Architectural Framework for Secure Composite Web Services

J. G. R. Sathiaseelan, Ph.D.
Head, Department of Computer Science, Bishop Heber College,
Tiruchirappalli – 620 017, India.

ABSTRACT

The exorbitant growth in the field of web services technology has challenged the web developers, vendors and researchers to design and development a variety of enterprise web applications for diverse organizations. Since, security is considered to be an essential part in the development of web applications, web services security has also become an emerging trend in Web services technology. Even though there has been considerable amount of research work carried out in these areas, there is no solid scheme offered so far to build a secure academic-oriented web application. Hence, a novel architectural framework is intended solely for the academic institutions with the aim of providing efficient and secure composite web services for the web users. The concept of multi-level security is also included in the proposed framework to handle various security concerns at different levels.

Keywords

Software Architecture, Web Engineering, Composite Web Services, Multi-level Security

1. INTRODUCTION

In recent times, the research in the field of software architecture [1] has emerged with an emphasis on the architectural issues such as domain-specific architectures, software reuse and architectural description languages. The concept of software architecture is to offer constructive high-level design solution to manage the complex problems. Shaw and Garlan [2] have proposed several architectural styles to simplify the architectural design like batch sequential, client-server, layered, object-oriented, pipeline, and real-time and thus provided a new paradigm of software architecture. Meanwhile, the intrusion of Internet has become predominant factor for pervasive and ubiquitous computing which has resulted in an excessive growth in the development of Web-based systems and applications. The proliferation and the variety of Web applications besides their indispensable quality factors have seemed to be more challenging which has paved the way to the emergence of a new discipline known as Web Engineering [3] which intended to successfully manage the diversity and complexity of Web application development. The main objective of Web engineering is to establish and to use disciplined and systematic approaches to successfully develop, deploy, and maintain high quality web applications [4].Seven broad categories of web applications based on their scope and complexity are cited in [5].Web engineering is multidisciplinary in the sense that contributions from a variety of disciplines such as information sciences, multimedia, hypermedia, graphic design, human-computer interaction, network management, simulation and modeling, and software engineering are significant for the successful Web application development, [6]. Ten key steps for the successful development of Web applications are recommended in [7]. A systematic approach has been proposed by Ginige [8] to develop large complex web applications.

Another dimension of Web Engineering is the Web services technology that offers a number of enterprise applications like on-line banking, e-shopping, and web portals. These domains have a greater demand in the software industry to urge the web developers and the software vendors for the design, development and deployment of numerous web applications for the diverse organizations. Nevertheless, there has been no attempt made so far towards the development of academic-related web application. Such a situation has motivated us greatly to develop a framework [9] for academic-related web services. While providing several benefits, web services technology has been facing serious threats like prefix hijacking and interception [10] in the Internet due to a man-in-the-middle attack [11]. Therefore, security has become the key issue in the field of web services technology. This scenario has also made to devise a secure architecture [12] for the previous model. As a consequence, an architectural framework is propose in this paper which is intended for the academic institutions with an aim of providing secure composite web services solely for the academic users. Therefore the multi-level security is incorporated in this architectural framework to handle various security issues like user authentication, message confidentiality, data integrity and authorization of web services at different levels.

The organization of this paper is as follows. Section 2 presents the summary of related work performed in the past. Section 3 describes the design of the proposed architectural framework. In section 4, the security architecture for proposed model is explained. The MLSF protocol required for the different processes for this model is presented and section 6 is the conclusion.

2. RELATED WORK

Currently, most of the business organizations have initiated the design of their own web sites with the development as well as the deployment of their web services over the Internet. Such enterprise web applications are subject to man-in-the-middle attack by intruders or malicious users which may cause severe financial and legal implications to any commercial organization. Hence there is a growing need for the concept of design and development of multilevel security architectures [13-14]. Consequently, a variety of organizations and standards groups like IBM and Microsoft including OASIS, BEA etc. have proposed the Web Services Security specifications. Together, they have made the major contributions for web service security such as WS-Security [15], WS-Trust [16], and WS-Federation [17]. Subsequently, there are quite a lot of research activities carried out so far in the field of Web Services Security. However, each model has been proposed with some limitations.

