# A new Security Framework for Protecting WSDL File of Web Service

**Arezoo Mirtalebi † And Mohammad Reza Khayyambashi††**

Department of Computer, Faculty of Engineering, Esfahan University , Esfahan, Iran

**Summary**
Being regarded as the new paradigm for Internet communication, Web Services have introduced a large number of new standards and technologies. Though founding on decades of networking experience, Web Services are not more resistant to security attacks than other open network systems. Quite the opposite is true: Web Services are exposed to attacks well-known from common Internet protocols and additionally to new kinds of attacks targeting Web Services in particular[1].
The attacks on Web Services might cause halt of the entire network communication or expose confidential information in an organization; therefore, the inevitable challenge facing organizations today is to implement adequate Web Service security. In this paper, a new strategy to enhance Web Service security level is proposed and while the other presented solutions by now have focused on SOAP messages to defend Web Services attacks and increase their security level, in this paper to the best of our knowledge, for the first time this problem has been analyzed from a new aspect and that is securing WSDL file . This Solution is suitable for Web services which have critical rules according to their policies and their WSDL faced with hacking problems.

***Key words:***
*XML Encryption, Web service Security, WSDL, WSDL attacks.*

## 1. Introduction

A Web Service is a software system conceived to support interoperable machine-to-machine interaction over a network. Put in another way, Web Services provide a framework for system integration, independent of programming language and operating system. Hiding implementation detail, these services are loosely coupled and can be discovered which make them so flexible to use[2, 3].
Hackers and other malicious users are getting smarter about Web services. They are uncovering new techniques at the SOAP [1]/XML data-level that bypass HTML and target weaknesses in Web service programming, technology and architecture. Among other things, a hacker can undermine security and penetrate systems by relying on the expressiveness of SOAP and WSDL[2] to develop multiple attack patterns to achieve his/her goal. Web server security must now complement Web services security in order to provide an adequate defense against these new types of attacks [3].
Web service technologies provide us with a new way of unifying disparate systems into an openly interoperable environment. The XML-based characteristic of Web service makes it available via the web no matter how different their background systems are between the service provider and service consumer [4]. This characteristic made it as XML-based technology. On the other hand the foundation of Web service technology is on XML because the main correlated components like WSDL, BPEL[3] and SOAP are all derivatives of XML languages. WSDL acts as Web services interface which describes all about Web service is in XML format. SOAP as a pattern for messages which are transferred between Web services to make communication between service provider and service consumer also is handled as XML files. BPEL as an executable language for specifying actions within business processes with Web services are serialized in XML. Hence, the security of Web services is significantly involved with the security of XML.
XML Signature and XML Encryption are main parts of XML security standards which have issued by World Wide Web Consortium (W3C) in 2002 [5, 6]. Many of presented solution by now have used these standards to secure Web service.
SOAP is secured by WS-Security is a standard that states a strategy and specifications to bring different security technologies together [7]. The WS-Security specification provides how XML Digital Signatures and XML Encryption may be used with SOAP messages.
Concurrent with this study, all presented approaches in security of Web services focused on securing SOAP messages, but another important danger hole which threats Web service security is WSDL attack, hitherto not discussed in Web service security. Securing the WSDL document is not as simple as security of SOAP messages. Requesters find the list of functions of Web service and their parameter by parsing WSDL. Hackers can attack to Web service by abusing this information.
This paper suggests a solution to repel to hackers who want to attack to Web service through this hole. It aims to offer a

[1] Simple Object Access Protocol
[2] Web Service Description Language
[3] Business Process Execution Language

Manuscript received September 5, 2012
Manuscript revised September 20, 2012

security level on WSDL by using the XML security standards in order to handle this security problem. The rest of this paper is organized as follows. Section2 provides a brief overview of XML and Web services security. Afterwards, top ten threats of Web service have been introduced in section 3. Section 4 discusses more details about WSDL threats and after that, section 5 addresses an approach which tries to repel WSDL attacks. Section 6 concludes this paper.

## 2. Web Service Security

The foundation of Web service is XML. So prior to dealing with web service security, the XML security has to be discussed. This section explains the XML structure as an international pattern, then focuses on its security and correlates it to security of Web service.

