# Web Service Security

**Dinesh Kr. Tiwari (Asst. Professor)**
Feroz Gandhi Institute of Engineering & Technology
Rae Bareli, UP, India
dk_tiwari_1975@yahoo.com

**Natthan singh( Asst. Professor)**
I. E. T. Lucknow
Lucknow, UP, India
ns_pipil@rediffmail.com

***Abstract***— Service Oriented Architecture (SOA) plays a very important role in Information System. Researchers are pointing out that the number of Information Systems based on SOA in next few years will significantly outnumber the legacy Systems. The reason behind this is the advantages that are offered by SOA itself and the technology used for development of SOA. SOA is based on Web Service (WS) Technology and inherits advantages and disadvantages of WS Technology. This is especially important in the context of SOA security issues that differ from legacy system security principles. SOA security issues are resolved through WS security solutions, like Trusted communication principles via SOAP, WS-Security, WS-SecureConversation; Trusted Web via WS-Trust, WS-Federation, and Trusted service via WS-Policy, WS-PolicyAssertion, WS-PolicyAttachment, WS-SeurityPolicy; WS-Authorization, WS-Privacy. This paper addresses the security mechanisms that are used in SOA based Information Systems in both design as well as implementation level. The brief explanation of each of the SOA security solution is given. An overview of compatibility issues as well as positive and negative sides of these solutions in SOA is also explained.

***Keywords***— Service Oriented Architecture (SOA), Web Service, WS-Security, WS-Trust, Trusted Communication, WS-Policy, WS-Privacy, Legacy System, Service-Oriented Security (SOS), SOS – Architecture (SOSA).

## I. INTRODUCTION

Service Oriented Architecture (SOA) is a flexible set of design principles for extensible, federated and interoperable services, and a new evolution in the application development and information systems. Increased number of SOA based applications brings out the SOA relates security issues that differ from legacy or traditional information system security. There are different aspects of Service-Oriented Security. Some are based on technical standards that are treated as SOA's foundation; some are organization oriented; there are security issue based on legislative SOA security policy and some based on inter-organizational cooperation, etc

Since today's Information Systems are mostly based and developed using Web service technology, they inherit multiple advantages and disadvantages of WS - Technology as well. SOA security issues are resolved through WS security solutions, like Trusted Communication principles via SOAP,

**Priyanka Gautam (CS Deptt.)**

WS-Security, WS-SecureConversation; Trusted Web via WS-Trust, WS-Federation, and Trusted service via WS-Policy,

WS-PolicyAssertion, WS-PolicyAttachment, WS-SeurityPolicy, WS-Authorization, WS-Privacy.

This paper tries to address the questions of security issues that can arise in Web services based SOA implementation. An overview of Web Service Security is given, their positive and negative sites, as well as compatibility issue in SOA.

The rest of the paper is structured as follows. Section II introduces Service-Oriented Security (SOS); section III defines SOS Ar chitecture (SOSA). Furthermore section IV outlines the Security of SOA implementation and finally section V draws conclusion and future work.

## II. SERVICE-ORIENTED SECURITY(SOS)

As mentioned in the previous sections SOA h as some unique characteristics that make it different compared to traditional Information System architecture. Because of this, SOA demands different approach of security systems [1]. SOA normally consists of more than one Web services that are developed by different organization and individuals, under different platforms; operate in different conditions and with different security systems. These components are connected into one IS that has to provide users with the same level of security as that of legacy systems.

Figure 1 shows the structure of the traditional SOA model [2]. The traditional SOA model consists of three components – service provider (SP) entities that develops and publish web services, service consumer (SC) entity that develop applications and binds to Web services developed by SP and service registry (SR) entity that acts as "yellow pages" and is used to publish Web services by SP and search desired Web services by SC. After the desired Web service is located a Service Level Agreement is defined between SP and SC. SR usually consists of several Web services developed by one or several SPs.

If the scope of SOA doesn't expand beyond organizational boundaries, then it is called internal SOA. If one internal SOA is connected and interacts with another internal SOA then they combine into external SOA (Figure 2). Internal SOA is also

known as Orchestration. Orchestration composes services within one organization. Web Service Orchestration relates to the execution of specific business processes. WS-BPEL is a language for defining processes that can be executed on an orchestration engine. External SOA is also known as Choreography.


**Figure 1. Traditional SOA Model**

Choreography composes Web services from different organization. Web Service Choreography relates to describing externally observable interactions between Web services. WS-CDL is a language for describing multi-party contracts and is somewhat like an extension of WSDL: WSDL describes Web services interfaces; WS-CDL describes collaborations between Web services [10].

