Efficient Prototype of Secure SOAP Message Transmission in Web Service

Amtul Waheed
Prince Sattam Bin Abdul Aziz University, College of Arts & Science, Riyadh Region, Saudi Arabia

Jana Shafi
Prince Sattam Bin Abdul Aziz University, College of Arts & Science, Riyadh Region, Saudi Arabia

Abstract

With the growth of web services security issues based on heterogeneous platform have become gradually more prominent. Web services Security provides basic means to secure SOAP messages. Secure transmission of SOAP messages play an essential task for the applicability of Web Services. The main confront to the secure transmission of SOAP messages includes: confidentiality, authentication, integrity, and both-party no repudiation. We explored and procure advantage of existing technologies to propose a Prototype of Secure Transmission (POST) for the main mechanism of Web Services security and secure communication between client and web server in heterogeneous platform. The study in this paper indicates that for the basic requirement towards secure transmission of SOAP messages, our Prototype of Secure Transmission (POST) is ensuring a high level security to SOAP message transmission over heterogeneous platform.

Keywords: SOAP, Web Services, Prototype of Secure Transmission (POST), Security

I. INTRODUCTION

Web services usage has gone far ahead of anticipation and prolongs to expand more in future. Services on the Internet are also more innovative than ever before. Web services provide a structure for system Integration without depending on network topology, programming language and operating system.

How to guarantee the security of services has become hot spot in foreign research institutions and scholars. IBM Tokyo Research Institute (Fumiko Satoh et al.) puts forward the best practice models and support tools for the specific service safety profile construction for the IBM Web sphere Server according to security policy using mapping rules [1].

Microsoft Research in University of Cambridge (Karthikeyan Bhargavan et al.) [2] Publishes a security policy configuration guidance to help developers construct the security policies of Web service according to security requirements. IBM Research Division in New York (Sam Weber et al.)[3]

SOA (service oriented architecture) is a service that aims to integrate and be interoperable with various implementation languages. SOA is a concept of distributed computing on heterogeneous platforms.

One of the important terms on which Web Services rely heavily is SOAP (Simple Object Access Protocol). In terms of a services-oriented architecture, SOAP is used to send data from one application to another in heterogeneous environment communication between client application and Web service is possible by exchanging XML documents. Simple information exchange protocol applied in dispersed or distributed Environment. SOAP's main advantage is loosely coupled [4]. Seen in terms of a service-oriented architecture, SOAP allows for applications to bind to other applications in order to make use of their functionality. SOAP can either be used for messaging between applications (called "Document-based SOAP") or for Remote Procedure Calls (called "RPC SOAP"). Both of messaging and RPCs are the important aspects of SOAP, but in most cases, messaging is preferable to RPC, since it means that applications do not have to share an object model, or rely on a synchronous always-on connection[6]. SOAP is defined as an enveloping protocol, so it is sometimes seen as a messaging protocol as well as a means of using functionality that is published by a remote application.

There are applicable security principles (WS-Security) of security information exchange linking diverse policy, WS-Security only gives a conceptual structure to attain security objective, together with XML signatures, encryption, authentication and authorization. As for how to use them to achieve the goal of SOA security, it presents a challenge both in theoretical and technical practices [7,8].

Web services security policies in heterogeneous platforms have been proposed in this paper by creating a secure SOAP message transmission prototype.

Pooled with existing functional paradigm, user authentication during a Web service communication as well as the safe handling of SOAP messages in heterogeneous platforms is achieved. The security prototype offers approximate hold on the security interaction of Web services in heterogeneous platforms and is verified by experiments. This Prototype gives assurance to security interactions of Web service effectively on heterogeneous platforms.

