Scientific Journal of Impact Factor (SJIF): 4.72
e-ISSN (O): 2348-4470
p-ISSN (P): 2348-6406

International Journal of Advance Engineering and Research Development

Volume 4, Issue 6, June -2017

"A Critical hypothesis of upgrading security of web service by identify the location of service provider and consumer"

[1]Sandeep Srivastava
[1]*Research Scholar, Mody University of Science and Technology, Lakshmangarh.*

[2]Dr. Uma Kumari,
[2]*Asst. Prof, Computer Science Department,*
Mody University of Science and Technology, Lakshmangarh

Abstract: - *In current scenario, internet is becoming the necessary for humans. Mostly, people never want to spend time for daily need of life, like shopping, searching, utility bill payment etc. People want to complete it as soon as possible by using some web applications, which is base on the internet. There are number of web application, which gives the service or facility to complete the basic need of humans but all works, not completed by a single web application. For this issue, web service developer wish to give maximum service at the same platform. For this kind of requirements the concept of web service takes place that gives the centralization service to user at same platform. Which, enclose all service provider, in such a way that user have feeling that he interact with single web application. In this, we propose the frame work with fulfill all need and discuss the concept of web service, security threats or vulnerability, service oriented architecture and security amendment in SOAP web service protocol.*

Keywords: *Web service, Web service security, Distribution system.*

Background:

In current research, selection of right software architecture pattern & style is very important because it's a process to provide structural solution to fulfill the all technical requirement, security process and operational requirement. Shaw and Clements [1], give the definition of architecture pattern & style as "Software architecture encompasses the set of significant decisions about the organization of a software system including the selection of the structural elements and their interfaces by which the system is composed; behavior as specified in collaboration among those elements; composition of these structural and behavioral elements into larger subsystems; and an architectural style that guides this organization. Software architecture also involves functionality, usability, resilience, performance, reuse, comprehensibility, economic and technology constraints, tradeoffs and aesthetic concerns" [1]. There are number of architecture like Client/Server Architectural, Component-Based Architectural, Domain Driven Design Architectural, Layered Architectural, Message-bus Architectural, N-Tier / 3-Tier Architectural, Object-Oriented Architectural and Service-Oriented Architectural Style. Meier et al,[2] focus on the some categories like communication, Deployment, Domain, Structure and for that, they suggest best use of architecture style, like communication, Service-Oriented Architecture (SOA), Message Bus architecture are the best for this categories etc[2]. Web service [3] is the message base communication between service provider and consumer. That why, we use the SOA with n-tier architecture.

Motivation:

Recently, web service widely use service oriented architecture (SOA) base system. SOA provide the scalability, interoperability, integration of heterogonous system etc. there are number of benefit to use it. In SOA [4], its collection of different module (H/W & S/W) and component, they may be constraint from local (e.g. LAN, WAN, MAN) and global (e.g. WWW). For that, it's defined interface for communication and the rules & regulation for data exchanging, which completely system supported. Web service is the message base communication between service provider and consumer. Web service implemented SOA-base system with help of SOAP web service and rest-compliant web service. In rest-compliant web service, it's implemented as client/server architecture. Client sends the request and server will response accounting the request and forward result to client. In this architecture, interface design uniformly and perform stateless operations which manipulate the XML documents. Rest-compliant web service has not any specific security protocol. In SOAP web service, it's enclosed the XML document with in it and sends it as message format. In SOAP web service includes web service security 1.1, which is approve security protocol by Organization for Advancement of Structured Information Standards (OASIS).

A secure web service is secured the system (hardware and software component) as well as massage confidentiality, integrity, and availability. In WS-security 1.1, only focus on the "who" can use the service and "what" are the permission, he have. But it's not focus on the "where" it consumes.

Literature review:

Maolin Tang et al.[5] are working on optimization of web service, they give a hybrid-genetic algorithm which provides the optimal solution for the web service for all type of platforms but security is a big issue in this algorithm because they are not using any security mechanism, information exchange in simple text format. So that attacker can easily read the information.

