Research Methodology on Security Engineering for Web Services Security Architectures extended for Integration of Cloud, Big Data and IOT

Dr.D.Shravani

Rayalaseema University, Kurnool, A.P, India

Abstract *- This research paper deals with proposed research methodology on Security Engineering for Web Services Security Architectures extended for Integration of emerging technologies like cloud and Big Data and Internet of Things (IOT). Apart from proposing research strategy it also provides list of engineering tools for carrying out research.*

Keywords — *Security Engineering, Security Architectures, Web Services, Cloud Computing, Big Data, IOT*

I. Introduction

Web Service A Web Service is a method of communication between two electronic devices over a network. The World Wide Web Consortium (W3C) defines a "Web Service" as "a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically Web Services Description Language, known by the acronym WSDL). Other systems interact with the Web Service in a manner prescribed by its description using SOAP messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards." The W3C also states, "We can identify two major classes of Web services, REST-compliant Web services, in which the primary purpose of the service is to manipulate XML representations of Web resources using a uniform set of "stateless" operations; and arbitrary Web services, in which the service may expose an arbitrary set of operations" [Chris Douligeris].

Web Services Security Development and Architecture: Theoretical and Practical issues, involves Web Services Security Engineering, Web Services Security Architecture, Web Services Security Standards, Web Services Security Threats and Countermeasures [Carlos Gutirez]. Web Services Security Engineering implies, Security Engineering integrated into software development which is one of the major topics developed during the last few years [Kanchan Hans]. Applying security engineering throughout the different steps devised by the different software development methodologies has been a major topic in both scientific and industrial literature [Mouratidis]. Web Services Security Architecture should define the highest level organization of the IT security infrastructure necessary to meet the security requirements specified for the systems to be built by articulating the necessary security mechanisms in such a way that reusability, manageability and (internal/external) interoperability is guaranteed [Asoke K Talukder]. The Web Services Security Architecture, as per National Institute of Science and Technology (NIST) is a layered architecture consisting of Web Service Layer, Web Services Framework Layer and Web Server Layer [Anoop Singhal]. The goal of the Web Services Security Architecture is to summarize out the details of message level security from the mainstream business logic [Marzouk S Mokbel]. In the Web Services Secure application design, authentication and authorization are important research issues, pertaining to Security Architecture [Mail Jiang]. Even though Web Services are existing from the year 2004 onwards, Web 2.0 had made Web as a platform, with mashup applications from the year 2009 [Tim O Rielly]. This Web 2.0 Services Security needs to be investigated for research Moreover extension of these Web 2.0 Services applications in terms of Spatial Web Services Security needs to be investigated for research, in the area of Security Architecture Design [Reza B Far].

Designing Dependable Solutions Designing Secure Solutions implies that, the task of developing Information Technology solutions that consistently and effectively apply security principles has many challenges including: the complexity of integrating the specified security functions within the several underlying component architecture found in computing systems, the difficulty in developing a comprehensive set of baseline requirements for security, and a widely accepted security design methods[J J Whitmore]. Dependability implies privacy management of the application [Bhavani Thuraisingham]. Securing the Software application in any application at the design phase is known as Security Architecture, with a focus on authentication and authorization [Durai Pandain M]. Now a days, most of the applications are developed as a Layered Security Architecture pattern, typically having layers like User Presentation layer, Business Logic layer and Database access layer [Heiko Tillwick]. Today Agile Modeling (like Test Driven Development) is used in all Web applications design (our focus on Web Services), because of shortened development time, with customers collaborations with developers (pairs). Unfortunately Agile Modeled architecture is given less importance in literature because of quick development schedule, this research focuses on Secure Agile architecture for web services. Agile Modeling, being an iterative development approach, securing its architecture will provide Privacy information of the user, in the subsequent iterations [Hossein Keeramati].Our security approach is based on Model Driven Architecture (MDA) based Agile Security Modeling for Web Services [Hohn S Lowis].