II. RELATED WORK

The development of the Web Services Security specification includes information on the Organization for the Advancement of Structured Information Standards (OASIS) Web Services Security specification. The OASIS Web Services Security specification serves as a basis for securing web services in WebSphere Application Server. [8]

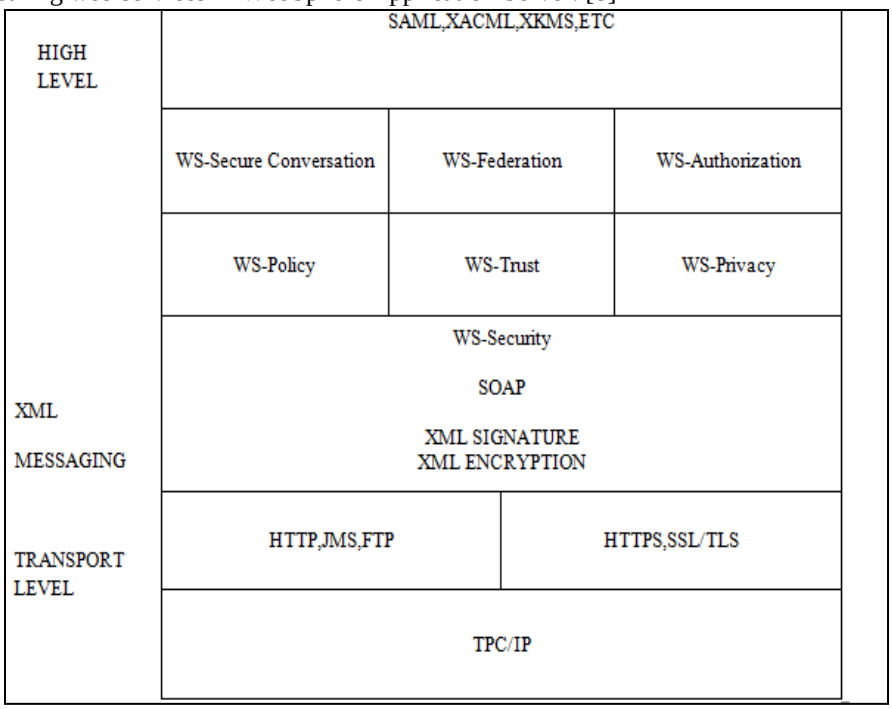


Fig. 1: Basic structure of web services architecture

This section presents the specifications used in the Web services architecture (as shown in figure)to provide message integrity, authentication and confidentiality, security token exchange, message session security, security policy expression, and security for a federation of services within a system. The specifications providing these features are WS-Security, WS-Trust [WS_Trust], WS-SecureConversation [WS-SecureConv], WS-SecurityPolicy [WS-SecurityPolicy], and WS-Federation.

A. XML security standards:

1) XKMS (XML Key Management Specification)
XKMS is a Web Service that provides an interface between an XML application and a Public Key Infrastructure (PKI). XKMS greatly simplifies the deployment of enterprise strength Public Key Infrastructure by transferring complex processing tasks from the client application to a Trust Service.XML Key Information Service Specification (X-KISS) defines a protocol for a trust service that resolves public key information contained in XML Signature <KeyInfo> elements.XML Key Registration Service Specification (X-KRSS) defines a protocol for a trust service that accepts registration of public key information. Once registered, the public key may be used in conjunction with other web services including X-KISS [10].
2) SAML(Security Assertion Markup Language)
is an XML-based, open-standard data format for exchanging authentication and authorization data between parties, in particular, between an identity provider and a service provider. The SAML specification defines three roles: the principal (typically a user), the Identity provider (IdP), and the service provider (SP). In the use case addressed by SAML, the principal requests a service from the service provider. The service provider requests and obtains an identity assertion from the identity provider. On the basis of this assertion, the service provider can make an access control decision – in other words it can decide whether to perform some service for the connected principal [11].
3) XACML (Extensible Access Control Markup Language)
the standard defines a declarative access control policy language implemented in XML and a processing model describing how to evaluate access requests according to the rules defined in policies.
4) Web service security:
This specification defines how to attach a digital signature, use encryption, and use security tokens in SOAP messages.
5) WS-Policy:
This specification defines the language that is used to describe security constraints and the policy of intermediaries or endpoints.
6) WS-Trust:
This specification defines a framework for trust models to establish trust between web services.