B. Simon et al. [6] are working to design and develop the service interface. Interface is used for the information exchange. Information exchange is done in some specific format. This specific format, they use the XML for the message exchange and WS-protocol for source and destination addresses.

J. Chen, et al. [7] identify the threats and vulnerability present in the web service. They use the threats model to identify the threats and vulnerability. Once the threat is recognized, it becomes very important question for all; how it can be removed and component can be protected. For this question they work on the reason and design the some security mechanism to avoid.

N. Gruschka et al. [8] are working on the security issue at server side, and use the security mechanism to protect the web service and establish the trust between the two applications. N. Gruschka's main focus is to develop the trust in the broken environment and for that they are working on security token.

Paul Rabinovich [9], Introduced a simple authorization model for sharing cookies between disparate DNS domains. Author has been issued cross-domain channel authorizations granting appropriate permissions to their holders and binding these permissions crypto-graphically to the cross-domain channels' owners. Cross-domain channel authorizations may be delivered in the HTTP stream that carries cross-domain channel cookies themselves, or looked up in the DNS. Secure cross-domain channel channels allow their owners to indicate that cookies may be shared only across SSL connections; this mitigates against DNS spoofing and ensures security and confidentiality of the cross-domain channel cookies in transit.

Martin Husak et al. [10], present real-time lightweight identification of HTTPS clients based on network monitoring and SSL/TLS fingerprinting. Experiment shows that it is possible to estimate the User-Agent of a client in HTTPS communication via the analysis of the SSL/TLS handshake. They has built up a dictionary of SSL/TLS cipher suite lists and HTTP User-Agents and assigned the User-Agents to the observed SSL/TLS connections to identify communicating clients. In this, they have shown that it is possible to estimate the User-Agent of a client in HTTPS communication.

M. Swami Das et al .[11], proposed a QoS web service architecture. This is a new model of QoS web service architecture for the rapid development of web services and applications in various domains such as B2B, e-commerce banking has led to the best quest for design in a QoS of web service architecture that can meet industry standards. they proposed a modified of 3-tier architecture with a new component quality service manager that can fall in the core layer (i.e middle ware technology). This new architecture that would help the organizations in best possible QoS service architecture solutions but this architecture do not improved quality parameters bandwidth, and access time & also can not set data of QWS from WSDL crawler and other web services.

Research Plan:

Actual SOAP web service use number of security protocol, but our research plan focuses on the tract the location of service provider and consumer. Our research objectives are

1. Analysis of current security protocol WS-security 1.1, present the limitation of it.
2. Propose the new schema to overcome from limitation which identify in object 1.
3. Construct the new parser and processor for the object 2.
4. Building a performance model to evaluate Objective 1 and Objective 3.
5. Present the experimental results and analysis based on Objective 4.

WS-Security 1.1 OASIS standards, basic use in the SOAP web service which provide the security at message layer and transport layer. It's very complex and produces a lot of overhead, especially with XML encryption, XML signature and X.509-based encryption etc. SOAP message are exchange between service provider and consumer, the overhead will increase drastically. Overhead will be produced for parsing and handling SOAP messages.

However, location of service consumer is not tackled by WS-Security 1.1 OASIS standards. This will become the security hole, if, a service should only be provided for a particular enterprise or domain.

Working process of web service:

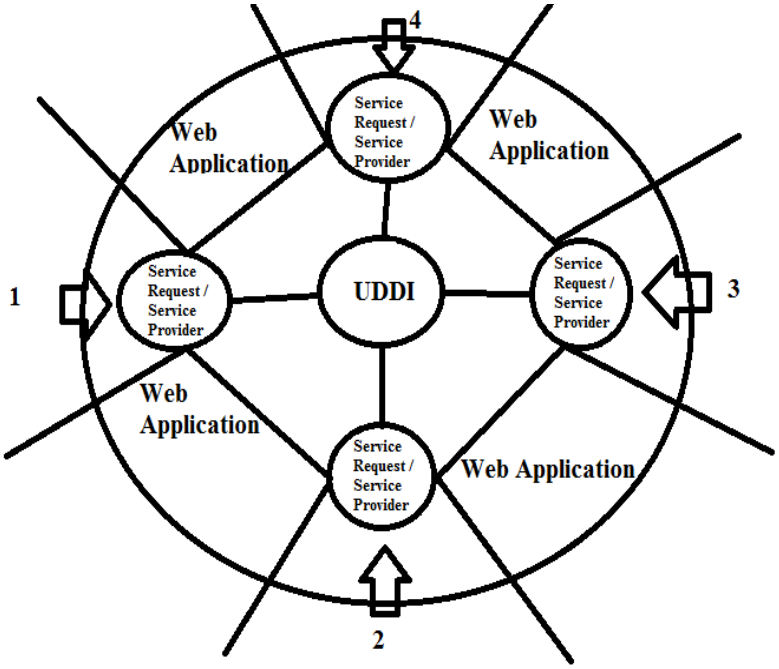
4
Service Request / Service Provider
Web Application
Web Application
1
Service Request / Service Provider
UDDI
Service Request / Service Provider
3
Web Application
Service Request / Service Provider
Web Application
2

Fig1: A basic concept of WS

Figure 1 is showing the basic concept of web service, based on the service oriented architecture. Its basic components are WSDL, XML, SOAP and UDDI. We know that WSDL is used for describing the web service, Xml is used for tag the DATA, and SOAP is Xml based data exchange protocol, and UDDI is used for register and show the availability web services.

In web service, more than two application is communicating to each other. We know that Web Service is the integration of web base application. Web applications are developed in different language, so that, web service uses XML for exchanging the information. This information needs some security mechanism. Web service is interoperable, because same work execute on different machines**.**

In Figure 1, we try focus on the concept of distribution system. Here, we introduce the four web base applications, which perform some unique work. Every web base application registered on UDDI with the help of WSDL. If users interact with web base application and requesting for some service. If the web base application is capable to serve the request then web base application fulfill the request. Otherwise, web base application forward the request to UDDI and get the list of service provider, now, web base application can chose any one and send the request and get the response. This response web base application serves to the user. In this manner, it's working and creates an virtual network, when user one, second, third and so on, Interact with system, he / She feel, it's a single system. This shows the concept of distribution system.

Security for the web service:

It's very important, completely depends on some factors like authentication, authorization, auditing / login, integrity, availability, configuration and exception management, impersonation or delegation, message encryption, message replay detection, message signing, message validation, sensitive data, session management, threats and attacks.

Proposal model are designed in such way that it consists all the above mentioned factors during its development. This model provides the end to end security, reliable, trust between client and server. More important is model's simplicity and user friendly working environment, that why it takes minimum time to establish trust and connectivity.

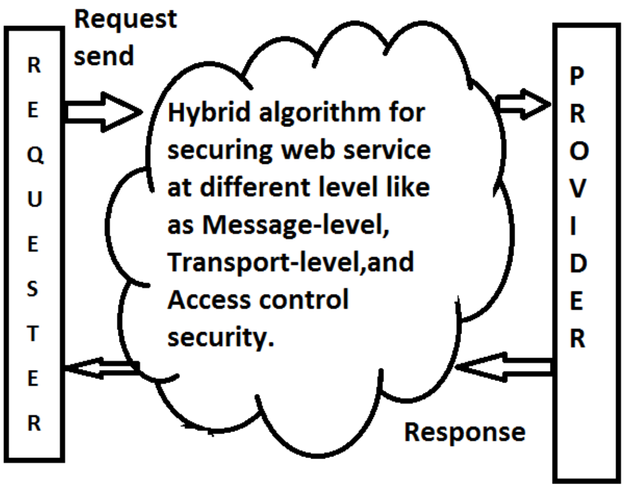
Request send
REQUESTER
Hybrid algorithm for securing web service at different level like as Message-level, Transport-level,and Access control security.
PROVIDER
Response

Fig 2: Methodology for preparing secure web service

Hybrid Algorithm:

STEP 1: Build the Secure web application using secure web development life cycle.