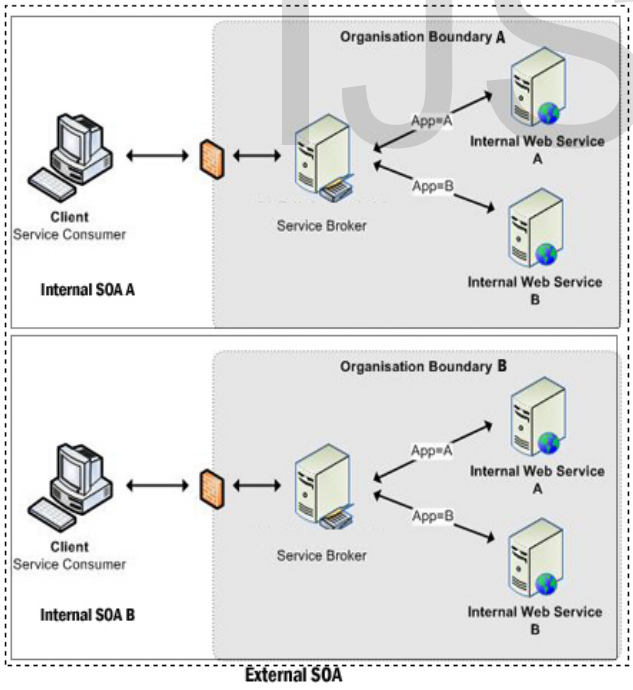


**Figure 2. Service Orchestration vs. Service Choreography**

And that leads to the major security challenges in use of SOA, which are [1],

- SOA development and usage,
- Data Security and data control,
- Authentication
- Requirement for "off-the-shelf" components.

To resolve those security issues a new field in the scope of information security was established – service - oriented security (SOS). SOS is a r elatively new field in the information security domain to resolve the security need in SOA. As mentioned above, SOA implementation is based on Web service technology. Based on this fact, SOA security issues are resolved through WS-Security.

In the area of WS-Security many security mechanisms are already developed. They can be categorized in several levels:

1. "Best Practices" defines general ways to establish WS-Security using security standards like ISO/IEC 17799, ISO/IEC 2700* standards family,
2. Standards that defines security mechanisms on implementation level, e.g. authorization and authentication, cryptography, security assertion protocols for Web service and similar, and
3. Web services implementation level – this is similar to second level – the difference is the fact it is based on implementation itself.

## III. SOS Architecture (SOSA)

To provide desired level of security in SOA above mentioned security issues have to be resolved. They can be resolved in several ways. Best solution is to use the Web Services Security Model (WSS) developed by OASIS.

WSS defines WS-Security in 4 specific areas [3], [4]:

1. **Trusted Communication** that consists SOAP, WS-Security, WS-Secure Conversation,
2. **Trusted Service** that consists WS-Policy, WS-PolicyAssertions, WS-Policy Attachment, WS-SecurityPolicy,
3. **WS-Authorization, WS-Privacy**, and
4. **Trusted Web** that consists WS-Trust, WS-Federation.

The relationship between these specifications is shown in the following Figure 3 [3] [4].

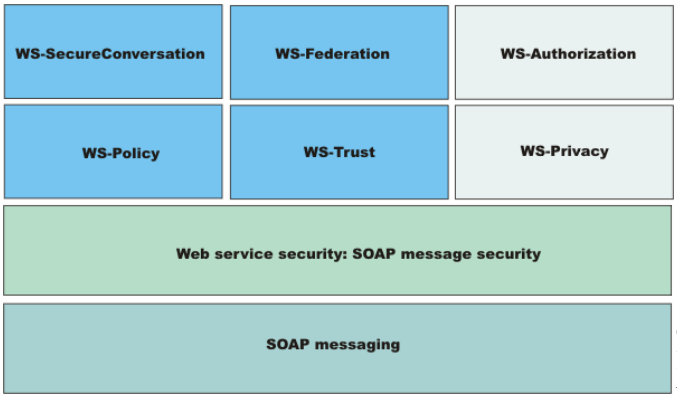


**Figure 3. The WSS Architecture Model**

If we analyse the traditional model (shown in figure 1), then the use of WSS specification on SOA development is basically reflected on SP. During WS development security elements have to be ensured through WS-Authorization, WS-Privacy, and Trusted Communication. WSS reflects on SR also – mainly in the process of WS-Composition (Web Service Orchestration and Choreography), where principles of WS-Authorization, Trusted Service, and Trusted Web are used.

### A. *Trusted Communication*

The Web Service Security specification defines a trusted layer as flexible and is used to implement wide variety of security models including Kerberos, PKI and SSL [4]. It supports different security tokens, digital signature; define trust domains, as well as various encryption techniques.

This is an extensible message header model for applying existing security techniques (digital signature, security token and encryption algorithms and methods). A trust based policy can be developed using this specification. Using this policy the recipient can trust message content and the sender. WS-Security offers a framework that can used to implement a range of security protocols. WS-Secure Conversation provides reusable shared security context for multiple SOAP message exchanges using a series of derived keys to provide increased security.

### B. *Trusted Services*

The WSS policy-based specification enables creation of extensible policy definitions and their attachment to Web service interfaces. The following standards define this policy framework [4]:

- **WS-Policy:** Defines policy, policy statement, and policy assertion models.
- **WS-Policy Assertions:** Defines generic policy assertions.
- **WS-Policy Attachment:** Defines how to associate a policy with a W eb Service, either by directly embedding it in the WSDL or by indirectly associating it t hrough Universal Description, Discovery and Integration (UDDI)
- **WS-Security:** Message Integrity Assertion, Message-Confidentiality Assertions, and Message-Security token Assertion.