Objectives of the Research Work:

This research “Designing Dependable Web Services Security Architecture Solutions” addresses the innovative idea of Web Services Security Engineering using Web Services Security Architecture with a research motivation of Secure Service Oriented Analysis and Design. It deals with Web Services Security Architecture for Web Services Secure application design, for Authentication and authorization, using Model Driven Architecture (MDA) based, Agile Modeled Layered Security Architecture design, which eventually results in enhanced dependable (privacy) management. All the above findings are validated with appropriate case studies of Web 2.0 Services, its extension to Web 2.0 Mashups Spatial Web Services and various financial applications.

Organization of Thesis:

The thesis is organized into seven chapters.

In Chapter 1, in this chapter, introduction to Web Services Security Architecture Design and Development, Objective of the thesis, Software Architecture security using Model Driven Architecture, Agile Methods are discussed.

In Chapter 2, in this chapter, a detailed literature survey was presented on Web Services Security Architecture, Model Driven Architecture, Agile Methodology, Security patterns for Agile Layered Security Architecture, UML 2.0, and Secure UML.

In Chapter 3, in this chapter we discussed design of Model Driven Architecture (MDA) based Agile Modeled Layered Security Architecture, (for Web Services), with initial case study validations using on simple secure Web Services Design using Agile Modeled Test Driven Development. Initially we discuss about Agile Security Architecture. Software Engineering covers the definition of processes, techniques and models suitable for its environment to guarantee quality of results. An important design artifact in any software development project is the Software Architecture. Software Architecture’s important part is the set of architectural design rules. A primary goal of the architecture is to capture the architecture design decisions. An important part of these design decisions consists of architectural design rules. In an MDA (Model-Driven Architecture) context, the design of the system architecture is captured in the models of the system. MDA is known to be layered approach for modeling the architectural design rules and uses design patterns to improve the quality of software system. And to include the security to the software system, security patterns are introduced that offer security at the architectural level. Moreover, agile software development methods are used to build secure systems. There are different methods defined in agile development as extreme programming (XP), Test Driven Development (TDD), Lean development, Scrum, Feature Driven Development (FDD) etc. Agile processing includes the phases like agile analysis, agile design and agile testing. These phases are defined in layers of MDA to provide security at the modeling level which ensures that “security at the system architecture stage will improve the privacy requirements for that system”. Later on we extend this approach for Web Services, with initial case study validations using on simple secure Web Services Design using Agile Modeled Test Driven Development [Nico Brehm].

In Chapter 4, in this chapter we presented, Designing Solutions using Agile Modeling for Web 2.0 Services Security Architecture and its implementations are discussed. Web 2.0 increases web based access to data processing particularly on the client side (AJAX Asynchronous Java Script and XML) that enables web applications which contains enriched functionality. Web 2.0 technologies have wide range of technologies and protocols which enables Web architecture to have greater access to data and functions. Traditional enterprises are skeptical in adopting Web 2.0 applications for internal and commercial use in public facing situations, with customers and partners. One of the prime concerns for this is lack of security over public networks. This chapter discusses and implements design of Web 2.0 services security architectures, for authentication over SSL/TLS.

In Chapter 5, in this Chapter, Dependability (Privacy Management) for Web Services Security Architecture is discussed with its implementations of a financial application for Secure Stock Market. Privacy is today an important concern for citizens, organizations and companies. We see an increasing number of organizations that collect data, very often concerning individuals, and use them for various purposes, ranging from scientific research, as in the medical data, to demographic trend analysis and marketing. Organizations may also give access to the data they own or even release such data to third parties, the number of increased data sets that are thus available poses serious threats to the privacy of individuals and organizations. To address such concerns, several privacy techniques have been developed. Despite such a large body of work, privacy issues specific to Web Services have not been yet investigated, or in other words, much work is not yet reported or little work has been done in this regard. . A very preliminary effort is represented by the identification of privacy requirements, as part of a larger set of Web Services Architecture Requirements, by a working model of World Wide Web Consortium. (W3C).