Yanjiang et al. [18] proposed a password-based user authentication and key exchange system by employing two-server architecture. This system is a password-only system and requires no public key cryptosystem aiming to secure against the offline dictionary attacks. Since the password-based user authentication systems are inexpensive and user-friendly, a password-based user

authentication is incorporated as the first level of security in the proposed model.

Several variants of password-based authenticated key exchange (PAKE) or password-only protocols exist today that do not entail any public key cryptosystem under PKI. Hence, these protocols are more striking for many real-time applications. Bellovin and Merrit [19-20] have pioneered a password-based protocol by founding an encrypted key exchange (EKE) algorithm for securing dictionary attacks. Wu and Zhu [21] provided a solution to key exchange problem for large groups in the password-based scenario. Douglas et al [22] proposed multi-factor password-authenticated key exchange.

In web technology, the access control [23-24] is one of the most essential security mechanisms the specification and management of access control policies is really a challenging problem. Somesh et al [25] developed formal verification techniques for access control policies and formalized classes of security analysis problems in the context of Role-Based Access Control (RBAC)pioneered by Ferraiolo et al.[26]. Subsequently the formal RBAC models were developed by Sandhu et al [27].The second level of security in the model is the authorization of users to the web services.

There are several architectural framework models launched for the secure web services in various realms like business domains, government sectors and so on. The following are some of such proposals.

Hwang et al [28] proposed an operational model for securing the Web services by fulfilling the essential security requirements such as authentication, confidentiality, data integrity with non-repudiation and by providing support for security mechanisms like encryption and digital signatures. This model still has the limitation of being not able to provide support for access control.

Wei She et al [29] presented an enhanced security model to provide better control of the information flow through service chains in composite web services by extending the basic security models with the introduction of delegation and pass-on policies to secure interactions in a composed web services. This model has the limitation of handling only the access control involved in the composite services and not intended for other security issues.

Joachim et al [30] proposed a framework for the decentralized execution of composite web services in order to ensure the correctness as well as the security of the execution. An attempt has been made to enforce the fundamental security requirements like confidentiality, authenticity, integrity, and availability but restricted to other security requirements such as access control or authorization of web services and non-repudiation.

Masahiro et al [31] proposed architecture for Service Supervision that monitors and controls the execution of a composite Web service in order to coordinate the Web services in an open environment. The framework proposed has been implemented on the existing standard languages such as WS-BPEL and WS-CDL which allows the use of existing tools and expertise for designing and operating composite Web services. However, the security issues for the access of composite web services have not been highlighted.

Stephanie et al [32] presented a generative environment for the orchestration of abstract services and the separate specification of non-functional properties. For the implementation of this proposal, a generic orchestration tool based on meta-models termed as Framework for Composition, Orchestration and Aggregation of Services (FOCAS) is used. A meta-model has been developed specifying the security concepts restricted to authentication, integrity and confidentiality but not extended to other issues like non-repudiation and authorization of web services.

In the web services scenario, there exist quite a lot of research activities, out of which some are essentially being carried out for the design and development of architectural framework models. Such proposals have been mostly intended for the secure enterprise web services but with some significant limitations. In this context, the design and development of Multi-Level Secure Framework (MLSF) for Composite Web Services [9] has become the pioneer model and an extended version of this proposal is presented [12].An enhanced model from the earlier versions is presented as a novel architectural framework in this paper. It is meant exclusively for the academic-related composite web services with multilevel security. The functionalities of the various components of this model have been illustrated along with the diagrams in the following section.

3. ARCHITECTURAL FRAMEWORK FOR THE PROPOSED SYSTEM

The great success of web application development is due to the excessive growth of Internet users in the global scenario which has prompted the web developers to develop a wide range of web applications. Meanwhile, a segment of Internet community has also been looking for reliable academic-related web services. As a result, the proposed model called *Multi-level Secure Framework* (MLSF) has been devised mainly for such web users envisioning the integration of academic institutions that are geographically dispersed. An architectural framework for the proposed system has the structure of multi-tier architecture as shown in Figure 1.

The entire architectural framework proposed for MLSF has been divided into four tiers like client tier, web tier, business tier and database tier that are mapped into four modules such as Client Interaction Module (CIM), Multilevel Security Module (MSM), Institutional Service Module (ISM) and Data Service Module (DSM) respectively. In this architectural framework design, the Multilevel Security Module (MSM) acts as an intermediary between the Client Interaction Module (CIM) where the academic web users have direct interaction with the system through browser and the Institutional Service Module (ISM) which is an integration of a very large group of academic institutions along with their respective databases represented as Data Service Module (DSM).