7) WS-Privacy:
This specification defines a model of how to express a privacy policy for a web service and a requester.
8) WS-Secure Conversation:
This specification defines how to exchange and establish a secured context, which derives session keys between web services.
9) WS-Authorization:
This specification defines the authorization policy for a Web service. However, the WS-Authorization specification has not been published. The existing implementation of Web Services Security is based upon the Web Services

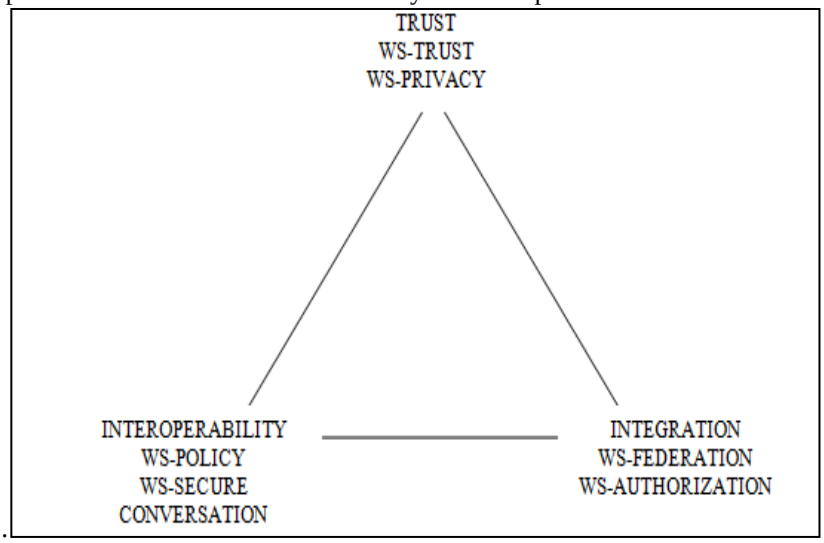


Fig. 2: Specification of Web service security

The recent specifications WS-Trust, WS-Secure Conversation and WS-Authorization provide mechanisms for communicating parties to establish shared security contexts and to use them to secure SOAP-based sessions as shown in figure 2. Web services are built on asynchronous communication of SOAP envelopes [W3C 2003].SOAP often travels over HTTP, but can also use other transports. The mechanisms of WS-Security provide means to secure these messages at the application level to achieve end-to-end security.

WS-Trust ,WS-Secure Conversation and WS-Authorization Building on top of WS-Security, WS-Trust describes how security tokens can be requested and issued by SOAP processors; it relies on a dedicated security token service (STS) to evaluate requests and issue tokens. Moreover, WS-Secure Conversation describes the usage of one such token, named a security context token. The token points to a security context (SC) typically shared between a client and a web service; its content can be used to derive keys to protect traffic between these two parties.

And WS-Authorization defines the authorization policy for the existing implementation of Web Services Security is based upon the Web Services. [9]

Security techniques of Web services in heterogeneous platform provide security policy configuration and security implementation method of SOAP message. Security service agents use WS-Security and other specifications to achieve the following aspects of security challenges of SOAP message [12]:

10) Confidentiality:
Unintended parties should not be able to understand the message. Special care must be taken especially because of SOAP's ASCII format. Otherwise even an amateur hacker can sniff the message.
11) Integrity:
Web service works in a distributed environment. The message may be received and forwarded on by an intermediary, who may not be completely trusted. No one should be able to modify the message during the transfer without being detected
12) Authentication:
We want to verify that the response is really originated from the claimed sender.
13) No-repudiation:
The sender must not be able to deny that he ever sent the message. This is one step above authentication

SOAP, which is a messaging protocol based on XML, is about sending messages, meaning that it specifies a way to send XML-based messages from one process to another, usually from one machine to another[8]. More Specifically, SOAP is a protocol that specifies an enveloping mechanism for sending data (via XML). Furthermore, it specifies how to send these messages to a final destination, and the processing model that applies if that message goes through several intermediaries. And, it specifies how to do this over HTTP.

