ISSN(Online) : 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 11, November 2014

SoapUI and Soap Sonar Testing Tool Using Vulnerability Detection of Web Service

Ramakrishnan.R [1], Anbarasi.J[2], Kavitha.V[3]

Associate Professor, Dept. of MCA, Sri Manakula Vinayagar Engineering College, Pondicherry, India[1]

MCA Student, Sri Manakula Vinayagar Engineering College, Pondicherry, India[2]

MCA Student, Sri Manakula Vinayagar Engineering College, Pondicherry, India[3]

ABSTRACT: Web Services are modular software applications that can be described, published, located, and invoked across a network, such as the World Wide Web .Web services provide a simple interface between a provider and a consumer and are supported by a complex software infrastructure which is typically includes in the applications to server, the operating system and a set of external systems. In this technology is susceptible to the Cross-site Scripting (XSS) attack is takes advantage to existing vulnerabilities. In the proposed approach is using two Security Testing techniques they are Penetration Testing and Fault Injection, which emulate the XSS attack against to Web Services. This technology, combined with WSSecurity (WSS) and Security Tokens can identify their sender and to guarantee the legitimate access control to the SOAP messages are exchanged. So we use the vulnerability scanner soapUI that is one of the most recognized tools of Penetration Testing. In another way WSInject is a new fault injection tool can introduces faults in Web Services to analyze the behavior in an environment not in their robust. Therefore results shows their use of WSInject is comparison to soapUI can improve the detection in the vulnerability allows to emulate XSS attack and generates new types of them.

KEYWORDS: XSS, injection attack, java script, scripting ,WS-Security; WSS; Security Token; soapU TooI; soapsonar Tool.

I.INTRODUCTION

Web Services give rise to new security challenges. Cross-site Scripting Known as XSS is a type of Injection Attack that intercepts information provided by users. Its purpose is used to store, modify, or delete requests, misleading the servers and the user of the Web Services. The Web Services using Security Testing technique like Penetration Testing and Fault Injection. These techniques allow to verify: i) vulnerabilities in Web applications and services against different types of security attacks such as Denial-of-Services or spoofing attacks; and ii) discover new vulnerabilities before they are exploited by attackers. Both techniques use tools to analyse the presence of vulnerabilities in Web Services and emulate XSS attack.

In recent years, researchers have taken more interest in developing software-implemented fault injection tools. Software fault-injection techniques are attractive because they don't require expensive hardware. Furthermore, they can be used to target applications and operating systems, which is difficult to do with hardware fault injection. The injection of permanent faults using software methods either incurs a high overhead or is impossible, depending on the fault. We also analyse the robustness of Web services with WS-Security and SecurityTokens against the XSS attack. These specifications are allow to authorized the use of WebServices through the authentication of users and others services.

Partial control of the network and ability to capture the SOAP messages. Ability to intercept and modify strings or expressions, delay or replicate message traffic. Knowledge of the status of all participants, i.e. the attacker intercepts messages and supplants client/server or just works as a mediator of communication between the client and the server (man in the middle attack).The attacker can recognize the access points, operations and parameters of WSDL in the Web Service tested.

ISSN(Online) : 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 11, November 2014

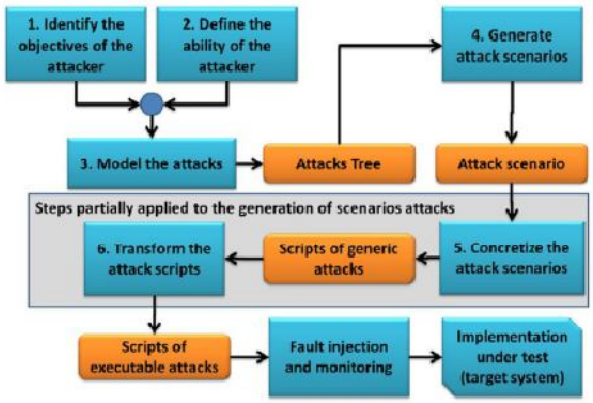

Fig. 1.**Security Testing Methodology**

- Partial control of the network and ability to capture the SOAP messages.
- Ability to intercept and modify strings or expressions, delay or replicate message traffic.
- Knowledge of the status of all participants, i.e. the attacker intercepts messages and supplants client/server or just works as a mediator of communication between the client and the server (man in the middle attack).
- The attacker can recognize the access points, operations and parameters of WSDL in the Web Service tested.

