Intelligent Security Model and Implementation on SOA Firewall

Nashat Al-Jallad
College of Graduate Studies
Palestine polytechnic University
Hebron, Palestine
jallad@student.ppu.edu

Faisal T. Khamayseh
College of Graduate Studies
Palestine polytechnic University
Hebron, Palestine
faisal@ppu.edu

Abstract— **During last decade, development of the internet has increased rapidly. There is an increasing amount of desired needs to support data sharing, distribution and data publishing on the web. XML has become a new standard for data representation and information exchange on the Internet.**

The primary objective of this study is the development of a framework using Service Oriented Architecture (SOA) in order to improve the security of XML. In order to achieve confidentiality, integrity and authenticity, researchers proposed an intelligent firewall using data mining mechanisms and perform SOA security using a mechanism of firewall coupled with data mining methods in order of update SOAP filter (firewall) parameters.

Keywords-SOA; SOAP filter, Security Model; Security Service; Firewall; styling; insert (key words)

I. Introduction

During last decade, Internet application became an important object of software industry. Service-Oriented Architecture (SOA) technology has revolutionized the distribution of applications by introducing web services. Web service characterized by set of attributes (e.g., Interoperability, Maintenance, Reuses, and Smart pipe), in addition to Web service based on XML and HTTPs.

Simple Object Access Protocol (SOAP) message is used as main protocol to communicate between the security services embedded inside the SOA security framework and between services provider and service requester. The service is described by Web Services Description Language (WSDL), using UDDI to find access service through service metadata. The development of SOA technology and related industry standards have created new opportunities for accountability and control, and added new dimensions to security concerns.

There are set of attacks we mainly consider through this paper such as the Denial of Service attack (DoS), which occur as a result of mutual vulnerability to file size. Other attack call, XML-injection, occurs by spoiling XML stream and inputting data that will overwrite the static portions of the stream.

Our approach is to build an intelligent XML-firewall in order to provide authenticity and to prevent these services from potentially attacked. This includes two portions: first part is a firewall with set of rules that guarantee protection to later web services and the second part is the intelligent unit that uses data mining algorithms, such as Association Rules mining algorithm in order to make an automatic updates for the set of parameters used in a proposed firewall.

II. Related work

Recently, many researchers studied certain attacks in order to get security solutions. Most of these solution focus on guaranteeing that the service stops a single or a group of attacks. SOA relies on three main components: Service Provider, Service Broker, Service Registry, and Service Requester (service client/consumer).

Bertino et al. [1] discussed three classes of security service (e.g., authentication, examines access control, and identity management) in terms of the service level. They proposed an architectural reference framework that found on the event-based approach. Event-based approach regarding distributed environments is special situation (i.e, environments without central control). So the authentication performance is then the area of interest. They proposed a mechanism to record the authentication events for each of subject and determine the data that the subject has used in authentication events. These data stored in a log files that can be reuse to authenticate the subject in to SOA system. But they did not discuss that their framework will do well with some problems such as in the case intrusion detection and trust management.

Other approach focused on authorization as a main issue. For example, the use of XACML, which is a general standard language for describing architecture of an authentication policy base. Priebe et al.[2] proposed an extension to the XACML architecture by introducing the way to apply anthology-based inference engine for collecting additional attributes that may required by web service, and then applying it to generic authentication and authorization infrastructures(AAIs). Nevertheless, El Yamany[3] pointed out this approach to still undergoing extensive research. Furthermore, he pointed out that the traditional security technique (e.g., virtual private net works and Security Dockets Layer (SSL)) can not protect the large number of transactions that web services/SOA execute in short period of time.

Despite SAML as being a standard way to represent authentication attribute and authorization information, El Yamany [3] pointed out that SAML is designed for solving web browser Single Sign-On (SSO) problem and not suitable for all authorization situations.

Many significant companies maintain security issue for SOA, such as IBM, Oracle and Microsoft. For example, IBM introduced a model consists of three fundamental levels: Business security service, Security policy infrastructure, and IT security service. In addition, Oracle developed anther three level tools consists of Policy manager, and Operational management and monitoring. But this Attempt is still in a need to be validated. Also the framework does not discuss the change of its services, and still limited to the fixed type of relation[14].

