Analysis of Security Service Oriented Architecture (SOA) With Access Control Models Dynamic Level

Erick Fernando[1], Pandapotan Siagian[2]
STIKOM Dinamika Bangsa,
Jl.Jend Sudirman The Hok , Jambi, Indonesia
[1]Erick.fernando_88@yahoo.com
[2]Siagian.p@gmail.com

***Abstract*— Now we are moving towards the "Internet of Things" (IOT) in millions of devices will be interconnected with each other, giving and taking information provided within a network that can work together. Because of computing and information processing itself IOT core supporters, So in this paper introduces "Service-Oriented Computing" (SOA) as one of the models that can be used. Where's it at each device can offer functionality as a standard service [4]. In SOA, we can make the resources available to each other in the IOT together. However, a major challenge in these service-oriented environment is the design of effective access control schemes. In SOA, the service will be invoked by a large number, and at the same time authentication and authorization need to cross several security domains are always used. In this paper, we present the analysis of data safety suatua *Workflow-Based Access Control Model associated oriented* (WABAC) to troubleshoot problems that occur within a system integration. The analysis showed that the point system function model based integration system that is lower than the legacy model of SOA-based systems, by designing several services using WOA approach. In addition, we have observed that the integrated model can guarantee the quality of service, security and reliability main, by applying SOA approach when needed. Finally, experimental results have proved that the service can be run side by side seamlessly without performance degradation and additional complexity.**

***Keywords*— Service Oriented Architecture (SOA), Integration, Operational Data, Web Services, Security, Access control Models Dynamic Level**

I. INTRODUCTION

In this paper, Describing a security that takes into account the needs of access control in a distributed environment such as service-oriented architecture-based services are handled. In a software development, as a whole, is a complex process that occurs in a safety, and the constantly changing requirements in the development stage. Configuration management software happens to be the most important part because it requires modifying large enough in doing software design and code. Here are a few examples of the architecture of access control models based services are analyzed with Workflow models - oriented Attributed Based Access Control (WABAC). Software development process provides a solution to a changing environment. WABAC models using an incremental approach to developing high-quality software within time, cost and other related constraints through several iterations.

In the process of this WABAC models raises some important factors in software project management, for example, scope, cost, time and quality. Software engineering explore constructive and dynamic way to manage the entire project life cycle.

According to analysis carried out with regard to WABAC models have a dynamic and flexible structure which is higher than the other models, so it can be concluded that this model is more appropriate for a dynamic environment such as service-oriented architecture environment and integrated systems on a system that occurred a considerable transaction.

II. SERVICE ORIENTED ARCHITECTURE (SOA)

Service Oriented Architecture (SOA) is a collection of services that communicate with each other to fulfill a particular business process. This paradigm passes data between service consumer and service provider either simply or complicatedly. SOA is a popular strategy to provide an integrated, flexible, and cost efficient (Web) Service-based enterprise. It promises interoperability, reusability, loose coupling, and protocol independency of services as core principles of SOA. Normally, this standard-based approach uses Web Services as building block to support particular business tasks. Web Services are published with Web Services Description Language (WSDL) interface and they use Simple Object Access Protocol (SOAP) as a communication protocol. Figure 1 shows the operation that each component can perform.

III. WEB SERVICES

According to, Web Services are loosely coupled computing services that can reduce the complexity of building business applications, save costs, and enable new business models. Web Services are application components that using open protocols to communicate and they are self-contained and self describing. Web Service can be discovered using UDDI and used by other applications. Extensible Markup Language (XML) is the basic for Web Services. Web Services can be able to publish the functions and data to the rest of the world. A Web Service is a software interface that describes a collection of operations that can be accessed over the network through standardized XML messaging. It uses protocols based

on the XML language to describe an operation to execute or data to exchange with another Web Service.

IV. SOA AND WEB SERVICES

Although much has been written about SOA and Web services, there still is some confusion be-tween these two terms among software developers. SOA is an architectural style, whereas Web services is a technology that can be used to implement SOAs. The Web services technology con-sists of several published standards, the most important ones being SOAP and WSDL. Other technologies may also be considered technologies for implementing SOA, such as CORBA. Al-though no current technologies entirely fulfill the vision and goals of SOA as defined by most authors, they are still referred to as SOA technologies. The relationship between SOA and SOA technologies is represented in Figure 1. Much of the technical information in this report is related to the Web services technology, because it is commonly used in today's SOA implementations.