In Chapter 6, in this chapter, a case study on Web 2.0 mashup spatial Web Services Security Architecture is carried for validating the research results. Role Based Access Control for Spatial Web Services implies that: RBAC model is a widely deployed model in commercial systems and for which a standard had been developed. The widespread deployment of location-based services and mobile applications, as well as the increased concern for the management and sharing of geographical information in strategic applications like environmental protection and homeland security has resulted in a strong demand for spatially aware access control systems. These application domains impose interesting requirements on access control systems. In particular, the permissions assigned to users depend on their position in a reference space; users often belong to well-defined categories; objects to which permissions must be granted is located in that space; and access control policies must grant permissions based on locations and user positions.

In Chapter 7, in this chapter, the results of the research are summarized and suggested the work for further research.

II WEB SERVICES SECURITY TOOLS FOR DESIGN AND DEVELOPMENT

Oasis Service Standards

In this appendix, we will discuss some of the relevant standards by OASIS. Application Vulnerability Description Language (AVDL): “The goal of AVDL is to create a uniform format for describing application security vulnerabilities.”

Common Alerting Protocol: “The Common Alerting Protocol (CAP) provides an open, non-proprietary digital message format for all types of alerts and notifications.” Digital Signature Service (DSS): Two XML-based request/response protocols are developed. Te client and server communicate through these protocols. One is a signing protocol, and the other is a verifying protocol. As stated in the documentation, “through these protocols a client can send documents (or document hashes) to a server and receive back a signature on the documents; or send documents (or document hashes) and a signature to a server, and receive back an answer on whether the signature verifies the documents.”

The Directory Services Mark-up Language: (DSML): It “provides a means for representing directory structural information as an XML document.”

Electronic Business using eXtensible Markup Language (ebXML): It is a collection of XML-based standards that enable organizations to interoperate with each other and carry out e-business activities.

Extensible Access Control Markup Language (XACML): As stated in the documentation, “the XACML is a collection of core XML schema for representing authorization and entitlement policies.”

Reference Model for Service-Oriented Architectures (SO): “The goal of this reference model is to define the essence of service-oriented architecture, and emerge with a vocabulary and a common understanding of SOA.”

Security Assertion Mark-up Language (SAML): It is an XML-based framework for communicating user authentication, entitlement, and attribute information. Universal Description, Discovery and Integration (UDDI): It is a plat-form-independent, XML-based registry used by everyone to register them-selves on the Web.

Web Service Resource specification (WS-Resource): It is a specification that describes the relationship between a Web service and a resource in the WS-Resource Framework.

Web Services Resource Framework: It specifies a generic and open frame work for modeling and accessing stateful resources using Web services.

Web Services Security (WSS): As stated in the documentation, WSS specification proposes a standard set of SOAP extensions that can be used when building secure Web services to implement message content integrity and confidentiality.

Web Services Products

The products list can be divided into two groups: one is the Enterprise Service Bus (ESB)-related products, and the other is Web Services Suites.

1 Enterprise Service Bus-Related Products

In computing, an enterprise service bus (ESB) consists of a software architecture construct that provides fundamental services for complex architectures via event-driven and standards-based messaging engines (the bus). Developers typically implement an ESB using technologies found in a category of middleware infrastructure products, usually based on recognized standards. Some ESB products are:

BEA Systems (BEA Aqua Logic Service Bus) Acquired by Oracle

As stated by BEA, this is an intermediary for use as a core element of distributed services networks. It enables service-oriented architecture (SOA), allowing accelerated service reuse and deployment.

IBM Corporation (Web Sphere Enterprise Service Bus) As stated by IBM, Web Sphere is for SOA environments that enable dynamic, interconnected business processes, and deliver highly effective application infrastructures for business situations.

IONA Technologies (Artix ESB)—Acquired by Progress As stated by Progress Software, this product comprises technology-neutral SOA infrastructure products that work together or independently to provide flexibility in SOA adoption.