Two types of web users are involved in this model namely External User (EUsr) and Internal User (IUsr).The EUsr is a general web user who has only *view* privilege since he/she is not registered user having no direct association with any academic institution. Conversely, the Internal User (IUsr) is a registered user who can be a student, faculty, staff or an administrator of a particular academic institution and thus can gain the access privileges according to his/her role playing currently in that institution.

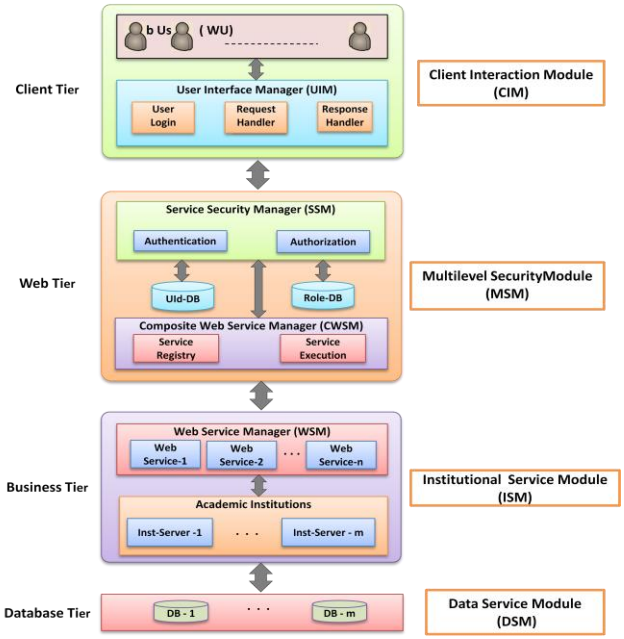

Fig.1 Architectural Framework of MLSF

In general, the web user initiates a request for academic information by interacting with the Client Interaction Module (CIM). The User Interface Manager (UIM) captures the request of the web user in the form of a URL and passes it to Multilevel Security Module (MSM). The EUsr being a general web user can access only to authorized information with the *view* privilege. After receiving the request, the MSM determines its type and resolves the path of the URL.

On the other hand, the IUsr logs into the User Interface Manager (UIM) by entering his/her UId and PWd. The MSM then consults databases such as *UId–DB* and *Role-DB* for user authentication as well as authorization respectively. The IUsr is authenticated by the verification of IUsr's UId stored in the User Identification database called *UId–DB*. If the IUsr's identification is valid, then the IUsr is authorized by the verification of IUsr's role defined in the User Role database called *Role-DB*. Activities of changing roles like adding a new role, modifying an existing role, and deleting a role that is no longer needed can be done by the administrator of this module. Once the IUsr is authorized, then the IUsr is granted permission to access only the authorized services according to his/her role.

Meanwhile, the incoming IUsr's request is examined for any malicious contents. If the incoming IUsr's request does not contain any malicious contents, then the request will be passed to the corresponding web service in the Institutional Service Module (ISM) by the MSM. Otherwise, an "*access denied"* message is sent to the CIM. If the web service request is a valid one, then the web service in the ISM will process the request and return the result to the MSM. Upon receiving the result from the ISM, the MSM then passes the result to the Composite Web Service Manager (CWSM). The CWSM composes several results thereby making one single result known as *Composite Web Service* which is then forwarded to UIM. After receiving the result from the MSM, the CIM processes the result and sends the appropriate messages to the web user through UIM.

4. SECURE ARCHITECTURE FOR WEB SERVICES

The architectural framework of MLSF has been proposed primarily to facilitate the web users to access the resources and services available in the academic institutions and to protect the sensitive data from the on-line attackers. The security architecture for the proposed model is shown in Figure 2.

