ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Special Issue 15, March 2016

HTTPS IS INTEGRATED WITH HTTPI PROTOCOL TO IMPROVE THE QUALITY OF SERVICE IN WEB SERVICES

Dr. S.Chakaravarthi
Assistant Professor,
Dept. of Computer science and Engineering,
Velammal Engineering College, Chennai.
Chakra2603@gmail.com

Yuvaraju.R
PG Student,
Dept. of Computer science and Engineering,
Velammal Engineering College,Chennai.
Yuvarajr87@gmail.com

ABSTRACT

The proposed new model combines the features of Hyper Text Transfer Protocol Integrity (HTTPI) and Hyper Text Transfer Protocol Security (HTTPS) for securing Simple Object Access Protocol (SOAP) messages. In addition, there is a need to analyze the Quality of Service (QoS) parameters such as scalability, integrity, accessibility, interoperability, accuracy and non functional QoS parameters like performance, response time, security, reliability and capacity. QoS information plays a vital role in Web Services Security (WSS) oriented applications. Moreover, finding good quality of WSS is always problematic for the requestors. Delivering QoS on the Internet is a critical and significant challenge because of its unpredictable nature. With the effects of this, proposed the form combines HTTPI and HTTPS, the contradiction between security and performance is solved. The important QoS factor of security is done by adding the HTTPS characteristic such as privacy preserving with HTTPI. Already HTTPI has excellent flexibility and also has enormous security packages. In this work, an agent based approach for improving the QoS in Web services have been proposed and implemented. This Proposed agent based method improves the Quality of service with high range of security for Web services. This agent based approach also provides an intelligent solution based on the characteristics of HTTPS with HTTPI. This combined property of HTTPS is integrated with HTTPI protocol to improve the Quality of Service in Web services with the requirement such as authentication, authorization, confidentiality and integrity. This is achieved with the standards of Web services namely Web Services Description Language (WSDL), SOAP and Universal Description Discovery and Integration (UDDI). In this work, Universal Description

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Special Issue 15, March 2016

Discovery and Integration registry finds the appropriate service based on the request by service requestor and the service provider. This work also concentrates and analyses the non-functional characteristics like performance response time, reliability and security.

Keywords: HTTPS, HTTPI, Quality of Service, Web Service Security, Privacy Preserving.

1. INTRODUCTION

With the rapid growth in internet communication and the development in technology has induced many organizations and industries to migrate their applications to Web based user interaction to their services by means of WS. Hence, Web service plays a major role in day to day life in real time applications such as On-line shopping, E-ticket reservation and for various mediums of transport. Web service is a standards-based, language agnostic software entity which accepts specially formatted requests from other software entities on remote machines via vendor and transport neutral communication protocols, producing application specific responses. Web services are developed and deployed by three standards namely SOAP, WSDL and UDDI. A Web service allows an organization to build their applications in loosely coupled manner in such a way the data is separated from other dependent components and logics which allows easy integration towards in benefits of service reuse. The simplest Web services system incorporates two participants namely the Service Provider (Server) and Service Consumer (Client). The service provider presents the interface and implementation of the service and the service consumer utilizes the service by means of Web. The registry acts as a broker for Web services and the provider can publish his services to the registry while the consumer can discover and utilize the necessary services in the registry.

The major advantages of creating Web services instead of using the technology such as Common Object Broker Architecture (CORBA), Remote Method Invocation (RMI) is that, Web services are platform independent, provide a loosely coupled environment, since they use standard eXtensible Markup Language (XML). Hence the client program can be programmed in C++ which is running under Windows, while the Web service is programmed in Java that is running under Linux. In addition, most of Web services use Hyper Text Transfer Protocol (HTTP) transport medium for transmitting messages (such as the service request and response) and with the mechanism of (HTTPS), the communication becomes secured when compared to other existing techniques.

1.1 Architecture for Improving the Quality of Service

In this Paper work, an agent based approach that improves the Quality of Service and detects the various attacks to provide high range of security has been designed and implemented. Figure 1.1 shows the Universal Description Discovery and Integration registry and the communication process between the service provider and service requester. In this, the UDDI registry has the entire details of the service provider and service requester involved in communication process. The requester agent located in client side analyses the service requester is request and takes appropriate decision in forwarding the request to the service provider through the secured communication channel with the parameters namely of Authentication, Authorization and confidentiality to improve the accessibility of service which in term increases the quality of service between the service requester and service provider.