RBAC is out of date with the advent of new Web Technologies Liu et al. [4], but it is an authentication model intended to solve SOA authorization administration problem. Zhang et al.[5] proposed extended RBAC model for XML security that can be administrated in distributed environments. In addition, Liu et al. [4] proposed an Attribute Role-based Access Control (ARBAC) as a hybrid of RBAC and ARBAC. In order to enhance the combination of RBAC and ARBAC, Emig et al. [6] introduced a hybrid access control for the development of an authorization verification service. Eric [8] defined a framework for Securing SOA, which is related to web service security and to illustrate web service security mode. Fernandez [7], classified XML security into two classes; first, the Transmission security: which focuses on looking for standers; like Assertion Markup Language (SAML), XML Key Management Specification (XKMS), XML Trust Assertion Services Specification (XTASS), Security Services Markup Language(S2ML), and Extensible Access Control Markup Language (XCML), Sun[8]. Second, the document security, which includes a mechanism of securing the XML document itself by either XML Encryption using different keys, or by XML Signature (digital signature).

An XML firewall was suggested by Loh et al. [9], in order to protect the web services by filtering the incoming SOAP messages. They have proposed three different filters (e.g., Message size filtering, Syntax parsing and XML schema validation). This approach succeeded in blocking some attacks (e.g., oversized payloads, recursive payload and SQL injections).

Rongbo [10] discussed three type of firewalls (Packet-Filtering, Application-Level and Circuit-Level firewall) . He presented the weakness of those filters to prevent attack form inside. He extended the Firewall Package Toolkit FWTK, by analysing the contents of network traffic passing through the firewall.

Data mining has success achievements is many fields, especially those processing massive amount of data, to detect any potential attack. Data mining used in the protection of military and investigation systems in order to predict scenarios that could cause damage to systems and threaten countries, such as terrorism El Yamany [3]. With these successive achievements, data mining used with SOA security to solve this issue of private violation. Zhang et al. [12] produced a method to quantify the level of privacy protection using learning-from-abstraction. They introduced theoretical study for measuring the privacy protection related to customer's credentials with SOA. In addition, Malekand Harmantizis [13] used Rule Induction Kit (RIK) and Enterprise Data-Miner (EDM) in order to mine data log files for detecting and mining signatures and web attacks such as Denial of Service (DoS).

III. METHODOLOGY

Security Service is responsible for providing authenticity to other services and to prevent these services from potential attacks using data mining methods in order to guarantee security using WSL. The main structure of Security Service is illustrated in figure 1.

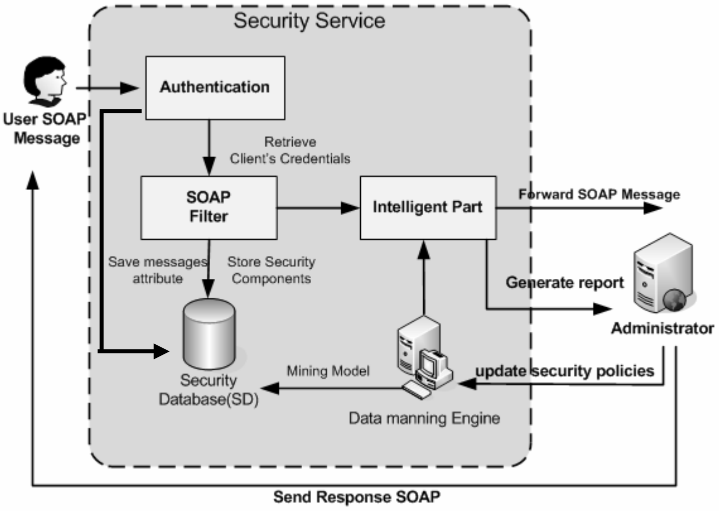

Figure 1: Security service components

The scenario of execution through this service clarifies the main functionalities:

The consumer sends a SOAP message; the authentication unit receives this message, and checks security database (SD) for user authentication credits to validate the request. The message is then forwarded to SOAP filter (Firewall) in order to extract the embedded security feature from Message Exchange Pattern (MEP) using request-response pattern. These features include security token (e.g., username/password token, Certificate X.509) and encryption algorithm.