Oracle Corporation (Oracle Enterprise Service Bus) as stated by Oracle, this product is a fundamental component of Oracle's services-oriented architecture that provides a loosely coupled framework for inter-application messaging. Oracle also states that Oracle Enterprise Service Bus (ESB) is not Oracle Service Bus (OSB). ESB was developed by Oracle. OSB, formally known as Aqua logic Service Bus, was acquired when Oracle bought BEA.

Progress Software Corporation (Sonic ESB) As stated by Progress, this is a messaging-based enterprise service bus that simplifies the integration and flexible reuse of business applications within a service-oriented architecture (SOA).

WSO2 (WSO2 ESB) WSO2 is an open source SOA company. As stated by WSO2, this product offers an approach to creating an SOA by adding monitoring, management, and virtualization to existing service interactions.

Web Services Suites

The Web Services suites provide a framework for developing services and managing service-oriented architectures. A list of some of the products follows.

BEA Systems, Inc. (BEA Aqua Logic)—Acquired by Oracle

As stated by BEA, Aqua Logic consists of a software suite developed by BEA Systems for managing SOA. Following the acquisition of BEA by Oracle Corp., most of the software has been renamed and the term Aqua Logic is not used in any new Oracle product.

iWay Software (iWay Data Integration Solutions)

As stated by IWay Software, iWay Software data integration solutions allow for direct access to all the data, so an organization can design its architecture to address the unique information needs of its users.

Magic Software Enterprises (iBOLT Integration Suite)

As stated by Magic Software, iBOLT integrates enterprise software applications including SAP, Salesforce.com, Oracle JD Edwards, Lotus Notes, Microsoft Office, IBM i (AS/400), HL7, and Google Apps, among others.

Novell (Novell exteNd Composer)

As stated by Novell, the exteNd platform provides a visual environment that simplifies the development and deployment of business solutions that exploit existing systems.

Software AG (web Methods Product Suite)

As stated by Software AG, the web Methods product suite delivers business infra-structure that helps an organization to integrate its applications and automate its business processes.

III OUTLOOK ON MOVING OF COMPUTING SERVICES TOWARDS THE DATA SOURCES[16]

The Internet of things(IoT) is potentially interconnecting unprecedented amounts of eaw data, opening countless possibilities by two main logical layers: become data in to information, then turn information into knowledge. The former is about filtering the significance in the appropriate format, while the latter provides emerging categories of the whole domain. This path of the data is a bottom_up flow. In the other hand, the path of the process is a top-down flow, starting at the stategic level of business and scientific institutions. Today, the path of the process treasures a sizeableamount of well-known methods, architectures and technologies: the so called Big Data.On the top, Big Data analytics aims variable association (e-commerce),data mining(predictive behaviour)or clustering (marketing segmentation).Digging the Big Data architecture there are a myriad of enabling technologies for data taking, storage and management. However the strategic aim is to enhance knowledge with the appropriate information, which does not need of data, but not vice versa. In the way, the magnitude of upcoming data from the IoT will disrupt the data centres. To cope with the extreme scale is a matter of moving the computing services towards the data sources. This paper explores the possibilities of providing many of the IoT services towards the data centres (NaDa).Particularly, data information processes, which usually are performing at sub-problem domains. NaDa distributes computing power over the already present machines of the IP provides, like gateways or wireless routers to overcome latency, storage cost and alleviate transmissions. Large scale questionnaires have been taken for 300 IT professionals to validate the points of view for IoT adoption. Considering IoT is by definition connected to the Internet, NaDa may be used to implement the logical low layer architecture of the services. Obviously, such distributed NaDa send results on a logical high layer in charge of the information-knowledge turn. This layer requires the whole picture of the domain to enable those processes of Big Data analytics on the top.