The SOAP specification describes four major components: formatting conventions for encapsulating data and routing directions in the form of an envelope, a transport or protocol binding, encoding rules, and an RPC mechanism.

A SOAP message consists of an envelope containing an optional header and a required body, as shown in Figure 3.

Envelope, the topmost container, comprises the SOAP message; Header contains additional blocks of information about how the body payload is to be processed; and Body contains the actual message to be processed. Each element contained by the Header is called a header block. The purpose of a header block is to communicate contextual information relevant to how the message is to be processed. This includes routing and delivery settings, authentication or authorization assertions, and transaction contexts. XML elements and attributes for the purpose of SOAP security are just placed inside the SOAP header. The body contains the actual message to be delivered and processed. Anything that can be expressed in XML syntax can go in the body of a message.

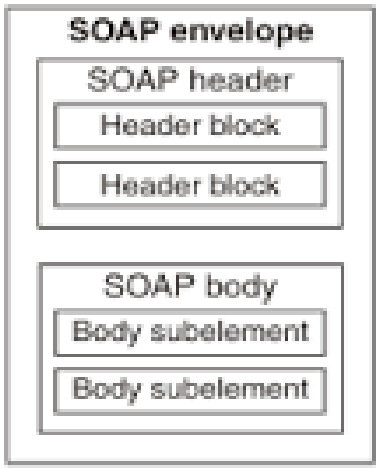


Fig. 3: SOAP message envelope architecture

SOAP does not yet have a standard binding for reliable messaging. The security provided by HTTPS cannot satisfy the more and more complicated requirement of SOAP message security. A number of technologies and solutions have been developed for the security of SOAP message transit [13].

XML Encryption provides not only a way of encrypting portions of XML documents, but also a means of encrypting any data and rendering the encrypted data in XML format. XML Encryption is ideal for confidentiality. The ability to selectively encrypt XML data makes XML Encryption very useful for Web Services. By selectively encrypting data in the SOAP message, certain information may be hidden from SOAP intermediaries as it travels from the originator to the destination Web Services [13].

XML Signature explains how to express the digital signature of any data as XML, as well as explaining how to digitally sign portions of an XML document. The power of XML Signature for Web Services is the ability to selectively sign XML data. For example, if a single SOAP parameter needs to be signed but the SOAP message's header needs to be changed during routing; an XML Signature can be used that only signs the parameter in question and excludes other parts of the SOAP message. If the SOAP request passes through intermediaries en route to the destination Web Service, XML Signature ensures end-to-end integrity [13].

III. PROBLEM STATEMENT:

To afford the security in transport level is extremely complex since the message handouts all the way through numerous intermediary points, so communication security can be offered by several encryption methods.

So it gets better the flexibility of the structure. Web services are release set (XML, SOAP, and HTTP etc.) based Web applications that work together with other web applications for the principle of exchanging data. Web Services can change existing applications into Web-applications. Web Services utilize SOAP over HTTP protocol for the message passing; therefore you can make use of your existing low cost internet for implementing Web Services. This key reduces the amount of cost in contrast to prior solutions [14].subsequently to SOAP over HTTP; Web Services can be executed on other reliable transport mechanisms like FTP etc.

The Web Services uses XML-based SOAP messages to call up functions and transfer the data. To ensure the message-level security in SOAP messages the Organization for the Advancement of Structured Information Standards (OASIS) defined the WS-Security standard [3]. WS-Security employee's two primary standards: XML Encryption [4] and XML Signature. These standards propose an adaptable usage of security methods in SOAP messages and progresses the Web Services technology to provide integrity, confidentiality, and authenticity. The utilization of security mechanisms in SOAP messages supports attacks on a Web Services. Providing security in the soap messages is not mandatory so if the client is transferring secure information some authentication features can be included in the header. This is done in order to avoid much time consumption while transferring huge data. In WS-Security data integrity and confidentiality is ensured but it is done for the entire soap message so it becomes too complex.