A. Existing system:

Web services enable to quickly integrate applications across multiple platforms and systems and even across businesses. Web services standards are SOAP, WSDL and UDDI will enable system-to-system communication that is easier and cheaper than ever before. Web services are distributed and open nature across internet and they are more susceptible to security risks while using. Many researches contribute WS-service testing tools to test web services to avoid vulnerabilities. in previous approaches uses some on testing tools to scan of vulnerabilities in web service. However, the variation in attacks which allows to inject scripts (e.g. JavaScript, VBScriptor Flash Script) in Web Services through its. infect every user who uses these Web Services .Due to difficulty to find vulnerabilities in Web Services like XSS, we apply a Security Testing Methodology in order to systematize the fault injection and remove vulnerabilities in web services

B. Proposed system:

The proposed approach makes use of two Security Testing techniques they are Penetration Testing and Fault Injection, to emulate XSS attack against the Web Services.WS-Security specifies how integrity and confidentiality can be enforced on messages and allows the communication of various security tokens such as SAML, Kerberos and X.509.The mainly focus is the use of XML Signature and XML Encryption to provide end-to-end security .The black box proposed approach, we used as information sources in the logs stored in tools (WS-Inject fault injector and soap-UI load testing) that contain the SOAP message (requests and response). One advantage of the proposed approach is that it relies on the use of a fault injector of general purpose, which can be used to emulate several types of attacks and may generate variants of the same, which is usually limited in the tools commonly used for security testing, as the vulnerabilities scanners.

The proposed steps in, composed of the following attributes:

- Attacker capability;
- Possibility of emulating the attack by a fault injection tool;
- The requirements of the attack to be run in the web service;
- The verification if the ws-security protects the web services from xss attack.

ISSN(Online) : 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 11, November 2014

II.LITERATURE SURVEY

1.WEB SERVICE SECURITY - VULNERABILITIES AND THREATS WITHIN THE CONTEXT OF WS-SECURITY.

A Web Service (WS) can be described a sanXML-based interface that can be used by a client application to invoke a computing service distributed in a network via standard Internet protocols. In order to WebServices to become a ubiquitous technique for program to program communication in place for how Web Services that utilizes the public Internet for transport can be properly protected and secured. As the situation appears today, most WebServices are not publicly exposed but are often deploye to inside the corporate, private network. This hampers the vision of WebServices that can be publicly published in directories which potential customers can search inorder to find a suitable service to satisfy their need. This report' outlines what the general vulnerabilities and threats are in deploying secure WebServices over publicly available. It then continues to give a brief overview of the most profound security standard, WS-Security, which is discussed in relation to the vulnerabilities and threats described previously. The conclusion of there port is that WS-Security is a promising security standardization effort, which can handle a number of security problems, specific for WebServices.

2. IMPROVING DATA PERTURBATION TESTING TECHNIQUES FOR WEB SERVICES

The wide use of service-oriented architectures (SOAs) and Web services is commercial software required a adoption of development techniques to ensuring the quality of Web services. Testing techniques and the tools is concern with quality and plays critical role in a accomplishing quality of SOA based systems. This paper presents new testing techniques can automatically generate a set of test cases and data for Web services. To support these techniques, a tool (GenAutoWS) was developed and applied to real problems.

3. FAULT INJECTION SPOT-CHECKS COMPUTER SYSTEM DEPENDABILITY

Computer-based systems is expected to more dependable. They have to operate correctly even in the presence of faults and fault tolerance was thoroughly tested the injection of faults both real and artificial. Users must start to request reports in manufactured on the outcomes of such experiments and the mechanisms built into systems to handle faults. To inject the artificial physical faults and then the fault injection offers a reasonably with Swift tools being preferred for most applications because of flexibility and the low cost.

4. AUTOMATICALLY TESTING WEB SERVICES CHOREOGRAPHY WITH ASSERTIONS

Web Service Choreography Description Language is a global view on the collaborations among the collection of services involving multiple participating or organizations. Since WS-CDL can aimed at a design specified for service composition and few approaches need to propose to testing WS-CDL programs. In this paper, we present the approach to testing WS-CDL program is automatically and the dynamic symbolic execution technique was applied to generating testing inputs and assertions are treated as the test oracles. Simulation engine for WS-CDL is used to perform the execution of WS-CDL programs during the process of symbolic execution.