1.2 WEB SERVICES SECURITY

In the deployment of Web services in business services, security becomes as one of the important issues to be addressed. Yamaguchi et al (2007) introduced an application programming model to simplify the programming experience for end users. It was designed by looking at WS Security processing from an abstract level and also it is designed to consider correctness, efficiency, usability, flexibility, portability, and extensibility.

The major threats that affect Web services are

- Message Alteration
- Loss of Confidentiality
- Man in the Middle Attack

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Special Issue 15, March 2016

- Reply of Message Parts
- Denial of Service

1.2.1 Message Alteration

An Attacker alters an original message by inserting, removing or modifying the contents created by the sender. The faked message is then received by the receiver as being sent from the original sender's. In addition, an attacker may also construct a new fake message to fool the receiver into believing it to have come from a valid sender.

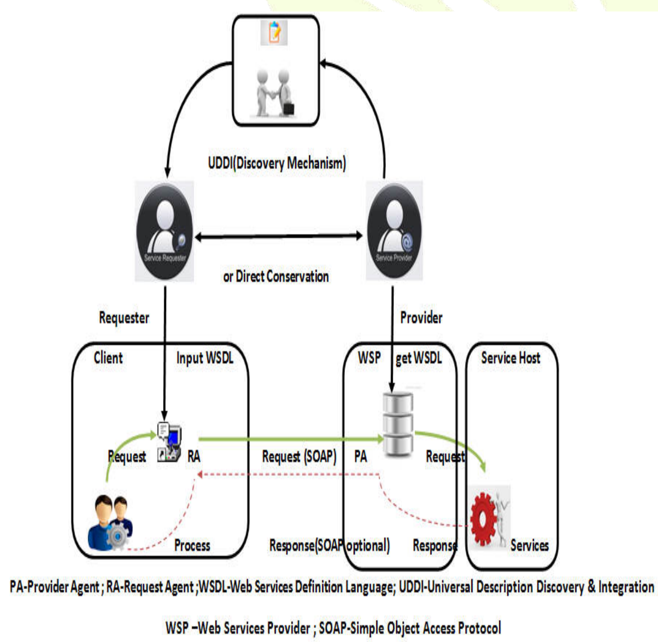

Figure 1.1 UDDI registry and the communication process between service provider and service requester

Similarly, the service provider agent analyses the service requester is request and responses and the appropriate service to the requester. The service providers are providing service by means of WSDL with the help of Universal Description Discovery and Integration registry. The communication between the service requester and service provider is by SOAP.

2. RELATED WORKS

Zhong Wu et al (2013) have been proposed and implemented to improve the performance of Web services by improving the quality of service. Some of the important works have been reported here. Among them, Cardoso et al (2004) presented the predictive QoS model that makes it possible to compute the quality of service for workflows automatically based on atomic task QoS attributes. They investigated an analytic approach that enables the service providers to deploy a differentiated service policy that offers this control. Also, they devised the relationships between the performance measures and the parameters to adopt a unique queuing theoretic approach for establishing the relationships. Another approach proposed by Yan-ping Chen et al (2006) to provide composed Web services based on the QoS requirements for clients. They were enhanced the current WSDL to describe the QoS of services, and then gave a way to choose the proper pre-exist services based on their QoS. A new approach called protecting high-security Web sites from network attacks using HTTP Security protocol was introduced by Jackson & Barth (2008). Their proposed HTTPS protocol has the potential to protect Web users from network attackers, but real- world deployments must cope with mis-configured servers, causing imperfect Web sites and users to compromise browsing sessions inadvertently. In addition, they also introduced a simple browser security mechanism called Force HTTPS for Web sites or users can use to opt in to stricter error processing, improving the security of HTTPS by preventing network attacks that leverage the browser's lax error processing.

A security issue that affects major Web sites and their customers in transaction was described by Michael Kassner (2008). Attackers exploiting this vulnerability are able to hijack an HTTP session even when the victim and the attacker's connection is encrypted using SSL or TLS. They also described the components that make up this vulnerability for those who do not have in-depth knowledge of HTTP and HTTPS. Shuo Chen et al (2009) designed an HTTP Security to provide secure Web communications over insecure networks. This designed protocol is evaluated by assuming the network as an adversary. They focused on a specific adversary named Pretty-Bad-Proxy (PBP) which is a malicious proxy targeting browsers' rendering modules above the HTTP/HTTPS layer. It also attempts to break the end-to-end security guarantees of HTTPS without breaking any cryptographic scheme. A simple benchmark was designed Shiping Chen et al (2007) and Chakaravarthi (2014) used to test the performance of the Web services against a variety of message sizes with/without applying the basic WS-Security policies. Their work provides a clear picture and some useful guidelines about WS Security performance behaviors for the current XML security technologies. The testing results show that the model is able to provide approximate performance estimation for a Web services with a variety of WSS policies and message sizes. This WSS performance model can be used by Web services architects and/or developers to evaluate the performance cost of applying Web services security.