### 2.1 XML

Extensible Markup Language (XML) is a simple data description language. It was standardized by W3C to bring a way of markup-based publishing to the Web [8]. The design goals of XML emphasize simplicity, generality, and platform independency [4]. Originally designed to meet the challenges of large-scale electronic publishing, XML is also playing an increasingly important role in the exchange of a wide variety of data on the Web. It can encode a hierarchical set of information using verbose tags [9].
XML is used in Web service foundation because it provides adequate flexibility and simplicity in WSDL and SOAP. It can provide platform independency, too. XML Web services, As a piece of technology, are a set of protocols built on the global connectivity made possible by SOAP, XML and HTTP [4].

### 2.2 Xml advantages and disadvantages

The World Wide Web Consortium (W3C) developed XML as a general-purpose markup language used to create special-purpose markup languages, or languages used to describe data. Simply put, XML is a way of describing data, Regardless of formatting, in an application-neutral way. XML works well with HTML in that both content and meaning (via XML) and formatting (via HTML) can be specified.
XML is very flexible; in many schemas the data fields are optional, or can appear in any order. Additionally, XML ignores formatting, so the data fields can be organized in a human-readable tree or simply mashed together. XML has been key to bringing together Communities of Interest (COIs). Using XML, COIs are able to discuss Common key terminology, algorithms and concepts without resorting to specific system languages or definitions. XML has been instrumental in the move towards SOA[4] and Web Services. Another significant advantage of XML is that it is well structured, which enables XML-aware firewalls to perform inspection of messages[6]. And as it was said before xml is a cross platform language so it does not need any special machine to interpret its tags.
Unfortunately, one can find as many faults with XML as advantages. XML is a very verbose language, where every data item requires an opening tag and a closing tag, often resulting in very large messages. Complex types comprised of multiple data items require tags on every contained data item, as well as the tags on the containing type. When a document includes many data items in large data structures, the overwhelming bulk of the document will be tags, instead of data. The larger files imply greater bandwidth requirements. XML also requires significant amounts of processing power to marshal the data in and out of the XML syntax. Programs need to use software components known as “parsers” to format data with XML tags, or to read the XML-formatted document. The more data an XML file contains, the more tags are required, the greater the processing power and time needed to parse the document. XML’s greatest strengths, its flexibility and complete application-neutrality, are also its Achilles’ heel from a security standpoint. The XML parsers are free to format or re-format data in any way they choose[6].

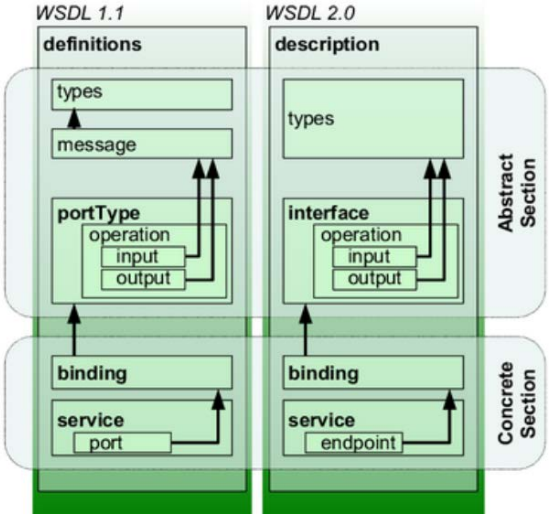


Fig. 1 WSDL1.1 versus WSDL2.0

### 2.3 Web Services threats through XML files

Because xml is easy to understand and interpret without any special machine, many vulnerabilities threat xml, for example malicious users can simply attack xml files read them, and simply change their formats in a way that they

[4] Service Oriented Architecture

prefer. So as far as Web Service is an XML-dependent technology both in message level for exchanging soap messages and interface level which is WSDL file as a unique way to know and interact with any Web Service, it always faces many xml threats via xml structure: soap messages and WSDL files. Many ways have been represented to secure soap messages but to the best of our knowledge no solution has been represented yet to secure WSDL files and, as it will be explained, attack to WSDL files is placed in the top level of importance in the category of threats and vulnerabilities of Web Services.

## 3. WSDL attacks

In order to illustrate the nature of the WSDL threats, it is worthwhile discussing the structure of WSDL documents

### 3.1 A Brief Overview of WSDL

WSDL is a W3C consortium recommendation and has become the de-facto standard for defining Web service interfaces[17] . In SOA, WSDL is used to describe the interfaces of all services irrespective of the underlying technology. It is an advertising mechanism for Web services to dynamically describe the parameters used when connecting with specific methods. These files are often built automatically using utilities. These utilities, however, are designed to expose and describe all of the information available in a method[16].