Combining cloud computing infrastructures with ubiquitous IoT allow more objects to be connected to the internet,to process data, share results and allow individuals to be well-informed with the latest trends and businesses to stay competitive. The benefit of doing so can produce a large amount of data. Problems in IoT and cloud computing collaboration mean that a new version of IoT platforms should be designed and made available. Other approach includes proposing a new framework and architecture will be deployed in order to access higher quality of data management and request-and-response model. With this perspective and studying previous work on literature, we take a look from a new angel at current IoT architecture model and try to give a new model,to solve various problems. since we do an investigation on cloud computing data centres and gain nano dta centres, some king of distributed data centres, with better performance, we conclude, could replicate it over IoT, too. With the exextensive use of NaDas make changes in IoT architecture, and move most of its data centres form platform layer, the third layer to a lower layer, which is a network layer. Future work of this research includes investigation of a new architecture that uses specific IoT simulator and undertake experiments in real settings of IoT concept for a NaDa architecture

IV. Conclusion and Future Work

This research on Web Services Security Architecture is done using an innovative idea and novel implementations, of design of Model Driven Architecture (MDA) based Agile Modeling, for authentication and authorization of Web Service secure application design, for privacy management. We had validated our research with implementations on Web 2.0 Services Security Design, with its extension to Web 2.0 Mash up Spatial application, and various financial applications case studies.

To start with, a methodology on Model Driven Architecture based Agile Modeled Layered Security Architectures is given based on preliminary research motivation. In this preliminary research, the major part is given to model architectural design rules using MDA so that architects and developers are responsible to automatic enforcement on the detailed design and easy to understand and use by both of them. This MDA approach is implemented in use of Agile strategy in three different phases covering three different layers to provide security to the system. With this procedure a conclusion can be given that with the system security the requirements for that system are improved. To summarize the preliminary work on Model Driven Architecture based Agile Modeled Layered Security Architectures:

Software Engineering covers the definition of processes, techniques and models suitable for its environment to guarantee quality of results. An important design artifact in any software development research implementation is the Software Architecture. Software Architecture's important part is the set of architectural design rules. A primary goal of the architecture is to capture the architecture design decisions. An important part of these design decisions consists of architectural design rules. In an MDA (Model-Driven Architecture) context, the design of the system architecture is captured in the models of the system. MDA is known to be layered approach for modeling the architectural design rules and uses design patterns to improve the quality of software system. And to include the security to the software system, security patterns are introduced that offer security at the architectural level. Moreover, agile software development methods are used to build secure systems. There are different methods defined in agile development as extreme programming (XP), scrum, feature driven development (FDD), test driven development (TDD), etc. Agile processing includes the phases like agile analysis, agile design and agile testing. These phases are defined in layers of MDA to provide security at the modeling level which ensures that “security at the system architecture stage will improve the requirements for that system”. Dependable Solutions for Security Requirements involves Privacy Management.

In chapter 1 we presented the basics of Designing Dependable Web Services Security Architecture Solutions, with some main concepts like security, MDA, agile methodology and software architecture, Web Services Security Design etc. Further detailed introduction to the thesis entitled Designing Dependable Web Services Security Architecture Solutions is given, with some main concepts like security, MDA, agile methodology and software architecture, Web Services Security Design etc.

In Chapter 2 we discussed about Review of Literature on Designing Dependable Web Services Security Architecture Solutions, with general Research Issues on Security Architectures using Model Driven Architecture based Agile Modeling for Security Architectures and Specific Web Services Security Architecture Development issues using Agile Modeling. Further detailed Review of Literature on Designing Dependable Web Services Security Architecture Solutions, with general Research Issues on Security Architectures using Model Driven Architecture based Agile Modeling for Security Architectures and Specific Web Services Security Architecture Development issues using Model Driven Architecture based Agile Modeling.

In chapter 3 we developed about Agile Modeling for Security Architectures, and we also developed Agile Modeling for Secure Web Services Architecture design, with simple case study and implementations. Finally we developed about Basic Secure Web Services Design using Agile Modeling. A methodology of Security design of Model Driven Architecture based Agile Modeled Layered Security Architecture is given, for Web Services Security Design with appropriate Web Services Case Studies design using Class Diagrams and Sequence Diagrams design.

