International Journal of Research
Available at https://edupediapublications.org/journals
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 05 Issue 12
April 2018

A Review on Web Services and Security Issues

Pardeep Nehra
Department of Computer Science
E-Mail:- par.nehra82@yahoo.com

ABSTRACT: *In this paper we are discussing that data mining and scoring tool providers require users to use provider-specific ways to invoke their services. The provider-specific approach could be a major factor affecting why data mining tools and applications are not currently as widespread as one might hope. Web services standards can address these proprietary issues. This paper discusses what web services are, in general, as well as in the context of data mining and scoring. One not-so-rigorous description of web services is as follows: A web service client passes a request in text while the service provider acts on the request and returns text to the client, all via the Web. Web services are identical in concept to this process. However, complicated web services often involve richer content as input than simple web page browsing with web services. XML is most often used to format the input. As to the output, the contrast between web browsing and web services is not about whether or not the content is complicated, but rather whether the format is HTML or not. Even though it is not entirely technically correct, one can view an HTML document as an instance of an XML document. However, HTML is particularly designed for web browser consumption, while an XML document is designed for a specific business need. It wouldn't be complete to describe web services without mentioning the SOAP protocol. Keep the following notes in mind if you are new to SOAP: SOAP is not really a simple protocol and "object" has nothing to do with the protocol. The Worldwide Web is based on the HTTP protocol. Currently the SOAP protocol fits nicely on top of the HTTP protocol.*

KEYWORD: XML, HTTP, SOAP, Web Services, WS-1, OASIS.

INTRODUCTION:

The IT industry has been talking about Web services for almost four to six years. Web services allows applications (e.g. automated business transactions, stock trading and order-tracking systems) to communicate with each other within organizations, across enterprises, and across the Internet in a loosely-coupled, platform- and programming language - independent manner. Several key standards have formed the foundation for Web services: XML (Extensible Markup Language), WSDL (Web Services Definition Language), SOAP (Simple Object Access Protocol), and UDDI (Universal Description, Discovery, and Integration). Since, the key benefit of Web services is to deliver integrated & interoperable solutions, ensuring the integrity, confidentiality & security is the most important key area that

needs to be addressed for Web services. Traditionally, the barriers of integration are due to the tight- coupling, where one application that calls another one is tied strongly by the function and the parameters. There is low flexibility or adaptability to changing environments or needs, due to:

1. Different programming languages
2. Different operating systems or hardware platforms
3. Different software vendors & in-house coding
4. It's difficult to integrate these systems internally
5. It's even harder to integrate with external business partners

WHAT ARE WEB SERVICES:

According to W3C, a Web service is defined as: "A Web service is a software system designed to support interoperable machine -to- machine interaction over a network. It has an interface described in a machine-process able format (specifically WSDL). Other systems interact with the Web service in a manner prescribed by its description using SOAP messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards." In substance, Web services are technology that allows applications to communicate with each other in a platform- , hardware- and programming language-independent manner. It uses XML based protocols to describe a collection of operations that can be accessed, executed or data exchanged over the network. A group of Web services interacting together in this manner defines a particular Web service application in a Service-Oriented Architecture (SOA).

WEB SERVICES EXHIBIT THE FOLLOWING DEFINITIVE CHARACTERISTICS:

1. Web services communicate using platform-, hardware- independent and language- neutral Web protocols. These Web protocols ensure easy integration over the network & loosely coupling between applications.
2. A Web service provides an interface that can be called from another program. This application-to- application programming interface can be invoked from any type of application client or service.
3. A Web service is registered and can be located through a Web Service Registry. The registry enables service consumers to find services that match their needs.

USES OF WEB SERVICES:

What can I do with Web services? While Web services provide all the advantages stated above, Web services allow us to implement as:

1. A credit card service that processes credit card

transactions for a given account number.

2. A market data service that provides stock market data associated with a specified stock symbol
3. An airline service that provides flight schedule, availability, and reservation functionalities.

