Integrity and Security Model using Web Services

Yuvaraju.R[1] Dr. Chakaravarthi.S[2]
[1]M.Tech. Student [2]Assistant Professor
[1,2]Department of Computer Science & Engineering
[1,2]Velammal Engineering College Chennai, India

Abstract— Now a days, We are using "open Applications", are (e,g Social Networking ,News and Blogging), these applications are non-confidential. My work involves securing the web services over SOAP, based on the HTTPI. Approach through an experiment and show that provides higher throughput, lower average response time and lower response size than HTTPS based web service security. The end to end security guarantee of HTTPs only allows web content Delivery Networks (CDN). Based on these observations, we have designed a lite protocol for secure web HTTP Integrity. HTTPI relies on HTTPS to share session keys and use them for keyed hashing HTTP page.

Key words: HTTPS, HTTPI, Quality of Service, Web Service Security, Privacy Preserving

I. INTRODUCTION

Web services provide a loosely coupled environment for building distributed systems with universal access and interoperability. This is also used to pack data into XML messages defined by SOAP (Simple Object Access Protocol) and also uses XML to describe the data types and services in the SOAP message. In addition, WSDL provides interoperability and describes the web service standards in a meaningful way.

It is used for easy integration even if they are developed in different layers. It also brings high risks to both sides of the message exchanges. SSL provides point-to-point security but for WSS environment we need to provide end-to-end security in which multiple intermediate nodes could exist between the two end-points. SSL operates only at the transport level but not at the application level. (i) Extra CPU times to process WSS-related elements and operations at client and service ends, (ii) Longer networking times to transport larger SOAP and messages due to additional WSS contents. In observation on above expenses, we propose the enhanced WSS model that includes integrity and security and intelligent agents for web service security, which can monitor the provision of service. The remainder of this paper is organized as follows: Section 2 provides a survey of related works. Section 3 describes the concept of Web Service Registration and Routing System (WSRRS) and Inter-Web Proxy Service (IWPS) based on the combination of SOAP security measures. Section 4 describes the concept of combination of HTTPI and HTTPS in WSS against attacks. Section 5 analyzes the experimental results of the proposal and compared with various aspects of previous proposals.

We also analyze the performance of our approach through an experiment and show that our proposed approach provides higher throughput, lower average response time and lower response size than HTTPS based web service security approach

A. HTTP:

HTTP does not provide any security assurance, but it is flexible, lightweight and supported by cache proxies in Internet. On the other hand, HTTPS (HTTP over TLS/SSL) provides all three security assurances

B. HTTPS:

HTTPS is less flexible, heavyweight, has no support for cache proxies and also additional latency in network

1) HTTPI:

A new protocol was proposed for these open applications, with two security guarantees, data integrity and client/server authentication, but no guarantee of any data confidentiality Benefit of using HTTPI in place of HTTPS is support by cache proxies and security against many cyber-attacks like Server Impersonation and Message Modification, Our work involves securing the web services over SOAP, based on the HTTPI

II. RELATED WORK

A. Security of Web Services

With Web services it has become possible to integrate Web-based applications using the XML[1][2], SOAP, WSDL and UDDI open standards over the Internet. XML is used to tag the data, SOAP – to transfer it, WSDL – for describing the available services and UDDI – for listing what is available. Web services are primarily used in business to communicate data with each other (B2B) and with clients (B2C).

Web services extensions and protocols they accommodate a wide variety of security models and security technologies. The problem with every security protocol is its vulnerability to a wide variety of attacks.

Both concrete and complex security mechanisms and methods to deal with these threats are utilized in our integrated security model.

B. HTTPI for Practical End-to-End Web Content Integrity

[2]SOP defines a principal model where web sites are mutually distrusting principals and where one site's script cannot access another site's content. However, the authenticity of the principal and the integrity of its content are often at question since much of the web is delivered over HTTP rather than HTTPS.

C. HTTP Integrity: A Lite and Secure Web against World Wide Woes

[3][4] As the World Wide Web focuses on scalability and performance rather than security, it suffers widely from classes of attacks including server impersonation, message modification, cookie theft and cookie injection. Contributing to these problems, wireless networks proliferate, and any attacker can easily eavesdrop and modify traffic from web clients in his or her proximity areas.

D. HTTPI: An HTTP with Integrity

[4]The World Wide Web supports two well-known transport protocols: HTTP [1] and HTTPS [2]. These two protocols have different costs and provide different security guarantees for the web applications deployed on top of them. At one end, HTTP is inexpensive to use but provides no security guarantees for any web application deployed on top of it. At the other end, HTTPS is expensive to use but provides three important security guarantees for any web application deployed on top of it. These three security guarantees are server authentication, message integrity, and message confidentiality

E. SOAP Processing Performance and Enhancement

[6] Simple Object Access Protocol (SOAP) is the protocol specification for message exchange amongWS. It is based on the XML data model, and usually relies on existing application layer protocols (e.g., HTTP, FTP, SMTP, etc.) for message negotiation and transmission In this paper, we adopt the following terminology: the process of translating a memory object according to a serialization format into an XML object is called serialization. The process of converting an XML structure into a memory object will be called deserialization.

F. Quality of service measure approach of web service for service selection

With the increasing popularity of the development of service-oriented applications, the truthfulness of quality of service (QoS) becomes an imperative concern for service consumers. It has great influence on degree of the service usability and utility, both of which influence the popularity and application of web service . Hence, QoS measure is crucial for selecting web services to take part in seamless and dynamic integration of business applications on the web. However, owing to the uncertainty of web service environment such as dishonest service providers, the different context of customers, QoS of web service often fluctuates with time.