Van Engelen & Wei Zhang (2008) analyzed the overhead of the WS-Security processing stages and evaluated the existing and new techniques for WS-Security signature performance optimizations to speed up end-to-end message integrity assurance and authentication. They analyzed that the End-to-end message security assures the participation of non-secure transport intermediaries in message exchanges, which is a key advantage for Web-based systems and service-oriented architectures. However, point-to-point message security based on TLS (Transport Layer Security) is known to significantly outperform WS-Security. Garcıa et al (2009) introduced the QoS Control Mechanism to Provide Service Differentiation and Overload Protection to Internet Scalable Servers, their first contribution was to analysis and defines the requirement that fulfills the control mechanisms and the second contribution was the design of control mechanism which fulfils these requirements and overcomes numerous deficiencies posed by previous mechanisms. The designed mechanism provides differentiation between distinct categories of service consumers as well as protection against server overloads. A service-oriented approach to generate and manage mashups was proposed by Bouguettaya et al (2010). Their approach is to realize using the Mashup Services System (MSS) which is a novel platform to support users to create, use, and manage mashups with little or no programming effort. The proposed approach relieves users from programming-intensive, error-prone, and largely nonreusable output process for creating and maintaining mashups. An sophisticated architecture for quality driven Web services evaluation was provided by Thirumaran (2010). The deployed agents are used to evaluate the QoS parameters. Their work focused about the quality attributes with organized set of design related questions which helps an evaluator to analyze the ability of the architecture to meet quality requirements, and provides a brief sample evaluation. The assessment justifies the

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Special Issue 15, March 2016

proposal in terms of the performance attributes such as reliability, availability, modifiability, security and interoperability etc. But, it doesn't protect the other types of parameter in real time environment.

A broker-based framework for integration and adaptation of non-functional aware Web services was proposed by Somayeh Karimi & Seyed Morteza Babamir (2010), they presented the Web services selection mechanism that selects the best Web services based on the both requester and provider's security issues and QoS characteristics of Web services to identify the optimal Web services composition. This proposed model is responsible for Web services collection, selection, optimization and composition based on the service consumer's requirements. It also verifies and certifies the functional and QoS specifications provided by service provider at the time of Web services registration before publishing it in Universal Description, Discovery and Integration (UDDI) registry. Zibin Zheng et al (2011) presented a collaborative filtering approach for predicting QoS values of Web services and making Web services recommendation by taking advantages of past usage experiences of service users. They first proposed a user-collaborative mechanism for past Web services QoS information collection from different service users. Secondly, they designed a collaborative filtering approach to predict Web services QoS values.

Cong Wang et al (2013) proposed a secure cloud storage system supporting privacy-preserving public auditing. The use of their proposal is to ensure the easiness of data storage and access to the information stored in the cloud. An adaptation approach that implements self-optimization based on fuzzy logic was presented by De Gyvés Avila & Djemame (2013) and Chakaravarthi(2014). The proposed optimization model performs service selection based on the analysis of historical and real QoS data, gathered at different stages during the execution of composite services. The use of fuzzy inference systems enables the evaluation of the measured QoS values helps in deciding whether adaptation is needed or not, and how to perform service selection.

3. QOS PARAMETERS FOR WEB SERVICES

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Special Issue 15, March 2016

3.1 Privacy Preserving

The other important aspect that is focused is to improve the QoS with high range of security using privacy preserving. Privacy preservation is the process by which the communication of data between the service requester and service provider is kept confidential. Figure 1.2 shows the proposed Web services process with the privacy preserving concept.