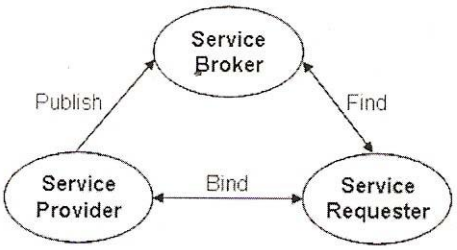
WEB SERVICES STANDARDS:

Web services are widely adopted standards such as HTTP and extensible Markup Language (XML). A few of the major Web services standards groups are listed below:

- **W3C (World Wide Web Consortium)** - The driving force behind the largest number of highly adopted standards in the Web services space including some Web building blocks such as HTML.
- **OASIS** - Source of the original specification from which XML evolved, as well as the home of the current XML and Universal Description, Discovery and Integration (UDDI) specification.
- **WS- I (Web Services Interoperability Organization)** - Acts as a watchdog group to ensure interoperability between implementations of Web services standards.

WEB SERVICES MODEL:

A



typical Web services model consists of three entities:

l. Service providers who create Web services and publish them to the outside world by registering the services with service brokers.

2. Service brokers who maintain a registry of Published services.

3. Service requesters who find required services by searching the service broker's registry.

IMPORTANCE OF THE SECURITY IN WEB SERVICES:

In February / March 2003 CBDI Forum carried out a survey of its subscribers who had practical experience in implementing Web services to understand how they were applying Web services, their motivation for adoption, and their experience to date as well as their detailed as well as their further plans for 2003. Security is important for any distributed computing environment. But, security is even more important for Web services due to the following reasons:

1. The boundary of interaction between communicating partners is expected to expand from intranets to the Internet. Obviously, security problem is much critical in Internet

because Internet communication is much less protected than intranet communication.

2. There will have more anonymous to access the web services since communicating partners are more likely to interact with each other without establishing a business or human relationship first. This means that all security requirements such as authentication, access control, non- repudiation, data integrity, and security must be addressed by the underlying security technology.

3. More and more interactions are expected to occur from programs to programs rather than from humans to programs. Therefore, the interaction between communicating partners using Web services is anticipated to be more dynamic and instantaneous.

SECURITY CONSIDERATIONS:

Security is about protecting assets. In the Web services context data and computational services are assets under consideration. The following security considerations must be considered as part of a comprehensive security framework:

1. Identification - The party accessing the resource is able to identify itself to the system.
2. Authentication - the proven identification of users in a computer system.
3. Authorization - There exists a set of transactions the authenticated party is allowed to perform.
4. Integrity - the prevention of unauthorized modification of data.
5. Confidentiality - the prevention of unauthorized disclosure of data.
6. Accountability - the provision of activity logs recording all user activity.
7. Non-repudiation - Both parties are able to provide legal proof to a third party that the sender did send the information, and the receiver received the identical information.

WEB SERVICES SECURITY SCHEMES:

Web services security language can be defined into two types: computer security and communications security.

1. Computer security is a node-oriented security focus and it is essentially access control within a computer system. A permission rule expresses restrictions on the usage at the server side and a client can execute the operations only if the permission rule is allowed.

2. Communications security is a connection-oriented security focus and it is about providing a secure logical connection between two agents. A requirements rule expresses the necessary security-relevant preparations for the use of a service, or security measures needed after the service execution. The activities authenticates and encrypts are associated with authentication and confidentiality, respectively. And currently, the most common security scheme available for today's Web service is SSL (Secure Socket Layer), which is typically used with HTTP. It provides authentication, confidentiality, and message integrity. However, SSL is designed to

provide point- to-point security, which falls short for Web services because:

1. We need end - to-end security, where multiple intermediary nodes could exist between the two endpoints.
2. SSL secures communication at transport level rather than at message level. As a result, messages are protected only while in transit on the wire.
3. HTTPS in its current form does not support non-repudiation well. Non-repudiation is critical for business Web services.
4. SSL does not provide element-wise signing and encryption.

