Danish Jamil et al. / International Journal of Engineering Science and Technology (IJEST)

**SECURITY IMPLICATION OF SOAP**

**AND WEBSERVICE INTERFACE TO**

**THE CLOUD COMPUTING SYSTEM**

DANISH JAMIL \*

Department of Computer Engineering, Sir Syed University of Engineering & Technology, Main University

Road, Karachi, Sindh-75300, Pakistan

HASSAN ZAKI

Department of Computer Engineering, Sir Syed University of Engineering & Technology, Main University

Road, Karachi, Sindh-75300, Pakistan

**Abstract :**

Cloud Computing system launched a evolution in IT industry with removing application from user’s PC. It

could be implemented in Webservice technology and its interface will be no more than a web browser.

Consequently, the inherent vulnerabilities of webservice also can make Cloud Computing System under the

same risk as it has. In this report, I am going to discuss the security issues of Clouding Computing System

caused by webservice. This paper is comprised of four parts; the first and second part is used to introduce some

back ground about Cloud Computing and Webservice. The security issues will be discussed in third and forth

parts. The security flaw in Saas layer, caused by webservice, is arranged in part three. The last part explained

what the vulnerabilities in Iaas layer are.

***Keywords— Security Issues, Webservice, Cloud Computing, Security flaw, Vulnerabilities, Saas layer.***

**1. INTRODUCTION**

Nowadays, traditional applications which are installed in user’s PC are facing many issues. For example, its

copy right could be overridden by reverse engine or PC may can not fulfill the minimum requirement of the

application. Cloud Computing which is considered as the future of Internet provides a solution for these issues.

Cloud Computing is a new concept whose purpose is making its customers get rid of installing any application

in their PCs. Users only need use their web browsers to access the application online and could be charged in

pay-as-you-go mode. It not only can provide tremendous capacity to calculate massive data, but also let the

products avoid being cracked. So far there is still not any manifest definition about how to implement Cloud

Computing system. However, after web 2.0 is widely deployed in the IT industry, most people prefer to design

Cloud Computing System by combining Ajax and Webservice. Webservice is an Internet based application and

it also has many security issues. The purpose of this article is explaining how the Cloud Computing System is

affected by the vulnerabilities of Webservice.

**1.1.  *BACKGROUND***

**A. What is Cloud Computing**

Cloud Computing is considered as a future technology by IT practitioners. However, currently it is still a

nascent concept in IT industry. Its purpose is using Internet based service to instead of the application installed

in user’s computers. Thus, users will no longer need to install application in their computer and can be charged

each time when they use it. The word “cloud” is an analogy of the service provided to customers. At present,

different organizations are still using their own principle to build their own system and none of them seems can

overcome others to become a standard. However, the architecture of cloud computing still can be summarized

by using Xaas (Every thing as a service)[1].

ISSN : 0975-5462

Vol. 3 No. 4 April 2011

3472

Danish Jamil et al. / International Journal of Engineering Science and Technology (IJEST)

Software as a Service (Saas)

Platform as a Service (Paas)

Infrastructure as a Service (Iaas)

Hardware as a Service (Haas)

Fig. 1 Architecture of Cloud Computing System

Figure 1 in above illustrated Cloud Computing System can be divided into 4 layers. The first layer represents the

application or software used to provide the service for customers. The second layer is the Platform where

application is implemented by its designers. Third layer is comprised of widely distributed infrastructures which

are used to provide data for the two upper layers. The bottom layer is build by the hardware, where the entire

system is running. Currently, there are already many Cloud Computing products in the market, for instance

Google App and Amazon EC2. Although their system architecture is compliant with the 4 Layers architecture,

they still are providing cloud service in different layer. Google App is a good example of Cloud Computing

whose service is running on Saas layer. It make users can use Word application online and saved it on the server

as well. On the contrary, Amazon EC2 provides their cloud service in the Iaas layer which allows user to upload

their own applications and run it online.

**B. Webservice for Cloud Computing**

Webservice, which actually is a kind of online application, is considered as the technology very suitable for

implementing Cloud Computing system. The structure of Webservice is designed in Client-Server format as

well. The server will provide a XML file, known as WSDL, for users to design their interface. WSDL dose not

have any restriction on the technology used to develop the interface, so it is possible to design it a variety of

ways. Generally speaking, Webserivce could be arranged in Saas, Paas and Iaas layers.[2]







Saas: In this layer, Webservice will run as the application which can be accessed directly via browser

by users.

Paas: Webservice could be combined with other techniques to provide an environment for the

application in Saas layer.

Iaas: Webserive play a role of massive data center for the application in the upper layer.