IV. PROPOSED SOLUTION:

Security structure and pattern policies for heterogeneous platforms are relatively diverse. To accomplish the secure communication of Web service in heterogeneous platforms, a model called Prototype Of Secure Transmission(POST) is created in which a Diplomats add credential of both client and server side, which works as certification agency. These Diplomats issue certification to client and web server after verification, now client can process for request call to web service. Both the client and Web server sets their own security check modules called Diplomat credential of client and Diplomat credential of server to carry out secure usage to SOAP messages in the service communication, together with the signature and encryption of the SOAP message. The authentication section of client user is resided at Web server, and following the verification process from both Diplomats, client can call the Web service. This approach to the security interactions of Web services in heterogeneous platforms can be accomplished for secure transmission of SOAP messages.

V. IMPLEMENTATION

To achieve the secure communication of Web service in heterogeneous platforms, a model called Prototype Of Secure Transmission(POST) is created in which a Diplomats add credential of both client and server side, which works as certification agency. These Diplomats issue certification to client and web server after verification, now client can process for request call to web service. Both the client and Web server sets their own security check modules called Diplomat credential of client and Diplomat credential of server to carry out secure usage to SOAP messages in the service communication, together with the signature and encryption of the SOAP message. The authentication section of client user is resided at Web server, and following the verification process from both Diplomats, client can call the Web service. This Prototype of Secure Transmission (POST) is based on SOAP header together with signature, encryption and authentication. This approach to the security interactions of Web services in heterogeneous platforms can be accomplished for secure transmission of SOAP messages.

A. Components of POST:

1) UDDI Server:

The Universal Description, Discovery and Integration (UDDI) protocol is one of the major building blocks required for successful Web services. UDDI creates a standard interoperable platform that enables applications to quickly, easily, and dynamically find and use Web services over the Internet.

The UDDI Registry is designed to store information about Businesses and Services and it holds references to detailed documentation.

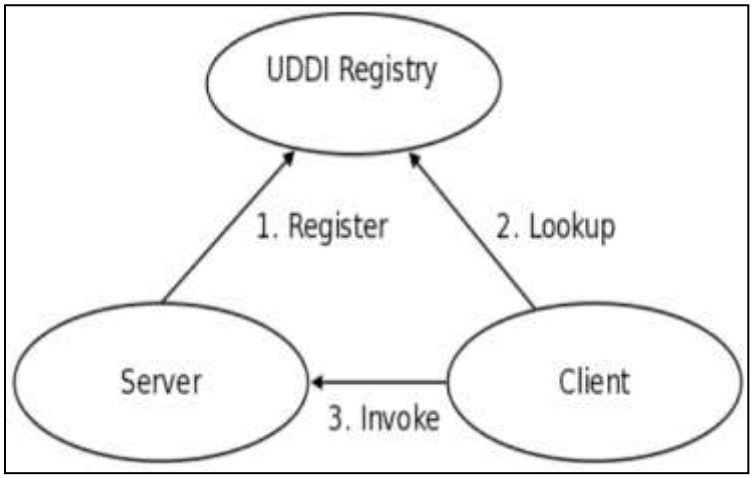


Fig. 4: Invocation Pattern using the UDDI Registry

B. WSDL Builder

A WSDL document describes a web service. WSDL is written in XML. It specifies the location of the service, and the methods of the service, using these major elements:

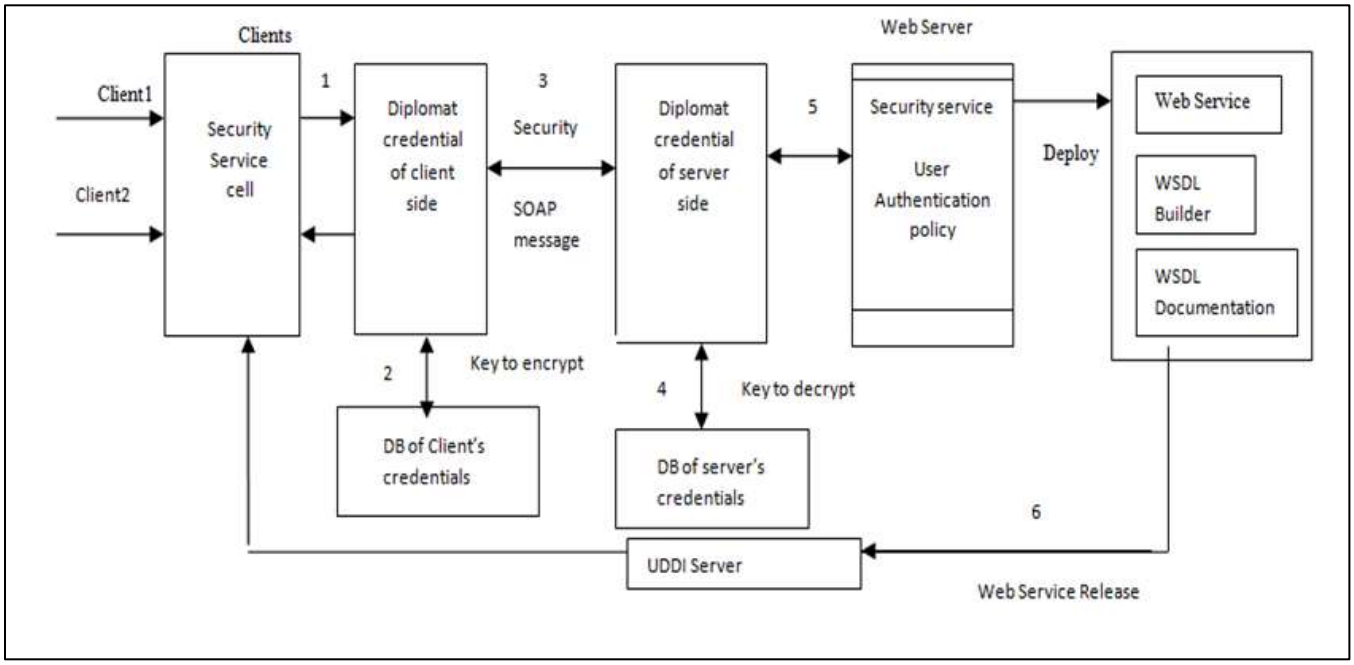


Fig. 5: POST Model.

Table – 1
Sample methods of WSDL

Element	*Description*
<types>	*Defines the (XML Schema) data types used by the web service*
<message>	*Defines the data elements for each operation*
<portType>	*Describes the operations that can be performed and the messages involved.*
<binding>	*Defines the protocol and data format for each port type*

C. Security Service Cell

It is responsible for the security of Web service during transmission and achieves the security requirements of the model including signature and encryption of the SOAP message.

D. User Authentication Policy

It is responsible for the request verification of client's identity, and only authenticated users can call the appropriate Web service.

E. Diplomat credential of client

Its mission is to maintain information about clients how are logging in to access data from web server.

F. Diplomat credential of server

Its task is to keep information about server side records accessed by clients thought web services