In June 2007, W3C has published the WSDL 2.0 recommendation [18]. These two versions of WSDL structure have some differences in their XML tags as illustrated in figure3.

WSDL 1.1 describes the interface of a service within the <PortType> section, where the operations are listed. WSDL supports one-way and two-way (request/response) operations. Each operation is defined by input, output and optional fault messages. The input message represents the payload an operation receives in order to perform the processing, while the output message represents the resulting payload. The fault message is used to signal faults. Output and fault messages can only be used in request/response operations. Each message consists of several parts. Message parts are defined within the <message> section. Each message part is defined by the corresponding schema as a simple XML type, a complex type or an element. The schema can be found in the WSDL document under the <types> section. Schema can be embedded in the WSDL document, or can be imported from a standalone schema (XSD) document. The <binding> defines the mapping of the interface/payload to the corresponding protocol, such as SOAP, or any other supported protocol. The <service> defines the service endpoint location [17].

WSDL 2.0 has simplified the interface description. The interface is defined within the <interface> section. It has dropped the concept of input and output messages. Rather the operations directly specify XML elements for input and output. For faults, fault messages can be specified within the <interface> section. This allows the same fault message to be used used in different operations, which simplifies fault handling. WSDL 2.0 has introduced the message exchange patterns, which specify the sequence in which the associated messages are to be transmitted between the service and the client. It also allows interface inheritance. The <binding> and <service> parts have not changed significantly apart from the new syntax. It has retained support for literal style of Web services only [18].

### 3.2 WSDL Threats

WSDL documents are the handbook to a company's Web services, and as such, they contain the recipe for interaction that yields improved communication between organizations. WSDL documents are also the handbook for attacking and hacking these same Web services. Since a WSDL document contains explicit instructions on how to communicate with previously private applications, they can cause a serious security breach if the Web services are compromised. Since the WSDL document includes all of the operations that are available to the consumer, it is straightforward for a hacker to run through all of the operations with different message request patterns until a breach is identified. This "knocking on every door until one opens" approach is usually effective when poor programming practices are employed or simply the result of operations that were excluded from published WSDL documents yet are still up and running for some reason[14].

In WSDL scanning threat, attackers may reveal sensitive information like types, messages, operations port types, bindings, and guessing other methods.

In parameter tampering, attackers tamper parameters within a WSDL document in order to retrieve unauthorized information. Since structures on how to use parameters are explicitly described within a WSDL document, malicious users can play around with different parameters to access confidential information [13].

### 3.3 Relation between XML Security and Web service Security

#### 3.3.1 Web Services Security Stack

By now, many standards have been developed to resolve Web Services security problems, the open standards communities that created Web Services developed a number of security standards for Web Services. Figure 2 illustrates a notional reference model for Web Services security standards. This reference model maps the different standards to the different functional layers of a typical Web

Service implementation. These layers are modeled after the OSI Reference Model but are not intended to be interpreted as strictly hierarchical[12].

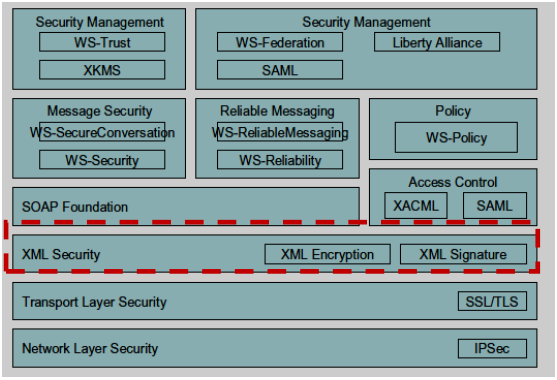


Fig. 1 Web Service Stack

As it is shown in Figure 2, in xml security layer there are two p's including: XML Encryption and XML signature, which are standardized by W3C[13].
Because WSDL document is an xml file it seems better to have an overview of these two standards.