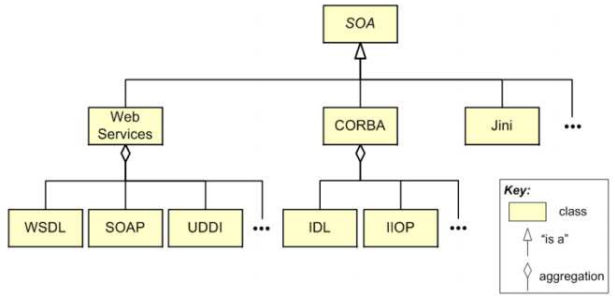

Fig. 1 SOA and SOA Technologies

V. WSARCH (WEB SERVICES ARCHITECTURE)

The WSARCH (Web Services Architecture) [7] is an architecture which allows accessing Web services using a combination of functional and non-functional aspects of Quality of Service (QoS). These QoS aspects aim at evaluating the performance of Web services in order to achieve QoS in a service-oriented architecture. These QoS attributes were mapped to the components participating in a service-oriented architecture that incorporates quality of service. The architecture provides the monitoring of service providers and the data obtained are used to locate the most appropriated service. A prototype for the WSARCH allows performance evaluation studies being conducted considering different components of the architecture, algorithms, protocols and standards.

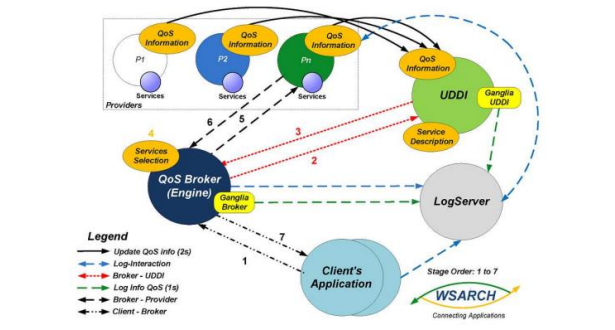

Fig. 2 WSARCH

By now, we want include security attributes in this architecture involving all. the components (UDDI, Broker, clients and providers). The WSARCH and its components are presented in Figure 2.

V. ACCESS CONTROL MODELS

So far various models have been proposed to solve accesses control problem that each one has its own advantages and disadvantages. In this section, some examples of such models are dealt with.

A. *Identity-Based Access Control*

Under this Model, permissions to access a resource is directly associated with a subject's identifier (e.g., a user name). Access to the resource is only granted when such an association exists. An example of IBAC is the use of Access Control Lists (ACL), commonly found in operation systems and network security services [7].The concept of an ACL is very simple: each resource on a system to which access should be controlled, referred to as an object, has its own associated list of mappings between the set of entities requesting access to the resource and the set of actions that each entity can take on the resource.

B. *Role-Based Access Control*

The RBAC model restricts access to a resource based on the business function or the role the subject is playing. The permissions to access a resource are then assigned to the appropriate role(s) rather than being directly assigned to subject identifiers [8]. When a user changes jobs, another user is allowed to take on that role. No ACL changes are needed. Of course, sometimes only a few of the user's rights change. In that case, a new role needs to be introduced. Often the rights associated with a role depend on which user is acting in that role. In that case, too, a new role needs to be introduced[9]. The RBAC reference model is defined in terms of four model components: Core RBAC, Hierarchical RBAC, Static Separation of Duty Relations, and Dynamic Separation of Duty Relations [10]. Although RBAC may take slightly different forms, a common representation as defined in [11] that is depicted in Fig. 3.

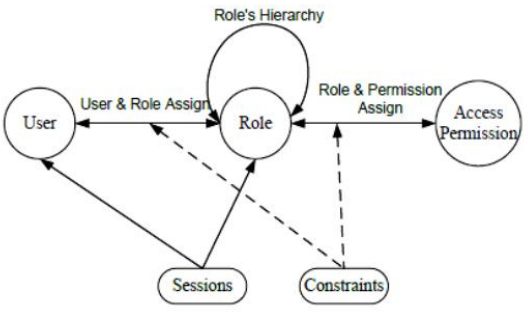

Fig. 3 Role-based access control model

C. *Attribute-Based Access Control*

Policy Based Access Control (PBAC), which is called Attribute-Based Access Control (ABAC) in the US Defense Department jargon, extends RBAC to a more general set of properties [1]. Unlike IBAC and RBAC, the ABAC model [9]

can define permissions based on just about any security relevant characteristics, known as attributes. For access control purposes, we are concerned with three types of attributes:

1. Subject Attributes (S). Associated with a subject that defines the identity and characteristics of that subject.
2. Resource Attributes (R). Associated with a resource, such as a web service, system function and or data.
3. Environment Attributes (E). Describes the operational, technical, or situational environment or context in which the information access occurs.

ABAC clearly provides an advantage over traditional RBAC when extended into SOA environments, which can be extremely dynamic in nature. ABAC policy rules can be custom-defined with respect to semantic context and are significantly more flexible than RBAC for fine-grained alterations or adjustments to a subject's access profile. ABAC also is integrated seamlessly with XACML, which relies on policy-defined attributes to make access control decisions. One additional benefit behind web service implementations of ABAC lies in the nature of the loose definition of subjects. Because ABAC provides the flexibility to associate policy rules with any actor, it can be extended to web service software agents as well [10].