G. Detecting Application Denial-of-Service Attacks:

A Group-Testing-Based Approach [5][6]DENIAL-OF-SERVICE (DoS) attack, which aims to make a service unavailable to legitimate clients, has become a severe threat to the Internet security. Traditional DoS attacks mainly abuse the network bandwidth around the Internet subsystems and degrade the quality of service by generating congestions at the network. Consequently, several network-based defense methods have tried to detect these attacks by controlling traffic volume or differentiating traffic patterns at the intermediate routers. However, with the boost in network bandwidth and application service types, recently, the target of DoS attacks has shifted from network to server resources and application procedures themselves, forming a new application DoS attack

III. PROPOSED SYSTEM

- Approach provides higher throughput, lower average response time and lower response size than HTTPS and HTTPI based web service security
- HTTP has nearly same performance as the HTTP and much better performance the HTTPS

IV. SYSTEM ARCHITECTURE

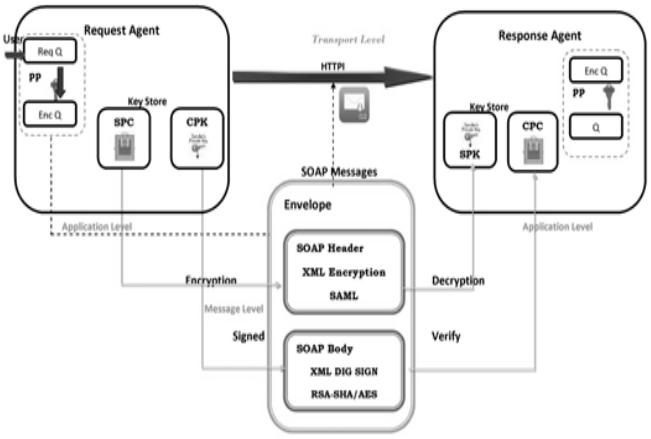

Fig. 1: System Architecture

After completing the UDDI registry, the web user want to use a public WS, client used to send an input (request) through requester agent, The Req Q requested query encrypt with sender's private key, After encryption the query converted into encrypted query Enc Q the step2 and step 3 is dealing with privacy preserving concept, The privacy preserving should be any software or application which hide the user data from the hackers, the application level security maintains here and its optional too, The encrypted data will be formed as SOAP message while the data transferring the same in HTTPI protocol. The SOAP message encrypted and signed using receiver's Public Certificate and sender's Public Key respectively,

XML Encryption - The SOAP header is bonded with self-signed certificate - Security Assertion Mark-up Language (SAML): binary token, XML Digital Signature - The SOAP body content is signed (integrity) using RSA SHA1 algorithm, The digested SOAP message is transferred to HTTPI which provide more secure data and avoid man in the middle attack. The encrypted request from the requestor as a SOAP message received from the receiver in the other end, The SOAP message decrypted and verified using Receiver's Private Key and Sender's public Certificate respectively The server/services provider got Encrypted Query from the above process Enc Q, The Encrypted Queries Enc Q have been decrypted using sender's public key and get the resultant query Q The query Q which provides web user data has been analyzed and responded

V. CONCLUSION

In this paper, we propose a web service security model based on HTTPI protocol over SOAP, with the security goal: client/server authentication and integrity on message, without confidentiality. As per our proposed scheme, we used Username/Password Tokens and Binary Authentication Tokens (X.509 certificates) for Authentication and XML Digital Signature (with RSA-SHA1 as a signature algorithm) for Message Integrity. We set up a Non-Encrypted session to secure the communication between two web services. We examined the performance of our scheme through an experiment.

From the results of our experiment, we conclude that our HTTPI based web service security scenario provides higher throughput (in transaction/seconds), lower

average response time (in milliseconds), and lower response size (in KB) than HTTPS based web service security scenario, when there is no need of message confidentiality, and having little overhead over the Non-secured and Username/Password scenario. Thus, the secured web services based on HTTPI can be used in non-confidential open applications (like: Social Networking, Blogging and News sites) in future to secure them effectively and efficiently in terms of authentication and integrity.

REFERENCES

[1] T. Choi and M. G. Gouda, (2009) "HTTP Integrity: A Lite and Secure Web against World Wide Woes", Department of Computer Science, the University of Texas at Austin, Tech.Rep.TR09-41.
[2] T. Choi and M. G. Gouda, (2011) "HTTPI: An HTTP with Integrity", Department of Computer Science, the University of Texas at Austin, IEEE.
[3] Singh K, Wang H, Moshchuk A, Jackson C, Lee W. (2011) "HTTPi for practical end-to-end web content integrity", In: Microsoft technical report.
[4] Hirsch, F. (2006). "Web Services Security: SOAP Messages with Attachments (SwA) Profile .1", OASIS Standard.
[5] Nadalin, A., Kaler, C., Hallam-Baker, P., Monzillo, R. (2004). "Web Services Security: SOAP Message Security 1.0", OASIS Security Standard.
[6] Atkinson B., Libera M., Hada M. and Hondo I. (2002). "Web Services Security (WSSecurity)", IBM, Microsoft, VeriSign, [Online] Available: http://www.106.ibm.com/developerworks /webservices/library/ws-secure/
[7] Jensen, M., Gruschka, N., Herkenhoner, R., and Luttenberger, N. (2007). "SOA and Web Services: New Technologies, New Standards - New Attacks", Proc., 5th IEEE European Conference on Web Services (ECOWS), pp. 35-44.