3.3.2 XML Signature
An XML signature is a digital signature obtained by applying a digital signature operation to arbitrary data. Digital signatures are an important element in electronic security because they can be used to ensure the integrity, authenticity, and non-reputability of data. However, while the existing technologies allow us to sign only a whole XML document, XML signature provides a means to sign a portion of a document. This functionality is very important in a distributed multi party environment, where the necessity to sign only a portion of a document arises whenever changes and additions to the document are required. For instance consider a patient file, in which different physicians have stored different information about their diagnoses, so there should be possibility that each doctor can sign her own part and can access her part by a unique digital signature. The extensible nature of XML also allows support for multiple signatures inside the same document [14, 15].
In summary xml signature acts as a filter for recipient of a message to become sure that the received message has not been tampered and what he has received is the exact copy of the original form. In figure 3 xml signature steps both on sender and recipient sides are illustrated.
On sender side the data to be signed are first digested (a digest is a fixed-length representation of a resource and is created using, for example, a hash function such as (SHA-1) and the resulting value is placed in an element, called Digest Value, together with other information[14].
In a digital signature process, a pair of keys called public and private key are used to complete the steps of signing a message. These two keys can be generated by different mathematic algorithms via a key generator server. These two keys are unique for each other. It means that if a message is locked by a private key it can be unlocked only by the corresponding public key and vice versa. A private key is a key which is known only for his owner and no one else, while the related public key is known for everyone.

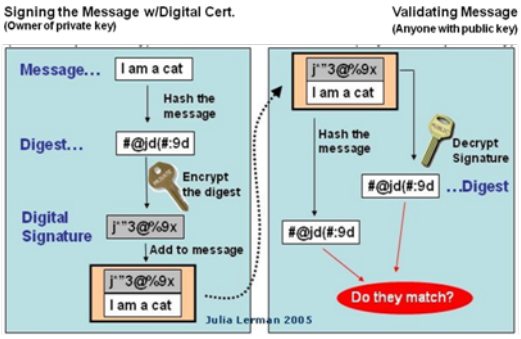


Fig. 3 XML signature Process

On the sender side after operating the hash function on the message, the digital signature is produced by using one of the encrypting algorithms which needs a key to perform, therefore the sender as it is shown in Figure 3 will use his own private key to operate the algorithm on the digested message, making the digital signature, adding the signature to the original message as a verification filter and finally send the package to the receiver. On the other side when the recipient receives this encrypted digest along with the original data, thus it can reproduce the digest of the received data and decrypt the encrypted data with the signer's public key. If these two values match, the signature is validated. Hence, the transmitted document is integral and authentic. The fallowing XML signature structure shows the structure of XML signature and its key elements. [11, 12] for more information about XML signature elements you can see [16].

```
<Signature>
 <SignedInfo>
  <SignatureMethod />
  <CanonicalizationMethod />
  <Reference>
    <Transforms>
    <DigestMethod>
    <DigestValue>
  </Reference>
  <Reference /> etc.
 </SignedInfo>
 <SignatureValue />
 <KeyInfo />
 <Object />
</Signature>
```

*Example4:XML Signature Structure*

3.3.3 Xml encryption

Secure XML document transmission over the Internet requires mechanisms and techniques for encrypting their contents. Many well established encryption techniques are currently in use for text documents. However, the main drawback of these techniques is that they have been designed for encrypting a whole document and thus they do not support selective encryption of a document. XML encryption can be used to encrypt arbitrary data. As for XML signature, the main advantage given by XML encryption is that it supports the encryption of specific portions of an XML document rather than the complete document[14, 17].The two primary encryption methods in existence today are:

• **Symmetric encryption**, also known as secret key cryptography, which requires the sender and receiver of a message to share the use of a single, common key for encryption and decryption. This pattern of encryption is illustrated in Figure 4.

• **Asymmetric encryption**, also known as public key cryptography, which employs two keys: a public key to encrypt messages and a private key to decrypt them. Figure 5 shows this encryption method.[18]