SOAP filter passes the message after adding some additional embedded information such as timestamp and then signs the message with one of the two types (e.g., save, danger). It then updates parameter values in SDB in order to use them later.

The intelligent part request, based on using data mining engine, produces new roles. The main objective for this unit is to:

- Generate security roles to predict the attack using the Association Rule Mining Model, that proposed by El Yamany et al.[3].
- Classify consumer to "clear", "suspect" or "prohibited" according to the number of rejected messages.

- Pass message if satisfies the roles , otherwise rejects it.
- Generate reports to provider or administrator.

In order to classify the consumer, there are two thresholds in SD. Administrator specifies "Max_Trust" parameter to define the maximum number of good message that should the consumer exceeded to be in "Trust" type. Similar, the administrator specifies "Max_Untrst" parameter to define the maximum number of bad messages (i.e., danger message) that should the consumer exceed to be classified as "danger" type, otherwise consumer is classified as "suspicious". Table [1] illustrates these classes:

Table [1]: Consumer classification parameters , suppose the counter of consumer

Class	Parameter	Relation
Trust	Max_Trust	G > Max_Trust
Danger	Max_UnTrust	B > Max_UnTrust
Suspicious	Max_Trust, Max_UnTrust	G<= Max_Trust

Security Service is divided into three basic parts: Authentication, intelligent part and SOAP filter (Firewall). Consumer type is divided in to three types (i.e., Trust, suspicious, and danger).

SOAP filter (Firewall): The consumer receives the authentication credits from a previous stage. This stage runs as a parser to analysis and extract information from the message in order to sign it as rejected message or accepted message, and to save these data in SD to be available for the next unit. Figure[2] illustrates the flowchart for the process in this unit.

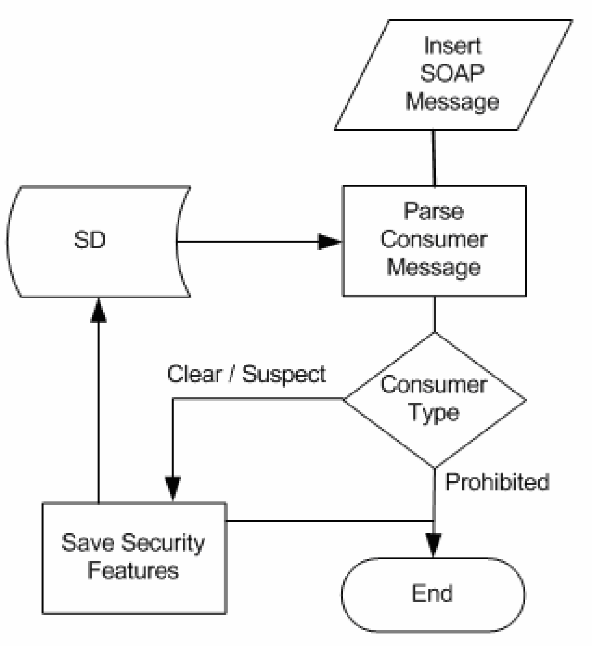

Figure 2: Firewall process

SOAP filter pares the SOAP message contents and determines if there exist any syntax error in XML contents or if it has malicious elements. SOAP filter relies on the set of parameters that are stored in SD, and known as "secure policies". The filtering policies include:

Message size restriction: This restriction specifies the limitations for the message (i.e., maximum size and minimum size), in order to prevent it from oversized payload attack. These two parameters are stored in SD in byte units.. This is a first defense line, because in this step, the system can validate this restriction without reading the details of the message. The size of the message is then stored in SD.

Syntax parsing: Through this stage, the system uses a special API component in order to check the structure of the XML by looking for syntax errors. XML should be well-formatted according to XML specification. This tool verifies the main parameters and plays the checking role:

- Checks message for the missing one or both signs of the tag.
- Checks the close tag for any open tag or vice versa.
- Checks the message for errors in original version or encoding or both.
- Each attribute in the message must have a quote. It checks for missing double quotes.
- Checks the namespace; if it has URL name differs from different web service. This filter will change the value of Syntax_Error in SD.