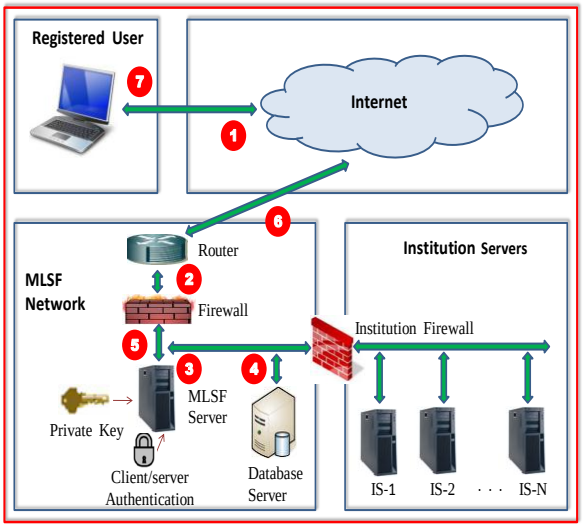

Fig.2 Secure Architecture for MLSF

1. The Web User (WU) is authenticated by user-id (UId) and password (PWd).
2. Encrypted Request to MLSF by secret key (PWd).
3. Decrypt the request with symmetric key.
4. Database validates the user with PWd.
5. MLSF responds to the WU.
6. MLSF sends list of Web Services.
7. The Web User (WU) obtains the Web Services.

5. PROTOCOL FOR PROPOSED SYSTEM

The MLSF protocol required for the different processes of the proposed model is presented in this section.

Let

EU	–	External User
IU	–	Internal User
URq	–	User Request
UId	–	User Identification
PWd	–	Password
WSC	–	Web Service calls

1. User → *URq*→ CIM
{ If (User = *EU*)
CIM → assign (*view* privilege) → *EU*
}
Else
{
CIM → assign (*view* privilege) → *EU*
}
Else If (User = *IU*)
{
IU → login(*UId* & *PWd*) → CIM
}

2. If (valid(*IU*) = *true*)
{
UIM → encrypt (*URq*) Using *PWd*→ CIM
}
If (user-login = *success*
{
CIM → process (*URq*)
CIM → initiate *WSC*
}
{
// *IU* re-enters *PWd* Go to step 1
}

3. MSM → intercept (*WSC, UId & PWd)*
If (*UId & PWd found* in *UId-DB)*
{
MSM → assign (*Role* from *Role-DB*) → *IU*}
Else
{
MSM → Send_msg (*"access denied"*) → CIM
}

4.If (*URq: permission valid &URq<>malicious_contents*)
{
MSM → dispatch (*URq*) → ISM
}
Else
{
MSM → Send_msg (*"access denied"*) → CIM
}

5. ISM → *ServiceResp*= process (*WSC*) → CWSM
CWSM →*ESResp*= Encrypt (*ServiceResp*) → UIM
UIM → *DSResp* = Decrypt *(ESResp)* → *IU*

6. RESULT AND DISCUSSION

Rigorous tests have been conducted with 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110 and 120 service clients generating the web service requests to the proposed system concurrently. Table 1 shows the number of web service requests chosen for the test.

Table.1 Number of Requests Chosen for the Test

Test	No. of Service clients	No. of Concurrent Requests
1	10	10
2	20	20
3	30	30
4	40	40
5	50	50
6	60	60
7	70	70
8	80	80
9	90	90
10	100	100
11	110	110
12	120	120
	Total	**780**

The statistical analysis is performed with the service clients and the test results are observed to determine the system response time. A summary of the test results is shown in Table 2.

Table.2 Summary of Test Results

Test	No. of Service clients	Standard Deviation (Sec.)	Throughput No. of Requests/Sec.
1	10	33.18	10.7
2	20	16.69	20.4
3	30	27.19	28.5
4	40	19.86	37.3
5	50	19.77	45.2
6	60	24.25	53.4
7	70	33.97	59.3
8	80	22.54	66.6
9	90	12.01	73.3
10	100	16.96	77.7
11	110	34.30	81.8
12	120	35.59	79.1

The maximum number of concurrent service clients to represent a test-case scenario is chosen as 120. During the performance tests, it is observed that the load on the web server was low with the less number of web servers. Meanwhile, it is also realized that the web server load was high with more number of service clients. Figure 3 shows the performance results based on the Table 2 to determine the throughput of the proposed system.