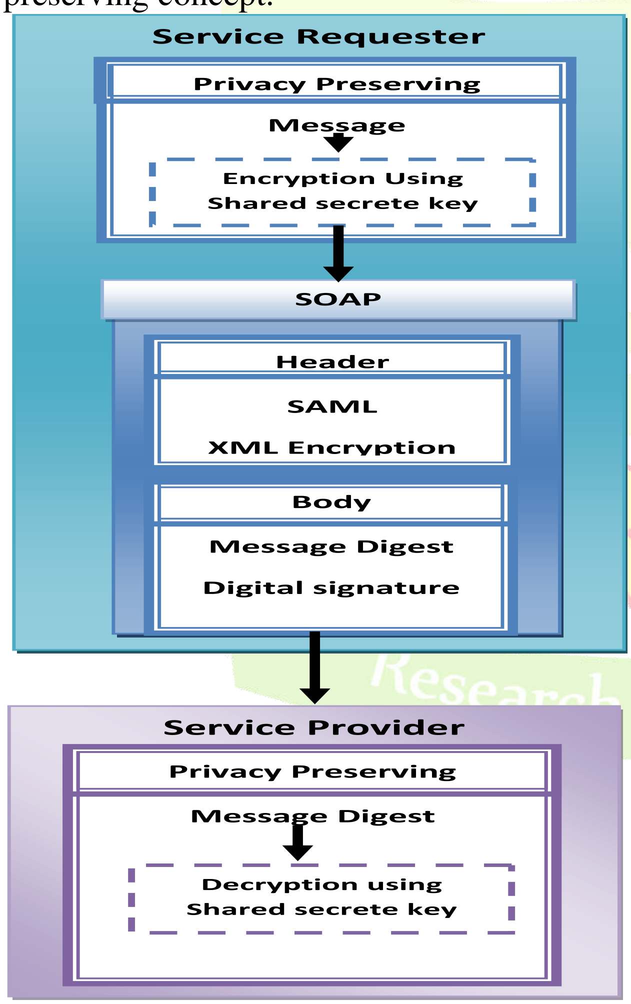

Figure 1.2 Proposed web services process with security measures

In this proposed model, the user affords the self encrypted data to the requester agent. The requester agent again encrypts the encrypted data with QoS requirements for security. This is achieved by the process of privacy preservation with the combination of Hyper Text Transfer Protocol Integrity protocol with Hyper Text Transfer Protocol Security characteristics. In this work, the property of the Hyper Text Transfer Protocol Security is integrated with Hyper Text Transfer Protocol Integrity protocol and hence satisfies the QoS requirements such as authentication, authorization, integrity and confidentiality. This QoS is achieved at all three levels namely Application level, Message level and Transport level. The service requested by the user requests the necessary service to the service providers through the transport level. The requester agent analyses the request and sends the request to the service provider with strong security. This is achieved by incorporating the security at header information which is converted into binary token form by XML encryption using Security Assertion Markup Language. The encrypted (message) header information is integrated with Hyper Text Transfer Protocol Integrity in the transport level to keep away from the man-in-the middle attack. Similarly, the requested content information is encrypted using message digest and digital signature is generated there by providing security for the entire communication at all the three levels.

4. EXPERIMENTAL EVALUATIONS

The proposed work has been implemented in a ASP.NET and C# environment in WSE 3.0 version. This approach uses several different evaluations about the request parameters and its relationship to improve the Quality of Service for Web services. The various parameters that are considered in this implementation are as follows.

In another experimental setup, we maintained configuration as I3 processor with 4 GB memory, we conducted extensive evaluations to analyze the WS and the HTTP protocol in various browsers such as Internet Explorer, Mozilla Firefox, Google Chrome for throughput, Average response time (in Milliseconds) and response message size in Kilobytes and the values are tabulated and the results are tabulated and the results are shown as graph.

4.1 Performance

The two important factors that improves the performance of Web services of Web service to improve the Quality of service are throughput and latency (response time). Hence the performance of Web services is calculated with these two factors are expressed by the formula.

$$\text{Throughput} = \frac{\text{No of Web services Request}}{\text{Given Period of Time}}$$

(6.1)

4.2 Response Time

The other important factor that ensures the performance of Web services to improve the Quality of Service in Web services is response time. The other important factor that ensures the Quality of service in Web services is response time between

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Special Issue 15, March 2016

Service Requester and Service Provider which is termed as round trip time. The throughput is directly proportional to QoS rank and latency is indirectly proportional to QoS rank. Therefore the response time is calculated using the formula.