III.CROSS-SITE SCRIPTING ATTACK

Cross-site Scripting (XSS) is one of the attack technique that involves attacker supplied code into a user's browser instance. A browser instance could be a standard web browser client(user),embedded in a software product such as the browser within WinAmp, an RSS reader, or an email client(user). The code itself is usually written in HTML/JavaScript Language , but may also extend to VBScript Language, ActiveX control, Java, Flash page, or any other browser-supported technology. When an attacker puts a user's browser to execute he/she code, the code will run within the security context (or zone) of the hosting web site. A Cross-site Scripted user would have he/she account hijacked (cookie theft) and their browser is redirected to another location of the page. Applications utilizing browser object instances which load content from the file system might execute code under the local machine zone allowing for system compromise.

It consist of Cross-site Scripting attacks:

- non-persistent
- persistent

ISSN(Online) : 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 11, November 2014

Non-persistent

Many web portals offer a personalized view of a web site and might greet a logged in user with "Welcome, <your username>". Sometimes the data referencing a logged in user is stored within the query string of a URL and echoed to the screen.

Portal URL example: http://portal.example/index.aspx?sessionid=459281&username=Kavi

In the example above we see that the username "Kavi" is stored in the URL browser. The resulting web page displays a "Welcome, Kavi" message. If an attacker are to modify the username field in the URL and inserting a cookie-stealing JavaScript. It could possible to gain control of the user's account if they managed to put the victim to visit their URL.

Persistent

Many web sites host bulletin boards where registered users might post messages are stored in a database of some kind. A registered user was commonly tracked using a session ID cookie authorizing to post. If an attacker were to post a message containing a specially crafted JavaScript, a user reading the message could having their cookies and their account compromised.

IV.PENETRATION TESTING OVERVIEW

A penetration test is one of the technique for scanner and authorized attempt to evaluate the security of an IT infrastructure by safely attempting to exploit system vulnerabilities including Operating System, services and application flaws is improper configurations and even risky end-user behavior. Such that the tool are also useful in validating the efficiency in defensive mechanisms is as well as end-users adherence to security policies.

Penetration tests are typically performed using manual or automated technologies to systematically compromise the other potential points of exposure. Once vulnerabilities had been successfully exploited on a particular system, testers may attempt to used the compromised system to launch subsequent exploits at other internal resources and specifically by trying to incrementally achieve higher levels of security clearance and deeper access to electronic assets and information via privilege escalation.

Information about any security vulnerabilities successfully through penetration testing is typically aggregated and presented to IT and network systems managers to help those professionals make strategic conclusions and prioritize related remediation efforts. The purpose of penetration testing is to measure the feasibility of systems or whether the web service is secure based on the vulnerability found in this techniques.

SoapUI is a tool developed by Eviware. SOAP stands for Simple Object Access Protocol. It is a communication protocol between user and service provide. It is a format for sending messages to user and browser. It is platform independent process. The tool injects scripts through the add-on Security Testing tools and analyses the response from servers and request to user. However classifying the responses in Web Services, vulnerable or not, by the injection of cross site scripting attack. It assists programmers in developing SOAP based web services. It consisting of

- Generating stubs of SOAP calls the operations declared in a Web service description language file;
- Send SOAP messages to the web service and display the outputs on the browser;
- Populate a data source and generate messages with data extracted from it (only in the commercial version; this feature requires a limited amount of domain knowledge and skill with the tool usage. Any choices, all, occurrences and other constructs are not automatically dealt with);
- Run batch test case used some partial information about coverage of the data value sets used in the operations.

As web service are new as compared to web applications, it's consist had secondary attack vector. However lack of concern or knowledge it is generally found that security measures implemented in a web service is worse than what is implemented in web applications. Which makes the web service a favourite attack vector and easy to penetrate as per the attacker's point of view.

ISSN(Online) : 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 11, November 2014

V. WEB SERVICES PENETRATION TESTING

Web application security is quite popular among the pen testers. Organizations developers and pen testers treat web applications as a primary attack vector. Web services relatively new as compared to web applications is considered as secondary attack vector. Due to the lack of concerns and knowledge is generally found that security measures implemented in a web service is worse than what is implemented in web applications. The web service had an attack vector and easy to penetrate as per the attacker's point of view. Another reason to write this article is that the use of web services increased in last couple of years in a major ratio and also the data which flows in web services are very sensitive. The use of web services increased suddenly because of mobile applications. Then we all know the growth of usage for mobile applications has increased rapidly, and most mobile applications use to sort the web service. It has relatively increased the use of web services? Due to the lack of security implementations and resources available on web services play a vital role making it a possible attacking vector.