STEP 2: Create Web API according web service and use the Cryptography concept for securing end to end and for trust establishment.

STEP 3: Use message security mechanism for the communication securing between two applications or more, and access control define the authorization, authentication and location of use of web service component.

STEP4: Use of Secure Sockets Layer for exchanging encrypted data.

Critical Analysis on Different tools (Microsoft web service development tool kit or Amazon web service):

Attacks	**Defensive mechanism**
XML Entity Expansion	Payload analysis and validation.
XML Injection / SQL Injection	Strict validation and analysis.
SOAP attachment with viruses	Scan attachment through antivirus engine.
Man in the middle/ Brute Force Attack	Encryption, digital signatures
Network Eavesdropping	Encryption and Decryption techniques
Cross-site scripting	• URL validation / Encoding • HTML Entity Encoding / Validation

Conclusion:

I completely focus the security issues; how to identify threats, it's become very much important question. Because, criteria will change, it depends on the categories or services provided by the web applications. So that, we propose the hybrid algorithm which include the threads modeling and remove the all above mentioned security issue.

The main focus and effort of our work include

1) a algorithms for calculating the protection requirements for a service's o/p messages from those of i/p messages and the service's own requirements.

2) a method for propagating the requirements together with the flow of the data in the system.

For the future use, we may try it with different cryptography security concepts.

References:

[1] Shaw M., Clements P., “A Field Guide to boxology: Preliminary Classification of Architectural Styles for Software Systems”, *The Twenty-First Annual International Computer Software and Applications Conference*, (COMPSAC ’97), 11-15 August 1997, Washington, DC, USA, 1997, pp. 6-13, ISBN 0-8186-8105-5

[2] Meier J. D., Hill D., Homer A., Taylor J., Bansode P., Wall L., Boucher Jr. R., Bogawat, Lonnie A., “Microsoft Application Architecture Guide 2nd Edition”, October 2009, URL http://msdn.microsoft.com/en-s/library/ff650706.aspx

[3] M. Hondo, N. Nagaratnam and A. Nadalin, "Securing Web services," in *IBM Systems Journal*, vol. 41, no. 2, pp. 228-241, 2002.

[4] N. Xu, S. Peng and Z. Wang, "Designing Geodata Service Composition Web Application Based on Service-Oriented Architecture," in *IEEE Access*, vol. 4, no. , pp. 4136-4147, 2016. doi: 10.1109/ACCESS.2016.2594066

[5] M. Tang et al., "A hybrid-genetic algorithm for the optimal constrained web-service selection problem in web service composition," *IEEE Congress on Evolutionary Computation*, Barcelona, 2010,pp.1-8.

[6] B. Simon et al., "A Performance Model for the WS Protocol Stacks," in *IEEE Transactions on Services Computing*, vol. 8, no. 5, pp. 644-657, Sept.-Oct. 1 2015.

[7] J. Chen et al., "Worst i/p mutation approach to WS-vulnerability testing based on SOAP messages," in *Tsinghua Science and Technology*, vol. 19, no. 5, pp. 429-441, Oct. 2014.

[8] N. Gruschka et al,"Server-Side Streaming Processing of WS-Security," in *IEEE Transactions on Services Computing*, vol. 4, pp. 272-285, Oct.-Dec. 2011.

[9] Paul Rabinovich, “Secure cross-domain cookies for HTTP”. *Journal of Internet Services and Applications*-2013, http://www.jisajournal.com/content/4/1/13

[10] Martin Husak, Milan Cermak, Tomas Jirsik and Pavel celeda. “HTTPS traffic analysis and client identification using passive SSL/TLS fingerprinting.” *EURASIP Journal on Information Security* (2016) DOI 10.1186/s13635-016-0030-7

[11] M.Swami Das and A.Govardhan and D.Vijaya lakshmi. “QoS of Web Services Architecture”. In proceeding of the Request permissions from Permissions@acm.org. ICEMIS ’15, September 24-26, 2015, Istanbul, Turkey.2015 ACM. ISBN 978-1-4503-3418-1/15/0