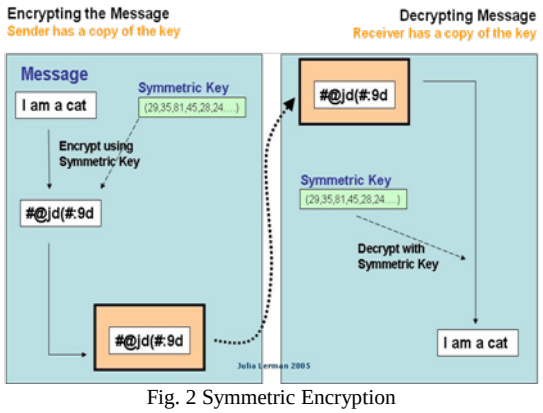


Fig. 2 Symmetric Encryption

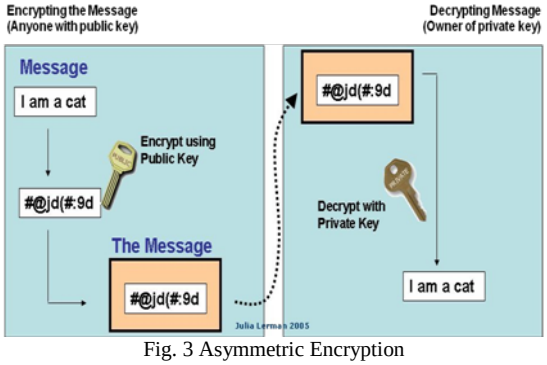


Fig. 3 Asymmetric Encryption

The study shows that symmetric encryption due to the use of a shared key both at sender and reciever side is much faster than asymmetric encryption but the most important problem which is explored about this pattern is its requirement to a secure channel which is need to transfer the shared secret key while the asymmetric pattern because of using two different keys at sender and reciepant side does not needed this secure channel on the other hand, asymmetric encryption method requires an infrustructure to manage public and private keys like PKI5 which is usually too expensive for organizations to implement. On the other side for the secure cahnnel problem of the latter pattern, some soloutions have been offered that can solve this issue to some extent.the main structure of xml encryption has been shown by the follwing XML encryption structure[3, 16, 19]:

```
<EncryptedDataId? Type? MimeType? Encoding?>
        <EncryptionMethod/>?
        <ds:KeyInfo>
           <EncryptedKey>?
           <AgreementMethod>?
           <ds:KeyName>?
           <ds:RetrievalMethod>?
           <ds:*>?
        </ds:KeyInfo>?
        <CipherData>
              <CipherValue>?
              <CipherReferenceURI?>?
        </CipherData>
        <EncryptionProperties>?
</EncryptedData>
```

*Example5:XML encryption structure*

In the follow example remember the customer credit xml file, which was presented in part 2, has been encrypted.

```
 <?xml version='1.0'?>
<PaymentInfoxmlns='http://example.org/paymentv2'>
          <Name>John Smith<Name/>
          <CreditCardLimit='5,000' Currency='USD'>
           <Number>

 <EncryptedDataxmlns='http://www.w3.org/2001/04/xmlenc#'
          Type='http://www.w3.org/2001/04/xmlenc#Content'>
       <CipherData>
       <CipherValue>A23B45C56</CipherValue> –
       </CipherData>
    </EncryptedData>
    </Number>
    <Issuer>Bank of the Internet</Issuer>
    <Expiration>04/02</Expiration>
 </CreditCard>
</PaymentInfo>
```

Now no one can decrypt this file expect the one who has the corresponding key.

*Example6: encrypted xml file sample*

[5] Public key infrastructure

3.3.3 Security Solution to Defend WSDL Attack

This section would like to present a solution which attempts to defend WSDL attacks. By the time of writing this study, many approaches have been presented to increase security level of Web Services, but by reviewing these methods, it becomes clear that most of these methods are trying to secure SOAP messages, and to the best of our knowledge there has not been any practical method to secure WSDL file. As it was analyzed in the section 6, WSDL documents are essentially advertisements of the functions that can be invoked by consumers, along with the expected results. WSDL file determines all functionality attribute of Web service. Everybody can access WSDL to find needed information for calling an operation of Web service. Therefore, it acts as a guidebook for hackers to use it and halt a Web service.

Enhancing Security of WSDL is provided by applying a security level on WSDL documents to protect it against malicious access. It is a good suggestion for Web services which have critical rules, so an attack to them can cause a crash in the system (e.g. management service). Notice that it's not suitable for all common Web services, because the new security level places a limitation on accessing the WSDL. A requester who wants to use WSDL should follow a scenario up in order to obtain needed information, so the access time is increased.

Due to the fact that the foundation of WSDL document is XML, so XML Security standards will be used in our solution to enhance the security of WSDL documents, too. Web service security standards, explained in section C, are use for securing soap messages. Although both WSDL and Soap messages build of XML, hitherto Security approaches of soap messages don't be used for securing WSDL.