Simple Object Access Protocol (SOAP):

Soap is XML-based protocol that applications interchange information over HTTP. Web services is using SOAP format to send XML requests. SOAP client send a SOAP message to the server. The server response back again to SOAP message along with the requested service. Web protocols are installed and available for major operating system platforms. SOAP specifies exactly how to encoded an HTTP header and an XML file so that a program in one computer can call a program in another computer and pass it information. It also specifying the called program can return a response.

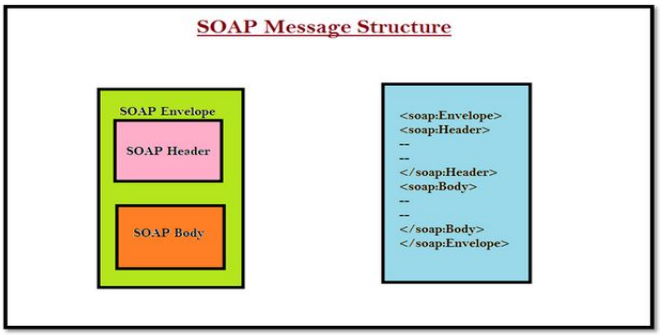

Fig. 2.SOAP Structure.

A typical SOAP request looks like

```
POST /ws/ws.asmx HTTP/1.1
Host: www.example.com
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://www.example.com/ws/IsValidUser"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://w
<soap:Body>
<IsValidUser xmlns="http://www.example.com/ws/">
<UserId>string</UserId>
</IsValidUser>
</soap:Body>
</soap:Envelope>
```

Fig. 3.SOAP request

If the service consumer sends a proper SOAP request then the service provider will send an appropriate SOAP response. A typical SOAP response looks

ISSN(Online) : 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 11, November 2014