In Chapter 4, methodology for Web 2.0 Services Security Design is provided, with implementations and validations of AJAX (Asynchronous JavaScript and XML) design, Web 2.0 Security design at Secure Socket Layer (SSL) and Web 2.0 Services authentication, is given, all of those design is based on Model Driven Architecture based Agile Modeled Layered Security Architecture design based on previous Chapter 2. Further, we developed Web 2.0 Services Security Design, using appropriate Agile Modeled Design strategies of Case Studies of AJAX Security, Web 2.0 Services design at SSL and Web 2.0 Services Authentication mechanism.

In Chapter 5, a methodology for Dependable (Privacy Management) of Web Services Security Design is provided, with various cases design like basic Web Services Privacy, Web 2.0 Services Privacy, Financial application and Secure Stock Market design. This Security Design and implementation is done using concepts of earlier chapters like Agile Modeling and Web 2.0 Security Design. Further we developed privacy management for web services security design for dependability with various case studies like web 2.0 services using agile modeling.

In Chapter 6, methodology on a case study of Web 2.0 Mashup Spatial Web Services Applications, case study is carried out using Agilr Modeled Web 2.0 Services Security for Privacy Management. Further, we discussed about security design for web 2.0 Mashup spatial application privacy using agile modeling.

FUTURE WORK:

The future scope of this work includes adapting agile lean project development strategies for web services security architecture design.

This preliminary research on MDA based Agile Layered Security Architecture summarizes that security is essential for every system at initial stage and upon introduction of security at middle stage must lead to the change in the system i.e., an improvement to system requirements.

Secure Services: Web services and service-oriented architectures are at the heart of the next-generation Web. We expect them to make use of semantic Web technologies to generate machine-understandable Web pages. This is one of the major developments in the late 1990s and early 2000s. While there are numerous developments in Web services, the application of semantic Web technologies and securing the Web services are major challenges. Furthermore, major initiatives such as the global information grid and the network centric enterprise services are based on Web services and service-oriented architectures. Therefore, securing these technologies as well as making Web services more intelligent by using the semantic Web will be critical for the next-generation Web.

Next, SOAD as well as secure SOAD are in their infancy. While initially there are various approaches for secure SOAD, we believe that eventually these approaches will be unified to develop a unified approach. In the same way, one can also expect secure SOAD approaches to be unified. However, first we need some approaches for securely modelling services, and research is just beginning in this area.

With respect to access control, a lot of work remains to be done. We need an appropriate security model for services. ABAC is one such model. We need to examine how ABAC can be integrated with UCON. We also need to examine the inference problem in more detail for services. Finally, we need to develop standards similar to SAML and XACML to include more sophisticated forms of fine-grained access control.

Finally, as Web services explode and we carry out more and more transactions on the Web, as well as get involved in social networks, it is critical that we protect the identity of individuals and ensure authorized access. Furthermore, a user may be involved in multiple social networks and multiple transactions. The user may have different identities in different systems. Therefore, we need an effective mechanism to manage the numerous identities of possibly billions of users. Research on identity management is just beginning. We need a lot more work in this area and also develop appropriate standards.

Dependable Systems: We have assumed that dependability includes trust, privacy, and integrity. We have also included multilevel security as part of dependability. We need trust management and negotiation techniques that take advantage of semantic Web technologies. We need to examine standards such as P3P and develop appropriate technologies to enforce the various privacy policies. When agents carry out activities on the Web, we need to ensure that the data is of high quality. We need a better annotation system to manage data provenance. With respect to multilevel security, we need to ensure that agents at multiple levels communicate with each other securely.

Secure Semantic Web: We need to develop tools and techniques to ensure the security of operation of the semantic Web. We need languages to express policies as well as agents to securely carryout various activities on the Web. In addition, we require techniques for securing semantic Web technologies such as XML, RDF, and OWL documents. We also should have a better handle on the inference problem using semantic Web technologies.