XML schema validation: Web Service provides set of functions, each one defined by the XML schema. Related to this, the filter uses to validate that message to conform to XML schema or not.

There set of attacks may prevented by this filter:

Parameter tampering attacks: This attack is based on manipulation of parameters exchanged between client and server, through using the information stored in cookies, hidden fields and URL queries.

Recursive payload attacks: This attack is based on modifying XML document with very deep nesting of data elements where the nesting is recursive, essentially leading to a denial of service

The running filter through this scenario

The filter validates the Namespace in the message to ensure that the received message is intended for the particular web service. (Note that the namespace is available in XML schema. Then, the filter validates the "method name" through looking for it in XML schema. Next, it checks the elements parameters by looking in XML schema in order to discover the differences. At last, it checks the number and the type of input parameter. It should be similar to those in XML schema. This unit will change the value of Schema_Erro in SD.

Message monitoring

This filter is used to prevent flooding attacks. This attack causes a denial of service. Usually, attacker use similar request messages at small period of time by repeating this message many times. This action will prevent legal user to access a web service. Particularly, each of these messages has a convergent send time.

Accordingly, using the identification information (e.g., timestamp, source address) will differentiate messages. This information will be appended to the message and stored in SD. Timestamp is known tool to deal with this type of attacks. There additional information should be then extracted in this unit to allow data mining engine produces new roles, such as parser time and consumer information.

IV. Intelligent Part Unit

The objective of this unit is to update the consumer type related the roles in table [1] and to specify the mechanism of the received letter. This objective can be done through comparing the type of the letter with the type of consumer in order to determine if the message is able to pass to other services or to be rejected otherwise. Table [2] represents the rules for dealing with the income messages.

Table [2] : The rules for income messages

Consumer Type	Message Type	Process
Trust	Good	Update good message counter, Pass the message
Trust	Bad	Update bad message counter, Pass the message
Danger	Good	Update good message counter, Reject the message
Danger	Bad	Update bad message counter, Reject then message
Suspicious	Good	Update good message counter, Pass the message, Run data mining engine
Suspicious	Bad	Update Bad message counter, Reject the message, Run data mining engine

Based on using these rules we contribute to reduce the process of running the data mining engine that takes long time to build rules online. In this case, it only runs with suspicious consumer. Therefore, consumer type may change as shown in the mechanism stated in Table 3.

Table[3]: The rule for change consumer type

From	To	Rule	Process
Trust	Danger	B> Max_UnTrust	Let G = 0
Danger	Trust	G> Max_Trust	Let B=0
Suspicious	Trust	G> Max_Trust	Nothing
Suspicious	Danger	B< Max_UnTrust	Nothing

Figure [3] shows the schema structure for security database system. The SD consists of set of tables, Token_lookup and Policy_lookup. These two tables consist of fixed information required for managing database and for updating service provide tables.