By encrypting the WSDL document, it changes to an meaningless document. The encrypted text can be apprehensible for customers who have the respective key of encryption. They use the key to decrypt the WSDL and gain necessary information. Just authenticated users have the decrypting key.

There is no need to protect overwhelm sections in WSDL document. Therefore, only the sections which hackers use them to understand total structure of Web service should be encrypted. In WSDL1.1 the most important parts are <Messages> and <PortType> section, in which Web service's methods and method signatures are described. At the same, in WSDL2.0 <Interface> section has this critical role. If security of these sections is provided, the security of WSDL document will be enhanced against hackers. Hackers can simply use these two elements to understand the total structure of a Web Service and disturb the organization performance in which the Web Service is working. In our approach it is believed that if security of these two parts are provided, the security of WSDL file will be provide too and so as it is not needed to hide all information of a WSDL, XML encryption is used to encrypt just these two elements, so no one can decrypt these parts until one can obtain the respective key of encryption. It is essential to mention again that not all the WSDL files are proposed to be encrypted, since WSDL as expressed in the name, should describe all functionality of a Web Service, but this security approach can be applied only on the WSDL of Web Services which are faced with hacking problems and in other words are positioned in critical security problems. Figure 6 shows the steps of this security approach.

The proposed approach is adoptable with WSDL versioning to apply change management which introduced in [19].

Step 1) first the Web Service generator who has the original WSDL file sends his encryption request to Secure_ WSDL Web Service.

Step2) the Secure_ WSDL Web Service gets the WSDL file and will do two things on the received original WSDL document (step 3 and 4)

Step 3) Signing process: due to the fact that we needed a strategy in order for the WSDL file receiver to make sure that he/she has received a copy of the original one. So in this step the Secure_ WSDL Web Service will sign the WSDL document with his own private key to make authentication possible for the other side.

Step4) Encryption process: Secure_ WSDL Web Service will encrypt the message and PortType elements of WSDL file. To fulfill the encrypting process, one can use either symmetric or asymmetric encryption. To do symmetric encryption the sender does not need to obtain recipient public key to do the encryption or in other words he does not need to know who the receiver of the WSDL file is. Because he has to obtain the receiver's public key by implying a type of key management infrastructure, so due to the fact that using asymmetric encryption is so expensive for both receiver and sender side the symmetric encryption seems to be more proper from many aspects. As it was said before if we use the symmetric encryption a secure channel will be required to transfer shared key between two sides. When the WSDL file was encrypted it will be sent through internet to recipient. When the recipient received the encrypted file again he will do two jobs on encrypted document (step 5, step 6) in order to obtain original WSDL file.

Step5) Decryption Process: in this step first the receiver uses the respective key to decrypt the encrypted file. If the symmetric encryption has been applied on the encrypted file, the receiver will use his own private key to decrypt the document, or else if the symmetric encryption is used the receiver will use their shared key to decrypt the encrypted document.

Step 6) Sign Decryption Process: at the end, the recipient will use his own private key to check the signature of Secure_ WSDL Web Service and become sure that the received encrypted message has been sent through a valid user not a malicious one and also it has not been tampered in the way. And then he can have the original WSDL file.

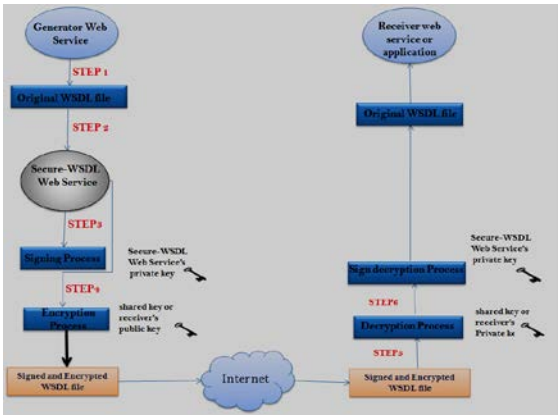


Fig. 4 proposed security solution steps

The proposed solution is applicable on any version of WSDL file because all WSDL files versions are in XML format this solution also can use WSDL versioning to apply change management.