In order to complement SSL, the technology industry has been working on various XML-based security schemes to provide comprehensive and unified security schemes for Web services. These schemes include:

1. XML digital signature - XML digital signature provides authentication, data integrity and non-repudiation. It is to develop XML syntax for representing digital signatures over any data type. The XML digital signature specification also defines procedures for computing and verifying such signatures. Another important area that XML digital signature addresses is the canonicalization of XML documents. Canonicalization enables the generation of the identical message digest and thus identical digital signatures for XML documents that are syntactically equivalent but different in appearance. XML digital signature provides a flexible means of signing and supports diverse sets of Internet transaction models.
2. XML encryption - Its goal is to develop XML syntax for representing encrypted data and to establish procedures for encrypting and decrypting such data. (Unlike SSL, with XML encryption, you can encrypt only the data that needs to be encrypted.)
3. XKMS (XML Key Management Specification) - XKMS consists of two parts: XKISS (XML Key Information Service Specification) and XKRSS (XML Key Registration Service Specification). XKISS defines a protocol for resolving or validating public keys contained in signed and encrypted XML documents, while XKRSS defines a protocol for public key registration, revocation, and recovery. The key aspect of XKMS is that it serves as a protocol specification between and XKMS client and an XKMS server in which the XKMS server provides trust services to its clients by performing various PKI operations.
4. XACML (Extensible Access Control markup Language) - Its goal is to standardize access control language in XML syntax.
5. SAML (Secure Assertion Markup Language) - It's to outline a standard XML framework for exchanging authentication and authorization information. As a framework, it deals with three things. First, it defines syntax and semantics of XML- encoded assertion messages. Second, it defines request and response protocols between

requesting and asserting parties for exchanging security information. Third, it defines rules for using assertions with standard transport and message frameworks.

6. WS- Security (Web Services Security) - It defines a set of SOAP header extensions for end-to- end SOAP messaging. Security. It supports message integrity and confidentiality by allowing communicating partners to exchange signed and encrypted messages in a Web services environment.

7. ebXML Message Service - The ebXML initiative is a set of next- generation XML-based standards enabling electronic business transactions via the Internet. One of the ebXML standards is ebXML Message Service, which defines how to securely and reliably send and receive SOAP messages. The SAML assertions can be digitally signed using XML digital signature. The same assertions can be encrypted using XML Encryption to ensure security. The public key used for digital signing and encryption can be validated and registered via XKMS. As for XACML, an SAML asserting party could use it to define an access control policy as a basis for handling SAML-based assertion requests.

Example: When a client placing an order, she uses XML digital signature and encryption to digitally sign and encrypt the purchase order XML document. She then sends the document to her supplier using SOAP, whose header structure is defined either in the WS- Security or ebXML Message Service standard. The document's receiver then could use XKMS to look up and validate the public key. Once the key is determined trustworthy, the receiver then validates and decrypts the purchase order. Finally, the receiver checks a policy server for authorization by sending and receiving SAML requests and responses. The policy server might maintain the access control policy information in XACML.

CONCLUSION: In this paper we discussed that web services are highly secure for using data. Because all web services converts all the data in XML format and it is well known that XML is platform independent. XML is the language of World Wide Consortium (w3c) and all the websites and all the web portals follows the rules and regulations of w3c. All these talking about the data so we can store our data in any format and on any platform we can easily get our data at any other platform.

REFERENCES:

[1] Java Specification Request 73: Java Data Mining (JDM), Version 1.0, Final Review.

[2] XML for Analysis Specification version 1.0.

[3] Predictive Model Markup Language, Version 2.1.2, http://www.dmg.org.

[4] OLE DB for Data Mining Specification, Version 1.0.

[5] SOAP Version 1.2, http://www.w3.org/TR/soap/.

[6] WS-Security, http://www-106.ibm.com/developerworks/webservices/library/ws-secure/.

[7] WS-Resource Framework, http://www.globus.org/wsrf/.

[8] XML Specification, http://www.w3.org/TR/2000/REC-xml-20001006.