Response time = Time taken by client + Time taken by server + Time
taken by Network + Time taken to Load
(6.2)

4.3 Reliability

The main fact among the QoS Parameters is the accuracy of the product or an application. Maintaining the exact quality of service in Web service is very important. The reliability of the Web service is measured by the number of positive response obtained in the particular span of time. The reliability of the service is calculated by the expression as follows.

$$\text{Reliability} = \frac{\text{No of Positive Responce}}{\text{Given Period of Time}}$$

(6.3)

4.4 Capacity

The capacity of the Service provider (or) the server is measured by the maximum size of the data replied in the single transaction. Sometimes the service provider will be able reply for more requests if the capacity of the server is more. The capacity of the server that it can respond at a time for the given request is calculated by.

$$\text{Capacity} = \frac{\text{Maximum Size of Reply Data}}{\text{Single Transaction}}$$

(6.4)

4.5 Security

The security is an important aspect with the provision of QoS in Web services, even though, the benefits of service provided by the service provider is more, it does not benefit really if the security is not provided properly. Hence in this work, security is incorporated by providing authentication, authorization, confidentiality and integrity with the QoS factors. The above factors are improves the QoS in communication between the Service Requester and Service Provider.

5. EXPERIMENTAL RESULTS

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Special Issue 15, March 2016

Table 1.1 shows the results obtained from the experiments conducted in this research work with various browsers to analyses the Response time comparison of each browser.

Table 1.1 Comparison of different response time of HTTP with different Browsers

Different Cycle with Input Size: 26	Response Time in milliseconds		
	Internet Explorer	Fire Fox	Google Chrome
Cycle -1	34.2	33.3	26.5
Cycle -2	15.7	16.3	11.6
Cycle -3	15.4	17.4	17.6
Cycle -4	15.6	17.1	16.8
Cycle -5	15.2	16.8	16.5

Table 1.1 describes the comparison of different response time of HTTP with different browsers. In this, it is observed that the response time of every browser is at the peek at the first transaction and gradually reduces to its normal response time. Internet explorer has very high response time at the first transaction and has very low response time in the following transactions than other two browsers. Google chrome has the lowest response time at the first transaction and then has the steady flow in the following transaction. Mozilla Firefox has the highest response time in comparison with the other two browsers.

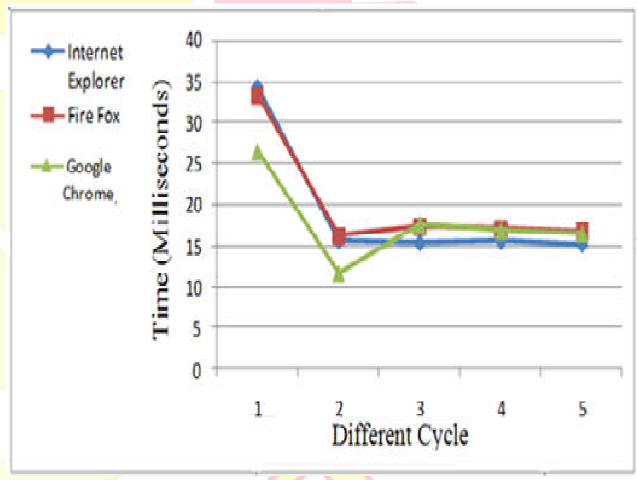

Figure 1.3 Comparison of different response time of HTTP with different Browsers

From Figure 1.3, it is clear that the response time indicates that our proposed model provides good results with the existing browsers when compared with other Browsers.

Table 1.2 Comparison of different reply size of HTTP with different Browsers

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Special Issue 15, March 2016

Different Cycle with Input Size: 26	Reply Size in Bytes

Different Cycle with Input Size: 26	Throughput in Bits/second		
	Internet Explorer	Fire Fox	Google Chrome
Cycle -1	25	15	38
Cycle -2	38	39	61
Cycle -3	40	46	57
Cycle -4	39	41	54
Cycle -5	38	39	53

Figure 1.4 shows the Comparison of different through put of HTTP in different browsers, in which every browser has the same reply size because the return value of the Web service will be same for all communications.

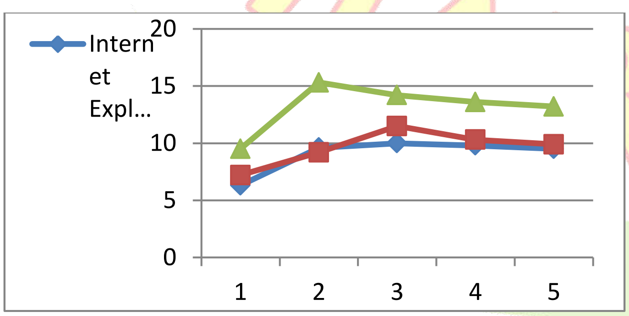

Figure 1.5 Comparison of different throughput of HTTP with different Browsers

From the graph shown in Figure 1.5, the throughput indicates that our proposed model provides good performance with the existing browsers when compared with other types of Browsers.