## 4. Conclusion

Security of WSDL is a challenging area in Web service security which has not been discussed in previous standards. Web service security architecture brings different security technologies together and offers a new standard as WS-Security which focuses on security of SOAP. The WS-Security specification provides how XML Digital Signatures and XML Encryption may be used with SOAP messages. Another most important issue that cause security problem for Web service is their WSDL file. Since a WSDL document contains explicit instructions on how to communicate private application, they can cause a serious security breach if the Web services are compromised. This paper presents a solution to secure WSDL document. A brief introduction to XML and Web services security standards and how they work together had been proposed. Then, Web Services threats were reviewed and after that focused on WSDL attacks and at last a new approach for securing WSDL by use of XML Security standards had been suggested. This solution encrypts WSDL by digital encryption in order to deploy a security level on it and just promises to authenticated users to decrypt it.

## Refrences

[1] S. Gaithersburg, "Web Services Security: Challenges and Techniques," IEEE International Workshop on Policies for Distributed Systems and Networks, vol. 7, pp. 282-288, June 2007.

[2] Robert Warschofsky, Michael Menzel, and C. Meinel, "Transformation and Aggregation ofWeb Service Security Requirements," IEEE Computer Society, pp. 43-50, 2010.

[3] R. Kanneganti and P. Chodavarapu, "SOA Security," 1 ed: Manning, 2008.

[4] Y. Liu, T. H. Yeap, and W. O'Brien, "Securing XML Web Services with Elliptic Curve Cryptography," in Canadian Conference on Electrical and Computer Engineering (CCECE 2007), Canada, 2007, pp. 974-977.

[5] D. Eastlake and J. Reagle, "XML Encryption Syntax and Processing," w3C Recommendition, 2002.

[6] D. Eastlake, J. Reagle, and D. Solo, "XML Signature Syntax and Processing (Second Edition)," w3C Recommendition, 2008.

[7] A. Nadalin, C. Kaler, P. Hallam-Baker, and R. Monzillo, "Web Services Security: SOAP Message Security 1.1 (WS-Security 2004) ": OASIS Standard, March 2004.

[8] T.Bray, J. Paoli, M.Sperberg, and E.Maler, " E. Extensible Markup Language (XML) 1.1 (Second Edition)," World Wide Web Consortium (W3C), Cambridge, MA, USA, 2000.

[9] F. T. Ammari and J. Lu, "Advanced XML Security Framework for Building Secure XML Management System(SXMS)," Seventh International Conference on Information Technology: New Generations,IEEE Computer Society Washington, DC, USA pp. 120-125, 2010.

[10] L. Sun and Y. Li, "XML and web services security," IEEE International Conference on Computer Supported Cooperative Work in Design. CSCWD, vol. 12, pp. 765-770, 16-18 April 2008.

[11] N. A. Nordbotten, "XML and Web Services Security Standards," IEEE Communication Survey & Tutorials, vol. 11, pp. 4-21, 2009.

[12] B. schneier, "Applied Cryptography ,Protocols, Algorithms, And Source Code In C(Second Edition)," 1996.

[13] P Lindstrom, "Attacking And Defeninig Web Services," Spire Security Research Report January 2003.

[14] "Anatomy of a Web Services Attack," Forum Systems, Inc, 2004.

[15] E.Moradian and A. Hakkanson, "Possible attacks on XML Eb Servicea," IJCSNS International journal of Computer Sience and Network Security, january 2006.

[16] S. S. kumar and A. Benameur, "A Formal Solution to Rewriting Attacks on SOAP Messages," Dans ACM International Workshop on Secure Web Services, pp. 53-60, 2008.

[17] Web Services Description Language (WSDL) 1.1: W3C, 2001.

[18] Web Services Description Language (WSDL) Version 2.0: W3C, 2007.

[19] M. B. Juric, A. Sasa, B. Brumen, and I. Rozman, "WSDL and UDDI extensions for version support in web services," Elsevier at The Journal of Systems and Software vol. 82, pp. 1326–1343, 2009.

**Arezoo Mirtalebi** is a student of master of science in university of Esfahan. She has dedicated the last five years to the field of SOA. Arezoo has worked on a number of IT projects for BASA company and has specialized knowledge and experience in the technical, business and organizational aspects of SOA.

**MohammadReza Khayyambashi**, Ph.D., is Associate Professor at the University of Esfahan.His research interests include engineering management, FaultTolerane, SOA, and auction quality.