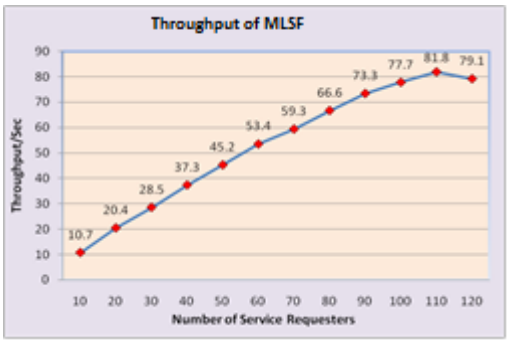

Fig.3 Performance Results for MLSF Throughput

Figure 3 depicts the standard deviation with respect to the corresponding web server load. It is also obvious that the MLSF throughput increases gradually up to 30 service clients and keeps on increasing till 110. It is observed that at 110 simultaneous service clients, the web server has reached the highest throughput of 81.8 due to the maximum load capacity of the web server and the throughput declines thereafter. It is studied from the graph that it does not produce a straight line. Therefore it is observed that the throughput of the proposed system is nonlinear.

7. CONCLUSION

An exceptional growth of Internet users and the technological development in the global software market have prompted the web developers to develop a wide range of web applications. Nevertheless, another section of Internet community is keen on reliable academic-related web services.

Having this vision in mind for such web users and envisioning the integration of academic institutions that are geographically dispersed, an architectural framework is presented in this paper for the previous model called Multi-level Secured Framework (MLSF) which is designed primarily for multiple secure academic web services such as admission service, fees payment service, course details service etc. for the students, teaching faculty and other employees.

This architecture provides the secured integrated composite web services for the academic web users through the proposed model that acts as an intermediary between the users and the various academic institutions located in different geographical area. This model has also been designed to enhance the security at multilevel such as user level, web services level, academic institution level and database level.

This model can provide a foundation for further research and development of an universal security paradigm for integrated composite web services pertaining to academic institutions situated all over the world by means of semantic web services.

8. REFERENCES

[1] Len Bass, Paul Clements and Rick Kazman, "*Software Architecture in Practice"*, Pearson Education, Second Edition, 2004.

[2] M. Shaw and D. Garlan, "*Software Architecture: Perspectives on an Emerging Discipline"*, Prentice-Hall of India Private Limited, New Delhi, 2007

[3] AthulaGinige and San Murugesan, "*Web Engineering: A Methodology for Developing Scalable, Maintainable Web Applications"*, Cutter IT Journal, Vol. 14, No. 7, Pages 24-35, July 2001

[4] S. Murugesan, Y. Deshpande, S. Hansen and A. Ginige, "*Web Engineering: A New Discipline for Development of Web-based Systems, Proceedings of the First ICSE Workshop on Web Engineering"*, International Conference on Software Engineering, Los Angeles, May 1999.

[5] A. Ginige and S. Murugesan, "*Web Engineering: An Introduction"*, IEEE Multimedia, vol. 8, no.1, pp. 14-18, Jan.–Mar. 2001.

[6] Y. Deshpande, S. Murugesan, A. Ginige, S. Hansen, , D. Schwabe, M. Gaedke, and B. White, "*Web Engineering"*, Journal of Web Engineering, vol. 1, no.1, pp 3-17, 2002.

[7] A. Ginige and S. Murugesan, "*The Essence of Web Engineering – Managing the Diversity and Complexity of Web Application Development"*, IEEE Multimedia, vol. 8, no.2, pp. 22-25, Apr.–Jun. 2001.

[8] A. Ginige, "*Web Engineering: Managing the Complexity of Web Systems Development*", Proceedings of SEKE 02,Ischia, Italy, ACM Press, July 2002.

[9] J. G. R. Sathiaseelan, S. Albert Rabara and J. Ronald Martin, "*Multi-Level Secure Framework (MLSF) for Composite Web Services*", In Proceedings of the ACM International Conference on Computer Sciences and Convergence Information Technology (ICCIT „09), vol. 2, pp. 580-585, Seoul, South Korea, 2009.

[10] Hitesh Ballani, Paul Francis and Xinyang Zhang, *A Study of Prefix Hijacking and Interception in the Internet*, SIGCOMM'07, ACM, Kyoto, Japan,27-31, August2007.

[11] Song Han, Wanquan Liu, Elizabeth Chang, *Deniable Authentication Protocol Resisting Man-in-the-Middle Attack*, World Academy of Science, Engineering and Technology 3 2005.