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://ww
<soap:Body>
<IsValidUserResponse xmlns="http://www.example.com/ws/">
<IsValidUserResult>boolean</IsValidUserResult>
</IsValidUserResponse>
</soap:Body>
</soap:Envelope>
```

Fig. 4.SOAP response

Web Services Description Language (WSDL):

It is really an XML formatted language used by UDDI. It describes the capabilities of the web service as, the collection of communication end points with the ability of exchanging messages. Or in simple words "Web Services Description Language is an XML-based language for describing Web services and how to access them".As per pen testing web services are concerned, understanding of WSDL file helps a lot in manual pen testing. We can divide WSDL file structure in to two parts according to our definition. 1st part describes what the web service and the 2ndparts tells how to access them. Let's start with basic WSDL structure as shown in Fig 5.

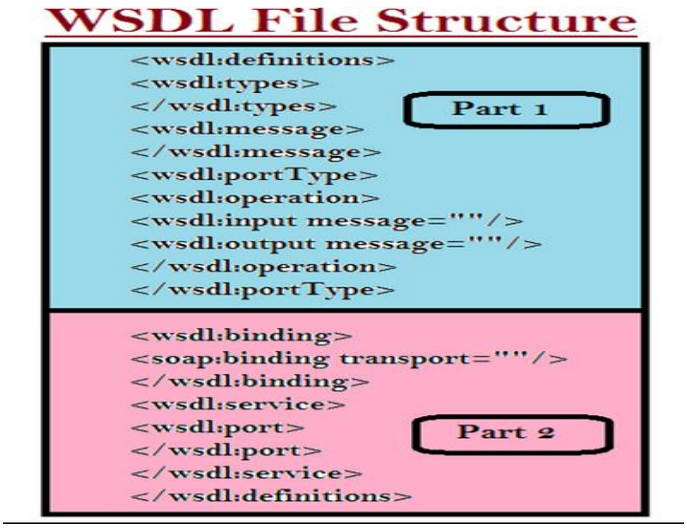

Fig. 5. WSDL

Universal Description, Discovery and Integration (UDDI)

UDDI is a distributive directory in the web every service providers needs to issue registered web services using its WSDL. The service consumer will search for appropriate web services and UDDI will provide the list of service providers offering that particular services. The service consumer choosing any one service provider and gets the WSDL. Web services are a standardized way of establishing communication between two Web-based applications are using the XML, SOAP, WSDL, UDDI and open standards is over an internet protocol backboned. XML is used to encoded that data in the SOAP message. SOAP is used to transform the information to HTTP, WSDL and is used to describe the capabilities in the web services and UDDI is used to provide the list of service provider details. In a real time scenario if a service consumer wants to use some sort to the web service then it should known the service providers. If a service provider validating the service consumer it will provide the WSDL file directly and then the service consumer creates a XML message to request for a required service in the form of a SOAP message and the service provider returns the service response. In another way, if a service consumer is not aware of the service provider to will visit UDDI and search for the required service. By choosing one service provider again the service consumer generates a XML message to request for a required service in the form of a SOAP message is specifying the WSDL file of that service provider. The service provider returns the service response. In general web service testing is assuming the service consumer and the service provider, start testing a web service is ask for the WSDL file.

ISSN(Online) : 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 11, November 2014

VI. FAULT INJECTION TOOL FOR WEB SERVICES

As it has been mentioned in the previous sections, the demand for SWIFI tools which enable a tester to create test specifications for fault injection and to apply them automatically at the specified parts of webservices is still growing. This section describes the development of the FIWS tool (Fault Injector for Web Services), it can be classified as tool for software implemented run-time fault injection and non-invasive robustness testing of SOAP-based, XML-RPC, and Restful web-services and their compositions. It allows to inject faults in run-time of tested web-services, into communication between the web-services and their partners. After the injection, the resulting abnormal states and erroneous behaviour of the web-services can be observed.

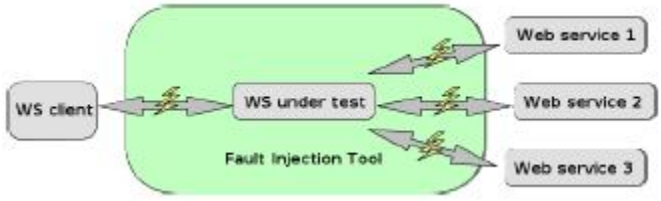

Fig. 6.The communication scheme.

Presentation layer that contains modules and classes responsible for a (graphical) user interface of the tool (it consists of classes representing windows and dialogs). The next one is the business layer implementing the application logic, which is composed of three units:

Proxy Monitoring Unit – Its goal is to act as a HTTP proxy, which listens at a given port for incoming connections and mediates the communication between services. Behaviors of this component are a bit different from common proxy servers. It waits until a whole HTTP message is read, and then, it analyses important parts of the message and forwards them to the Fault Injector.

Fault Injector – It receives specific parts of service call messages from the Proxy Monitoring Unit and processes them by classes representing different types of conditions and faults. The processing is defined by a particular test specification in the tool's user interface and received by the Controller.

Controller – It is a unit controlling the business layer according to events from the tool's user interface. It also implements the persistence of tests and results which are stored in a XML database.
Finally, the bottom-level layer is the data layer, which consists of classes accessing the mentioned XML database with test specifications and results.

VII. SOAP SONAR