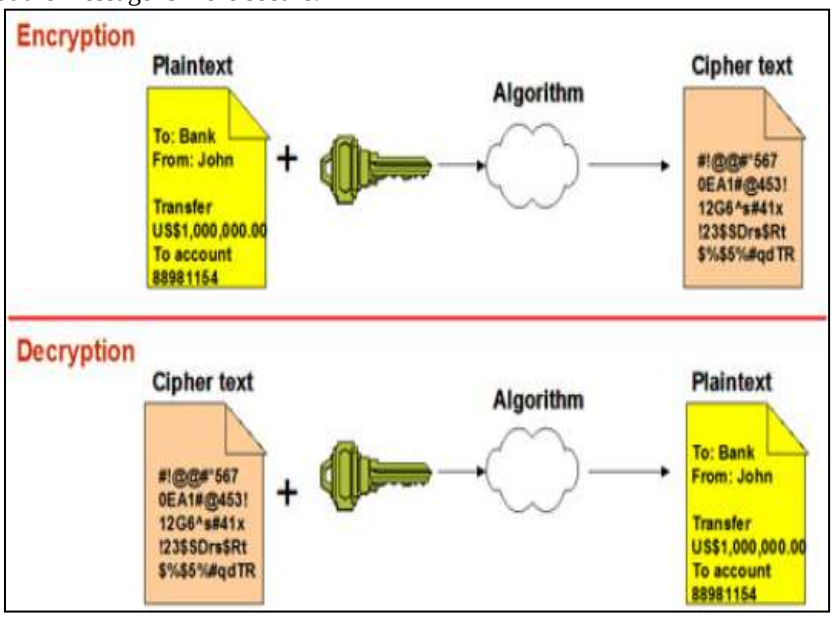
G. Message transmission among POST:

1) Step 1: Client's security cell sends clients information to Diplomat credential of client side
2) Step 2: All clients' information (such as user name, passwords etc) is encrypted and stored in database of client's credentials.
3) Step 3: Then client's information is compare at Diplomat credential of server side.
4) Step 4: if required information is agreeable them encrypted information at clients side is decrypted at database of server's credentials.
5) Step 5: a Security service provides user authentication and verification of encryption and decryption Keys.
6) Step 6: Web services are deployed from web server using WSDL Builder

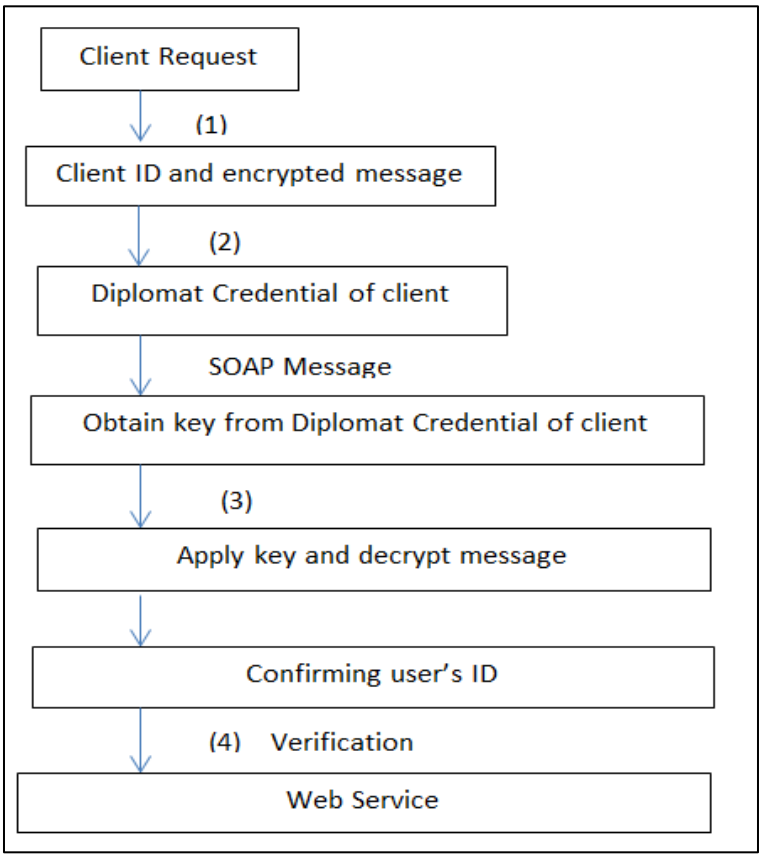
H. Encryption technique in POST model:

The security strategies of the POST model take in the encryption and signatures of the soap messages. In which the entire soap message is encrypted and defined security policy is considered then validation is done. On the server side decryption and

verification takes place. After applying the security policy on the server side the encrypted message specifies the certificate information so it states that the message is more secure.