[12] J. G. R. Sathiaseelan, S. Albert Rabara and J. Ronald Martin, "*Multi-Level Secure Architecture for Distributed Integrated Web Services*", In Proceedings of 2010 3rd IEEE International Conference on Computer Science and Information Technology (ICCSIT '10), vol. 8, pp. 180-184, Chengdu, China, 9 -11 July2010.

[13] Daryl McCullough, "***A*** *Hookup Theorem for Multilevel Security*", IEEE Transactions on Software Engineering, Vol. 16, No. 6, June 1990.

[14] Timothy E. Levin, Cynthia E. Irvine, Clark Weissman and Thuy D. Nguyen "*Analysis of Three Multilevel Security Architectures*",CSAW'07,Fairfax, Virginia, USA, ACM, November 2007.

[15] OASIS, "*Web Services Security: SOAP Message Security*". WSS TC Working Draft, August 2003.

[16] BEA et al. "*Web Services Trust Language (WS-Trust)*", May, 2004.

[17] IBM, "*Web Services Federation Language, (WS-Federation)*", July 8, 2003.

[18] Yanjiang Yang, Robert H. Dengand FengBao, "*A Practical Password-Based Two-Server Authentication and Key Exchange System*", IEEE Transactions on Dependable and Secure Computing, Vol. 3, No. 2, April-June 2006.

[19] Steven M. Bellovin and Michael Merritt, "*Encrypted Key Exchange: Password-Based Protocols Secure against Dictionary Attacks*", Proc. IEEE Symposium on Research in Security and Privacy, pp. 72-84, 1992

[20] S. M. Bellovin and M. Merritt, "*Augmented Encrypted Key Exchange: A Password-Based Protocol Secure against Dictionary Attacks and Password File Compromise"*, Proceedings of the 1st Annual Conference on Computer and Communications Security, pp. 244-250, ACM1993.

[21] Shuhua Wu and Yuefei Zhu, "*Efficient Solution to Password-based Key Exchange for Large Groups"*, Journal of Networks, Vol. 4, No. 2, pp. 244-250, April 2009.

[22] Douglas Stebila, PoornaprajnaUdupi and Sheueling Chang, "*Multi-Factor Password-Authenticated Key Exchange"*, CRPIT, Vol. 105, No. 2, pp. 56-66, Information Security 2010.

[23] R. S. Sandhu, "Lattice-*Based Access Control Models*", IEEE Computer, Vol. 26, No. 11, pp. 9-19, 1993.

[24] Ravi S. Sandhu and Pierangela Samarati, "*Access Control: Principles and Practice*", IEEE Computer, Vol. 29, No. 2, pp. 40-48, 1996

[25] SomeshJha, Mahesh Tripunitara, Qihua Wang, and William H. Winsborough, "*Toward Formal Verification of Role-Based Access Control Policies*", IEEE Transactions on Dependable and Secure Computing, Vol. 5, No. 4, 2008.

[26] DavidF. Ferraiolo and D.Richard Kuhn, "*Role-Based Access Control*", Proc. 15th National Computer Security Conference, pp. 554 – 563, 1992.

[27] Ravi S. Sandhu, Edward J. Coyne, Hal L. Feinstein and Charles E. Youman, "*Role-Based Access Control Models*", IEEE Communications Magazine, Vol. 29, No. 2, pp. 38-47, 1994.

[28] G. H. Hwang, Y. H.Chang and T. K.Chang, "*An Operational Model and Language Support for Securing Web Services*", IEEE International Conference on Web Services (ICWS), 2007.

[29] Wei She, I-Ling Yen and Bhavani Thuraisingham, "*Enhancing Security Modeling for Web Services using Delegation and Pass-on*",IEEE International Conference on Web Services (ICWS), 2008.

[30] Joachim Biskup, Barbara Carminati, Elena Ferrari, Frank Muller and Sandra Wortmann, "*Towards Secure Execution Orders for Composite Web Services*" ,IEEE International Conference on Web Services (ICWS), 2007.

[31] Masahiro Tanaka, Toru Ishida, Yohei Murakami and Satoshi Morimoto, "*Service Supervision: Coordinating Web Services in Open Environment*", IEEE International Conference on Web Services (ICWS), 2009.

[32] Stephanie Chollet and Philippe Lalanda, "*An Extensible Abstract Service Orchestration Framework*", IEEE International Conference on Web Services (ICWS), 2009.