One additional advantage of ABAC web service implementations is related to the nature of the loose definition of the subjects. Because ABAC provides the flexibility to associate policy rules with any actor, it can be extended to web service software agents as well. Figure 4 illustrates how an ABAC attribute authority (AA) can be integrated into a SAML framework. In this diagram, the AA generates attribute assertions containing all attributes necessary for an ABAC policy-based access control decision written in XACML. The PDP uses the attribute assertions, the authentication assertion, and the XACML policy to generate an authorization decision assertion [2].

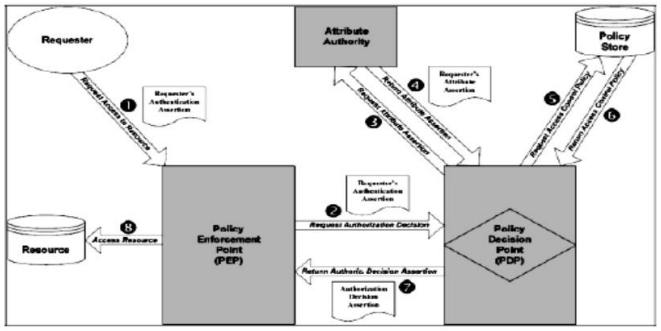

Fig. 4 Use of SAML and XACML in implementing ABAC

D. *Risk Adaptive Access Control*

Risk Adaptive Access Control (RAdAC) [13] is another variation access control method. Unlike IBAC, RBAC and ABAC, however, Radii makes access control decisions on the basis of a relative risk profile of the subject and not necessarily strictly on the basis of a predefined policy rule. Fig.3 illustrates the logical process governing RAdAC, which uses a combination of a measured level of risk the subject poses and an assessment of operational need as the primary attributes by which the subject's access rights are determined.

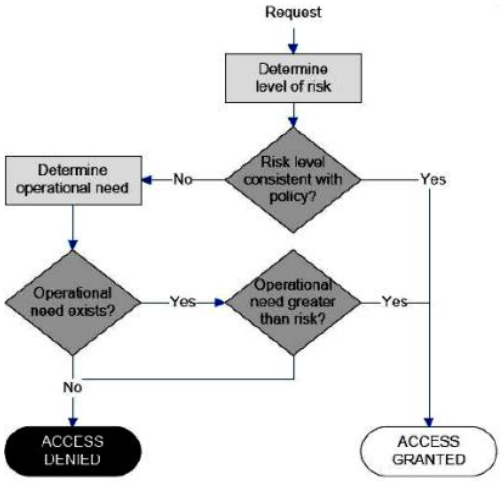

Fig. 5 RAdAC Decision Tree

E. *WABAC Access Control Framework*

The model of WABAC can realize fine-grained access control of cross-domain system; also it can manage subject's permissions dynamically. This model is suitable for access control of SOA, especially workflow based distributed computing system [6]. Fig.3 depicts the access control view of WABAC. The following will discuss the implementation of WABAC model and present an access control framework.

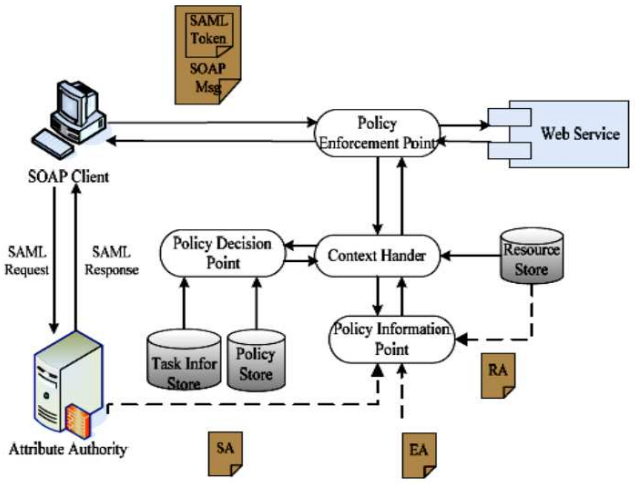

Fig. 6 WABAC Access Control Framework

With Web services implemented and the inclusion of their security policies, experiments and data collection were performed for this analysis. Thus, the performance of a Web service without security with other Web services using the WS-Security to add encryption and digital signatures in SOAP messages exchanged in communication have been compared. Furthermore, the results obtained with the WS-Security were compared with results obtained in an experiment where the Web service using the SSL security standard. As could be seen, despite having a relatively lower response time, SSL does not guarantee end-to-end security. Due to the inherent characteristics of the protocols that make up a service-oriented

architecture, security becomes a key item. Thus, studies and performance evaluation of the inclusion of security in this environment are important, since such inclusion causes a considerable reduction in the performance of a service-oriented architecture. The study presented in this paper demonstrates that in addition to encryption factor, the number of concurrent clients requesting a particular service confirms the performance degradation.