WS-Policy is a predefined model for starting different types of domain-specific models: resource usage policy, transport-level security, and even end-to-end business-process-level policy. WS-Policy can incorporate policy models like SAML and XACML [4].

### C. *Trusted Web*

Purpose of Trusted Web SOA environment is to provide service provider (SP) and service consumer (SC) a direct and secure cooperation. The trusted communication enables a SP and SC to directly interoperate in a secure and trusted manner. This can be assured under assumption that the same underlying security technique is used at both endpoints. In that, they assume both endpoints are encompassed by a single "trust domain" or "sphere of trust" [5].

Trust domains with different cipher key technique (e.g. PKI, Kerberos) are not able to send secure message from one trust domain to another. This type of federated end-to-end security interaction will be enabled through "the trusted Web" [4]. To enable this federated end-to-end interaction, a trust authority that acts as a broker between trust domains has to be used. This trust authority will transform security tokens in one sphere of trust into other sphere of trust with an equivalent trust degree.

### D. *SOSA and Traditional Security Architecture*

The WSS is a service-oriented security architecture based on traditional security architectures. The WSS is an attempt to generalize the essential features of the mainstream security architecture. WSS does not define any concrete security measures. The WSS is focused mainly on federated interoperability and it does not address any of the challenges facing existing security techniques other than interoperability. The WSS is developed to solve one specific and vital question for a SOA – federated interoperability and it should be used more as a management integration technique than a management implementation model [4], [5]. The WSS is mainly concerned with a universal set of identifiers, formats, and protocols for interoperating among diverse security domains. Other security standards enable federated administration (e.g. Kerberos, Active Directory forests, PKI CAs), but each requires that all administrative domains implement the same technology. The WSS, on the other hand, does not specify any unique security technology. It is simply defined in terms of how it binds to existing standards [4], [5].

## IV. Conclusions

The process of ensuring security controls in SOA is one of the crucial prerequisites for successful use of SOA. The main characteristics of SOA-Security (availability and flexibility, interoperability and secure component management) are often stated as the critical one. The process of SOA implementation must be done parallel with Service-Oriented Security (SOS) implementation [6], [7], [8], [9].

SOS has the same level of objectives as that of any other security architecture – to ensure integrity, availability and privacy of system resources (hardware, software, data-ware....). The SOS has different preconditions than other forms of Information System Security (ISS) – mainly because of the way SOA environment is build and how it functions. One of the main technical solutions used nowadays to ensure SOA security is Enterprise Service Bus (ESB). Its role as integration layer and middleware is used to implement different security controls needed for Service-Oriented Security. An ESB can effectively ensure security issues regarding data control and integrity, SOA implement and usage, but it cannot replace the process of SOS implementation and maintenances. A systematic and continuous process of established SOS should be defined and use in SOA environment. This article addresses the question of security issues in Web services. In our future work, we intend to implement and show our proposed model fits with the real life SOA Information System.

## References

[1] Gerić, S., Hutinski, Ž., Service Oriented Security, MIPRO 30th International Convention, Proceedings of Information System Security, Opatija, 2007., pp. 125 -132.
[2] Sloane, Elliot B., "SOA What?" Tech Talk, http://www.24x7mag.com/issues/articles/2007-06_08.asp.
[3] "Web Services Security specification - a chronology", http://publib.boulder.ibm.com/infocenter/wasinfo/beta/index.jsp?topic=/com.ibm.websphere.express.doc/info/exp/ae/cwbs_wssv6chron.html
[4] "Web Services Security: SOAP Message Security 1.1(WS-Security 2004), OASIS Standard Specification", 1 February 2006 http://www.oasis-open.org/committees/download.php/16790/wss-v1.1-spec-os-SOAPMessageSecurity.pdf
[5] Gerić, S., Hutinski, Ž., Standard Based Service-Oriented Security, Proceedings of the 18th international conference "Information and intelligent systems", Varaždin, Croatia, September 2007, pp. 327-335
[6] Web Service Security, OASIS, 2006, http://www.oasisopen. org/committees/download.php/16790/wss-v1.1-spec-os-SOAPMessageSecurity.pdf, 19.03.2007.
[7] SOA and Web services: New to SOA and web services, http://www.ibm.com/developerworks/webservices, 12.03.2006
[8] Service-oriented architecture in Pervasive environment, http://www-128.ibm.com/developers/websphere/library/techarticles/0409_ammanud/, 03.04.2006
[9] Service-oriented modelling and architectures, http://www-128.ibm.com/developers/webservices/library/ws-soa-design-1/, 24.04.2006.
[10] Reynolds, John, "Service Orchestration vs. Service Choreography", http://weblogs.java.net/blog/johnreynolds/archive/2006/01/service_orchest.html, 19.01.2006