I. User Authentication Policy:



The user authentication policy is shown in Figure 2.
Web server's user authentication process as follows:

1) Client request
2) Obtains the Key used to encrypt Message from the SOAP message through Diplomat credential of client.
3) Extract the client ID and Message from the SOAP message that client sends.
4) Use the Key to decrypt the Message and verifying the user's ID and allowing to use web service.

J. Instance of security communication:

The SOAP message format

```
<authentication>
<clientID>admin</clientID>
<password>admin111</password>
<serverID>server</serverID>
</ authentication>
```

Client Request SOAP Message

```
<ToClient>
<clientID>admin</clientID>
<Message>encrypted
message</Message>
</ ToClient>
```

SOAP message that security diplomats reverted to the client.

```
<ToClient>
<clientID>admin</clientID>
<Message>encrypted
message</Message>
</ ToClient>
```

SOAP message that security diplomats reverted to the server.

VI. CONCLUSIONS

Web services security policies in heterogeneous platforms have been proposed in this paper by creating a secure SOAP message transmission prototype.

Pooled with existing functional paradigm, user authentication during a Web service communication as well as the safe handling of SOAP messages in heterogeneous platforms is achieved. The security prototype offers approximate hold on the security interaction of Web services in heterogeneous platforms and is verified by experiments. This Prototype gives assurance to security interactions of Web service effectively on heterogeneous platforms.

REFERENCES

[1] F. Satoh, et al., "Adding Authentication to Model Driven Security," IEEE International Conference on Web Services (ICWS), Chicago, 2006, pp. 585-594. doi:10.1109/ ICWS.2006.25
[2] K. Bhargavan, C. Fournet, et al., "An Advisor for Web Services security Policies," Proceedings of the 2005 workshop on Secure web services, New York, 2005, pp.1-9. doi:10.1145/1103022.1103024
[3] S. Weber, P. Austel and M. McIntosh, "A Framework for Multi-Platform SOA Security Analyses," IEEE International Conference on Web Service, Salt Lake City, 2007,pp. 102-109.
[4] David Chappell,Tyler Jewell, Java Web Services, O'Reilly, March 2002, 28-50.
[5] Dongxi Zheng, Shaohua Tang,Shaofa Li, "XML Web Services Security Technology Overview", Computer Engineering and Application, 2004.7, 38-41.
[6] Doug Tidwell, James Snell, Pavel Kulchenko,Programming Web Services with SOAP, O'Reilly,December 2001, 39-61.
[7] J. Viega, "Why Applying Standards to Web Services is not Enough," IEEE Security and Privacy, Vol. 4, No. 4,2006, pp. 25-31.doi:10.1109/MSP.2006.110
[8] IBM Knowledge Center, Web Sphere Application Server Network Deployment 8.0.0, Web Services Security specification.
[9] Karthikeyan Bhargavan,"Secure Sessions for Web Services" To appear in ACM TISSEC, Vol. V, No. N, December 2006,
[10] VeriSign, Microsoft, webMethods. XML Key Management Specification. Jan. 2001
[11] "What is SAML? - A Word Definition from the Webopedia Computer Dictionary". Webopedia.com. Retrieved 2013-09-21.
[12] N. Bieberstein, S. Bose, M. Fiammante, K. Jones, R.Shah and Z. Ning, "Service-Oriented Architecture Guide," in Chinese, Posts & Telecom Press, Beijing, 2008,pp. 160-166.
[13] Jian Jin, Hong Zhang, Jiahua Liang, Hualin Qian,"Analysis of Web Services Security", Micro-electronics and Computer, 2004.3, No3, Vol 21, 19-24.
[14] G. H. Hwang, Y. H. Chang and T. K. Chang, "An Operational Model and Language Support for Securing Web Services", IEEE International Conference on Web Services (ICWS), 2007.