As per the above explanations now we have to implement an innovative Tool called the SOAP Sonar which is considered to be the industry leading client emulation service Testing Tool. SOAP sonar is one of the comprehensive features combined with an intuitive user interface tool to enable code –free service testing that covers almost functional, Automation, Regression, Performance, Compliance and security test for XML, SOAP and JSON testing. It is a software testing and diagnostics tool for SOAP, XML and REST based web services. Generally SOAP Sonar interface is classified into Project View, Run View and Report View whereas the Project view is used to create the individual test cases, variables and success criteria. Run View is used to combine the test cases into automated test scenario. In Report View the results of independent test cases can be viewed to run and compared against the success criteria.

ISSN(Online) : 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 11, November 2014

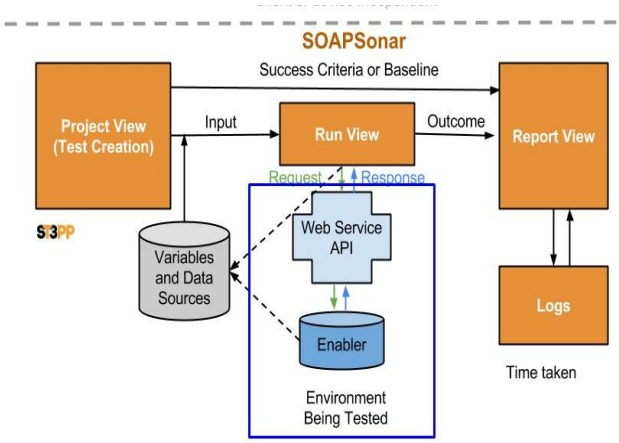

Fig 7.SOAP Sonar

VIII. CONCLUSION

In this paper, The results of the Penetration Testing phase helped to develop the rules for vulnerabilities analysis. However, the results obtained by soapUI show a large percentage of false positives and false negatives. We also verified the security provided by WS-Security standard with the add-on Security Token against XSS attack. In both phases, the use of WS-Security reduces significantly the number of vulnerabilities. However, this can be improved with the use of other specifications. One advantage of the proposed approach is that it relies on the use of a fault injector of general purpose, which can be used to emulate several types of attacks and may generate variants of the same, which is usually limited in the tools commonly used for security testing, as the vulnerabilities scanners.

REFERENCES

1. M.I. Ladan Web services: Security Challenges Proceedings of the World Congress on Internet Security, WorldCIS11, Londres, Reino Unido, 21–23, Fev 2011, IEEE Press (2011)
2. Zhao G., W. Zheng, J. Zhao, and H. Chen, *An Heuristic Method for Web-Service Program Security Testing*, In Proceedings of the 2009 Fourth ChinaGrid Annual Conference, CHINAGRID '09, IEEEComputer Society Press, Yantai, China, 21-22, Aug 2009.
3. Valenti AW, and E. Martins, *Testes de Robustezem Web Services porMeio de Injeo de Falhas*, Thesis(Master in Computer Science), Institute of Computing, UNICAMP, State University of Campinas,Brazil, 29, Jun 2011.
4. Holgersson, J., and E. Soderstrom, *Web Service Security-Vulnerabilities and Threats within the Contextof WS-Security*. SIIT 2005, ITU.
5. Lawrence, K., C. Kaler, A, Nadalin, R. Monzillo, and P. Hallam-Baker, *Web Services Security: SOAPMessage Security 1.1 (WS-Security 2006)*, OASIS, 2006.
6. Vieira M., N. Antunes, and H. Madeira, *Using Web Security Scanners to Detect Vulnerabilities inWeb Services*, In Proceedings of theIEEE/IFIP International Conference on Dependable Systems &Networks, DSN 09, IEEE Computer Society, Lisbon, Porgugal, 2009.
7. Hsueh MC, TK Tsai, and RK Iyer, *Fault Injection Techniques and Tools*, IEEE Computer SocietyPress, Computer; Volumen 30, Ed. 4, Apr 1997.
8. Morais A., E. Martins, A. Cavalli, and W. Jimenez, *Security Protocol Testing Using Attack Trees*, InProceedings of the International Conference on Computational Science and Engineering, 2009, CSEIEEE Computer Society Press, So Paulo, Brasil, 29-31, Aug 2009.
9. Valente AW, MY. Maja, E. Martins, F. Bessayah, and A. Cavalli, *WSInject: A Fault Injection Tool forWeb Services [Technical Report]*, Institute of Computing, UNICAMP, State University of Campinas,Brazil, July 2010.
10. Della-Libera, G., et al, *Security in a Web Services World A Proposed Architecture and Roadmap*, IBMCorp, Microsoft Corp, 7, Apr 2002.
11. Consideration Points: Detecting Cross-Site Scripting: SumanSahaDept. of Computer Science and Engineering ,*Vol. 4, No. 1 & 2, 2009.*

BIOGRAPHY

Prof.Mr.R.Ramakrishnan received his PG Degree in Master of Computer Applications, from Madurai Kamarajar University, Madurai in 2000, M.Phil in Computer Science from Periyar University and M.Tech in Computer Science and Engineering, from Pondicherry University, He has over 13 years of experience in teaching and 10 years in software industry. He is working as Assistant Professor in Department of Master of Computer Application at Sri

ISSN(Online) : 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 11, November 2014

Manakula Vinayagar Engineering College, Pondicherry. He has published and presented more than 25 papers in various International and national journals and conferences. His area of interest includes wireless networks and multimedia systems.

J.Anbarasi obtained her B.Sc.(Computer Science) degree from Barathidhashan Govt Girls College, Pondicherry, India. She is currently pursuing her MCA degree in at Sri Manakula Vinayagar Engineering College, Madagadipet, India. Her areas of research interest accumulate in the areas of Software Testing.

V.Kavitha obtained her B.Sc.(Mathematics) degree from Saradha GangadharanCollege, Pondicherry, India. She is currently pursuing her MCA degree in at Sri Manakula Engineering College,Madagadipet, India. Her areas of research interest accumulate in the areas of Software Engineering.