Specialized Secure Services: We have to develop specialized secure services for various types of data such as geospatial and sensor streams. We also must develop services to manage multimedia data. In addition, we have to have better domain-specific secure services for applications such as healthcare, finance, and defense, among others.

REFERENCES

[1] Alastair Airchison [2009], "Beginning Spatial with SQL Server 2008", Apress Publisher, ISBN 978-1-4302-1829-6, Chapter 1, pp. 1 – 37.

[2] Alessandra Bagnato (Eds.), SEC MDA [2009], "Security in Model-Driven Architecture", European workshop on Security in MDA 2009, Netherlands, ISSN No. 0929 – 0672. pp. 01 – 56.

[3] Anders Mattsson, Bjorm Lundell, Brian Lings, Brian Fitzgerald [2009], January/February 2009, "Linking Model-Driven Development and Software Architecture: A Case Study", IEEE Transactions on Software Engineering, vol. 35, no. 1. pp. 83-93.

[4] Anoop Singhal, Theodore Winograd [2006], September 2006, "Guide to Secure Web Services", National Institute of Standards and Technology (NIST) Draft, (800-95).

[5] Annekayem [2009], "Security in Service-Oriented Architectures: Standards and Challenges", IGI Global,

Information Science Reference, DOI:10.4018/978-1-60566-950-2.ch009, pp. 187 –211. .

[6] Antonio Mano, Gimena Pujol, Antonio Munaz [2009], “Policy based Security Engineering of Service Oriented Systems”, IGI Global, Information Science Reference, DOI:10.4018/978-1-60566-950-2.ch006, pp. 118 – 133.

[7] Asoke K. Talukder and Manish Chaitanya [2009], “Architecting Secure Software System” CRC Press, chapter 2, pp. 45 – 90.

[8] A Mohammad, G.Kannan, R.Kannan, T Khdour, S.Bani-ahmad, A.Alarabeyyat, [2011], “Toward Access Control Model for Web Services applications” in International Journal of Research and Reviews in Computer Science (IJRRCS) Vol 2 No 2 pp. 253- 264.

[9] Barbara Russo, Maro Scotto, Alberto Silliti [2010], “Agile Technologies in Open Source Development” IGI Global publishers 2010, pp. 217 – 244.

[10] Basin D, Burri S J, Karjoth G [2011] ,“Separation of duties as a service”, Proceedings of the Sixth ACM Symposium on Information, Computer and Communications Security, ACM, China, pp. 1 – 7 .

[11] Bernard Menezes [2010], “Network Security and Cryptography”, Cengage Learning India Pvt. Ltd., ISBN 978-81-315-1349-1, pp. 245 – 290.

[12] Bhavani Thuraisingham [2011], “Secure Semantic Service Oriented Systems”, Auerbach Publications, Chapter 1, pp. 1 – 17.

[13] Bruce Powel Douglass [2009], “Real-Time Agility, the Harmony/ESW Method for Real-Time and Embedded Systems Development”, Copyright at 2009 Pearson Education, Inc., pp. 1-31.

[14] Carlos Gutierrez, Eduardo Fernandez-Medina, Mario Piattini [2009], “Web Services Security Development and Architecture: Theoretical and Practical issues”, IGI Global, Information Science Reference. ISBN 978-1-60566-950-2, pp. 1 – 14.

[15] Cenzic Inc., [2009], “Web Application Security Trend Reports”, A White Paper, pp. 1 – 4.

[16] Farzaneh Akhbar, Victor Chang, Yulin Yao, Victor Mendez Munoz,”Outlook on moving computing services towards data sources”,

[17] Karandeep Kaur,”A reviw of Cloud Computing Service Models”, International Journ al of Computer Applications, Vol 140, No.7 April 2016, pp.15-18

[18] Angel Lagares Lemos, Floarian Daniel, Boualem Benatallah,” Web Service Composition: A survey of techniques and tools”, ACM Computing Surveys, Vol 48, No 3, Article 33, December 2015, pp.33.1 – 33.41