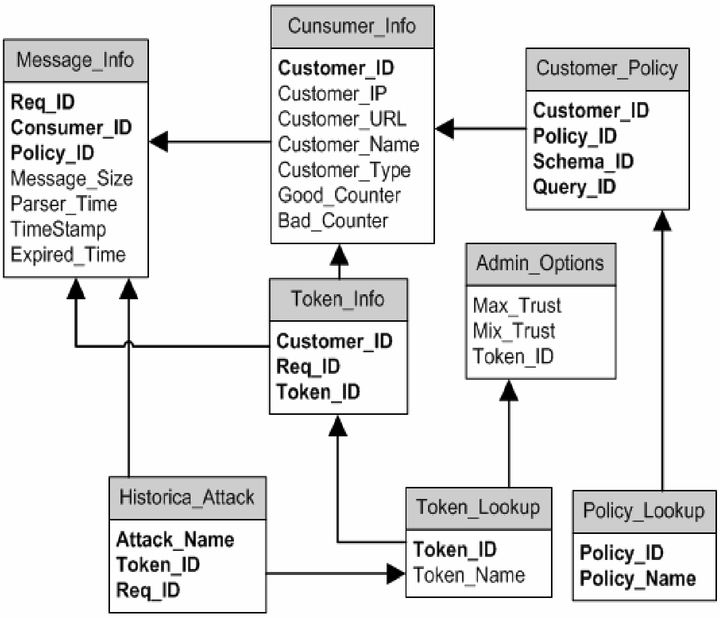

Figure [3] The schema for security database system

The other main table is Message_info to save information for all messages that pass SOAP filter, while unit.Customer_Info save information about each user connection in order to reduce the time of computing for suspicious or prohibited consumer. Consumer_policy is required to define the consumer rights and runs as access control list, Brown et al. [13]. In SBD, the relation between security token, consumer, request are captured to facilities generating security rules. Security rule such that discussed in El Yamany et al. [3], define the relations between consumer token, attacks, message size, and parse time.

V. CONCLUSIONS AND FURTHER WORK

In this study, researchers perform SOA security using a the mechanism of firewall coupled with data mining methods in order of update SOAP filter (firewall) parameters. This mechanism improves authenticity without loose of execution performance.

Researcher are looking forward to adding additional layer such as materialization of services to improve that materialization that already being used by underlined databases; and to go forward in the implementation with required empirical analysis.

VI. REFERENCES

[1] E. Bertino and L.D. martino, Aservice-oriented Approach to security –Concepts an Issue, in: The proceedings of the 11th IEEE international Workshop on future.

[2] T. Priebe, W. Dobmeier, C.Schlager and N. Kamprath, Supporting Attribute-based Access Control in Authorization ans Authentication infrastructure with Ontology, journal of software 2 (2007) 27-38

[3] H.F. EL Yamany, M.A.M. Capretz,Use of Data mining to Enhance Security for SOA, Conference on Convergence and hybrid information Technology, IEEE,2008 update 2010

[4] M. Liu, H.Guo and J.Su, An Attribute and Role based access control model for web service, in: The proceedings of the Fourth International Conference on Machine Learning and Cybernetics, vol. 2,2005.pp. 1302-1306

[5] X.Zhang, J. Park and R. Sandhu., Schema Based XML Security:RBAC Approach, http://www.25hoursaday.com/StoringAndQueryingXML.html ,May 2011

[6] C. Emig, F.Brandt, S.Abeck, J. Bierman and H. klarl, An Access Control Metamodel for Web Service-Oriented Architecture, in: The process dings of the IEEE international Conference on Software Engineering Advances (ICSE 2007), 2007

[7] M. Fernandez, W. Tan, and D. Suciu SilkRouter : Trading between Relations and XML. http://www.www9.org/w9cdrom/202/202.html , April, 2011

[8] X. Sun, XML security and relevant to E-business, Seminar on Netword security ISBN 951-22-5897-2,2002

[9] Y. Loh, W. Yau, C.Wong and W.Ho, Design and implementation of an XML firewall, in: The proceeding of the international conference on computational intelligence and security,2006,pp 1147-1150

[10] Rongbo Du, R. S.-Naini, W.Susilo, Design and Implementation of A Content Filtering Firewall, [2001] JlLawInfoSci 8; (2001) 12(1) Journal of Law, Information and Science 96

[11] X. Zhang, H. Wong, W. K. Cheung, A privacy- Aware Service-oriented platform for Distributed Data mining, in: The Proceedings of the 8th IEEE International Conference on E-Commerce Technology and the 3rd IEEE International Conference on Enterprise Computing, E-Commerce, and E-services (CEC/EEE'06),2006

[12] Manu Malekm , Data Mining Techniques for Security of Web Services, Steven Institute of Technology, Hoboken, NJ 07030, USA

[13] E. Brown, M. Bauer and M. howard., Computer Security: Principles and practice.,7-20

[14] Understanding SOA Security Design and Implementation, IBM.Corp,seconded.,2007. http://www.redbooks.ibm.com/abstracts/SG247310.html