VI. CONCLUSION

In this paper, Describing a security that takes into account the needs of access control in a distributed environment such as service-oriented architecture-based services are handled. In a software development, as a whole, is a complex process that occurs in a safety, and the constantly changing requirements in the development stage. Configuration management software happens to be the most important part because it requires modifying large enough in doing software design and code. Here are a few examples of the architecture of access control models based services are analyzed with Workflow models-oriented Attributed Based Access Control (WABAC). Software development process provides a solution to a changing environment. WABAC models using an incremental approach to developing high-quality software within time, cost and other related constraints through several iterations. In the process of this WABAC models raises some important factors in software project management, for example, scope, cost, time and quality. Software engineering explore constructive and dynamic way to manage the entire project life cycle.

According to analysis carried out with regard to WABAC models have a dynamic and flexible structure which is higher than the other models, so it can be concluded that this model is more appropriate for a dynamic environment such as service-oriented architecture environment and integrated systems on a system that occurred a considerable transaction.

REFERENCES

[1] A.H.Karp and J. Li, "Solving the Transitive Access Problem for Service-Oriented Architecture", IEEE International Conference on Availability, Reliability and Security, DOI 10.1109/ARES.2010.

[2] Singhal, T. Winograd and K. Scarfone, "Guide to Secure Web Services", National Institute of Standards and Technology Special Publication. .2007.

[3] D.F. Ferraiolo and D.R. Kuhn. "Role Based Access Control", 15th National Computer Security Conf.: 554-563. 1992.

[4] D.Smith,"Migration of legacy assets to service-oriented architecture environments," in Proceedings of the 29th International Conference on Software Engineering, 2007, pp. 174-175.

[5] E.Yuan and J. Tong. "Attributed Based Access Control (ABAC) for Web Services", IEEE International Conference on Web Services (ICWS'05). 2005.

[6] Zhang and J. Liu, "A Model of Workflow-Oriented Attributed Based Access Control" , I. J. Computer Network & Information Security,1, 47-53.2011.

[7] Thies and G. Vossen, "Web-oriented architectures: On the impact of web 2.0 on service-oriented architectures," in Proceedings of IEEE Asia-Pacific Services Computing Conference, 2008, pp. 1075-1082.

[8] Jorstad, S. Dustdar, and D. Thanh, "A service oriented architecture framework for collaborative services," in Proceedings of the 14th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprise, 2005, pp. 121- 125.

[9] C. Estrella, R. T. Toyohara, B. T. Kuehne, T. C. Tavares, R. C. Santana, M. J. Santana, and S. M. Bruschi. "A Performance Evaluation for a QoS-Aware Service Oriented Architecture". IEEE Congress on Services, pp. 260-267. 6th World Congress on Services, 2010.

[10] J.Tong, "Attribute Based Access Control: New Access Control Approach for Service-Oriented Architectures", Workshop on New Challenges for Access Control, Ottawa, Canada, Apr.2005.

[11] M. Beadley, "Function point counting practices manual, release 4.1," International Function Point Users Group (IFPUG), 1999.

[12] Mohammad Mahdi Shafiei , Homayun Motameni and Javad Vahidi. "Analyzing Access control Models Dynamic Level and Security In Service–Oriented Architecture Environment" International Journal of Mechatronics, Electrical and Computer Technology Vol. 4(11), pp. 470-484, ISSN: 2305-0543, Apr. 2014

[13] P.C. Cheng, P.Rohatgi, and C. Keser, "Fuzzy MLS: Experiment on Quantified Risk-Adaptive Access Control", IEEE Symposium on Security and Privacy, PP. 222-230.2007.

[14] Phil Bianco, Rick Kotermanski and Paulo Merson. "Evaluating a Service-Oriented Architecture", Software Architecture Technology Initiative, Carnegie Mellon University, September 2007

[15] R. S.Sandhu et al, "Role-Based Access Control Models. IEEE Computer", pp. 38-47. 1996.

[16] R Kuhn, American National Standards Institute. 2003.

[17] S. Balasubramaniam, G. Lewis, E. Morris, S. Simanta, and D. Smith, "Challenges for assuring quality of service in a service-oriented environment," in Proceedings of ICSE Workshop on Principles of Engineering Service Oriented Systems, 2009, pp. 103-106.

[18] T. Uemura, S. Kusumoto, and K. Inoue, "Function point analysis for design specifications based on the unified modeling language," Journal of Software Maintenance and Evaluation, Vol. 13, 2001, pp. 223-243.