13th International Conference on World Wide Web*, USA, 2004.

[4] Bhargavan K., Fournet C., and Gordon A., "Verifying Policy Based Security for Web Services," *in Proceedings of the 11th ACM Conference on Computer and Communications Security*, USA, pp. 730-733, 2004.

[5] Delphi Group, "Portal Lifecycle Management: Addressing the Hidden Cost of Portal Ownership," http://www. mongoosetech.com /downloads/ portal_ownership.pdf, 2010.

[6] IBM, Microsoft, Oracle and SAP, "Universal Description, Discovery, and Integration (UDDI v3.0)," http://www. oasis -open. org/ news/ oasis_news2_03_05.pdf, 2010.

[7] Imamura T., Tatsubori M., and Nakamura Y., "Web Services Security Configuration in a Service Oriented Architecture," *in Proceedings of Special Interest Tracks and Posters of the 14th International Conference on World Wide Web*, Japan, pp. 14-22, 2005.

[8] Kropp A., Leue C., and Thompson R., "Web Services for Remote Portlets Specification," http://www. oasis-open.org/ committees/ wsrp, 2010.

[9] Maruyama H., Nakamura T., and Hsieh T., "Optimistic Fair Contract Signing for Web Services," *in Proceedings of the ACM Workshop on XML Security*, USA, pp. 279-289, 2003.

[10] Pierce M., Fox G., and Youn C., "Interoperable Web Services for Computational Portals," *in Proceedings of the ACM/IEEE Conference on Supercomputing*, USA, pp. 83-92, 2002.

[11] Sirer E. and Wang K., "An Access Control Language for Web Services," *in Proceedings of the 7th ACM Symposium on Access Control Models and Technologies*, USA, pp. 3-5, 2002.

Sawsan Abu-Taleb a lecturer in the Department of Information Technology, Al-Balqa' Applied University, Faculty of Prince Abdullah Bin Ghazi of Science and Information Technology, Jordan. Her research interest includes digital image processing, information security and computer graphics.

Hossam Mustafa a senior web developer in the computer center, Al-Balqa' Applied University, Jordan. My research interest includes digital image processing, information security, and semantic web. He received the Master degree in computer science in 2006 from Al-Balqa' Applied University, Jordan.