6.4 CONCLUSION

In this work, an effective approach that improves the Quality of Service with high range of security has been proposed and implemented by combining the characteristics of Hyper Text Transfer Protocol Security with the Hyper Text Transfer Protocol Integrity protocol in communication. The overall performance of Web service is improved when the characteristic of Hyper Text Transfer Protocol Security in permuted with Hyper Text Transfer Protocol Integrity in addition to high range of security. In addition, the privacy preserving is done by encrypting the header information by the technique XML encryption with SAML and the body content is encrypted using message digest where the digital signature is generates in order to provide high range of security with QoS.

REFERERENCES

1. Bouguettaya, A, Nepal, S, Sherchan, W, Xuan Zhou, Wu, J, Shiping Chen, Dongxi Liu, Li, L, Hongbing Wang & Xumin Liu 2010, 'End-to-End service support for mashups', IEEE Transactions on Services Computing, vol. 3, no. 3, pp. 250-263.

2. Cardoso, J, Sheth, A, Miller, J, Arnold, J & Kochut, K 2004, 'Quality of service for workflows and web services processes', Journal of web Semantics, Elsevier, pp. 1-40.

3. Cong Wang, Chow, SSM, Qian Wang, Kui Ren & Wenjing Lou 2013, 'Privacy-Preserving public auditing for secure cloud storage', IEEE Transactions on Computers, vol. 62, no. 2, pp. 362-375.

4. De Gyves Avila, S & Djemame, K 2013, 'Fuzzy logic based QoS optimization mechanism for service composition', Proceedings of the 7th IEEE international symposium on service-oriented system engineering, pp. 182–191.

5. Garcıa, DF, Garcı'a, J, Entrialgo, J, Garcı'a, M, Valledor, P Garcı'a, R & Campos AM 2009, 'A QoS control mechanism to provide service differentiation and overload protection to internet scalable servers', IEEE Transactions on Services Computing, vol. 2, no. 1, pp.3-16.

6. Jackson, C & Barth, A 2008, 'ForceHTTPS: protecting high-security web sites from network attacks', Proceedings of the 17th ACM international conference on world wide web, pp.525-534.

7. Michael Kassner 2008, HTTPS: surf jacking makes it vulnerable, Available from: <http://enablesecurity.com>. [19 August 2008].

8. Shiping Chen, Zic, J, Kezhe Tang & Levy, D 2007, 'Performance evaluation and modeling of web services security', Proceedings of the IEEE international conference on web services, pp. 431-438.

9. Shuo Chen, Ziqing Mao, Yi-Min Wang & Ming Zhang 2009, 'Pretty-Bad-Proxy: an overlooked adversary in browsers' HTTPS deployments', Proceedings of the IEEE symposium on security and privacy, pp. 1-13.

10. Somayeh Karimi & Seyed Morteza Babamir 2010, 'A QoS broker based architecture for secure intelligent web services composition', Proceedings of the international conference on communications engineering, pp. 110-183.

11. Thirumaran, M, Dhavachelvan, P, Abarna, S & Aranganayagi, G 2010, 'Architecture for evaluating web services QoS parameters using agents', International Journal of Computer Applications, vol. 10, no. 4, pp. 15-21.

12. Van Engelen, RA & Wei Zhang 2008, 'An overview and evaluation of web services security performance optimizations', Proceedings of the

IEEE international conference on web services, pp. 137-144.

13. Yan-ping Chen, Zeng-zhi Li, Qin-xue Jin & Chuang Wang 2006, 'Study on QoS driven web services composition', Proceedings of the lecture notes in computer science, springer, pp. 702-706.

14. Zhong Wu, Guihua Nie, Donglin Chen & Peilu Zhang 2013, 'web services description framework and selection mechanism towards balanced perceptions of FQoS and QoS', Journal of Theoretical and Applied Information Technology, vol. 47, no. 1, pp. 398-405.

15. Zibin Zheng, Hao Ma, Lyu, MR & King, I 2011,'QoS-Aware Web services recommendation by collaborative filtering', IEEE Transactions on Services Computing, vol. 4, no. 2, pp. 140-152.

16. Chakaravarthi, S & Selvamani, K 2014, 'A flexible web services environment with high range of security and QoS mechanisms', Journal of Theoretical and Applied Information Technology, vol. 61, no.1, pp. 60-66.

17. Chakaravarthi, S & Ramachandran, V 2014, 'Permutation of HTTPI and HTTPS in web services against attacks for security enhancement', Journal of Computer Science, vol. 10, no.3, pp. 521-529.