2. SECURITY ISSUES IN SAAS LAYER

In the SaaS layer, Webservice can be employed to provide cloud service to users. In the Cloud Computing

system, browser is supposed to be the only interface used by users to communicate with server. Thus,

Webservice is quite suitable to implement Saas layer via combined with Web2.0. For this reason, the security

issues of Webservice also make Cloud Computing Vulnerable.

**A. Problem of SOAP Message**

SOAP is a protocol employed by Webservice to transfer message between client and server sides. The message

in fact is designed in XML format and its content is no more than plain text. In this case, it causes Cloud

Computing system is extremely vulnerable to data tampering, sniffing and intercepting. These problems could

be solved by adding a Header[3] to the SOAP message. The header contains tags which can be used to encrypt

content of the XML file. In the header, the tag, <xenc:ReferenceList>, defines which portion of SOAP message

will be encrypted. The encrypted portion is called <xenc:EncryptedData> and its encryption key is stored its sub

element called <ds:KeyInfo>. Figure 2 is an example of the header.

ISSN : 0975-5462

Vol. 3 No. 4 April 2011

3473



Danish Jamil et al. / International Journal of Engineering Science and Technology (IJEST)

Fig.2 Encryption of SOAP message

**B. The problem of Browser**

As what has been discussed in above, the SOAP message could be protected by using encryption. However,

currently all the browsers in the market can not generate XML header for encrypting SOAP message. There are

two solutions to tackle this issue. SSL is the first method which can be used to replace the XML header. It can

use Public and Private Keys to encrypt message in order to guarantee only recipient is able to understand it.

However, SSL is also vulnerable to Phishing attack. [2][4] Hackers can use Phising attack to send a spoofed

webpage to user instead of the one delivered from the real server. If the user fails to note the webpage has been

tampered, all the input data will be received by attackers.

Comparing with the first method, the second one is more secure. It is implemented by combining SSL with

Microsoft’s Passport. [2] Microsoft Passport is a third party agent, which can be used to store the redirection to

the requested server. After user successfully login the passport server, the agent can generate a Kerberos token

by utilizing the user’s credentials. (e.g. username & password) The token could be utilized to generate the XML

header and will be transferred to the expected recipient through the pre-stored redirection. Thus, the SOAP

message could be added a header by combining with Microsoft Passport. However, it also makes SOAP

message under a high risk of being attacked, because only Microsoft Passport only protects Kerberos Token

with Same Original Policy (SOP). Fortunately, this problem could still be solved through invoking SSL in many

ways.[2] For example, using SOP to compare Public Key received by each client rather than its DNS

information.

**C. Bypass Authentication**

For the reason that the communication between web browser and server should strictly follow the definition of

WSDL, hackers can exploit it to bypass authentication as well. When users send message to server, according to

the WSDL they have to specify the type of input and action to execute. In this case, the system will be extremely

vulnerable, if intruders can intercept the transmitted data. OWASP has offered a tutorial for explaining how to

exploit this vulnerability. Hackers can gain more right via tamper the name of action. Figure 3 is an example of

attack described above. The attacker intercepted the request and successfully executed unauthorized action via

change the action’s name.

ISSN : 0975-5462

Vol. 3 No. 4 April 2011

3474



Danish Jamil et al. / International Journal of Engineering Science and Technology (IJEST)

Fig.3 The request action intercepted by DataTamper

**D. SQL injection**

This attack is deemed a serious threat against database and seems to be irrelevant to Webservice. However,

Cloud Computing requires webservice has to employ browser as interface and sending data via HTTP protocol.

In this case, the technique used by hackers to perform SQL injection on traditional website is also effective to

Cloud Computing System. The countermeasure for this problem should be the same as the traditional one.

E. Deny of Service

Deny of Service attack can be divided into two categories, Inputting massive data and flooding attack. These

two types of attacks could be achieved through inputting or flooding massive data to the server. They can cause

server has to arrange more memory for dealing with these requests and finally make the system collapse. When

attackers want to achieve it via inputting massive data, normally they will not just only input hundreds simple

characters. Instead of those characters, they will input many XML elements which are defined in WSDL. As a

result, the server will keep busy with dealing those tags until it collapsed. Following figure is an example of Dos

attack via inputting numerous elements.

Fig.4 DOS attack through inputting many elements

The data in the flooding attack is same as regular request. However, this attack is accomplished by many

computers. These computers could be zombie computers controlled by hackers via Trojans. Thus, if an attacker

starts 100 requests in the same time, target will be stopped.

**3. THE SECURITY ISSUES IN IAAS AND PAAS LAYERS**

**A. XML WRAPPING ATTACK**

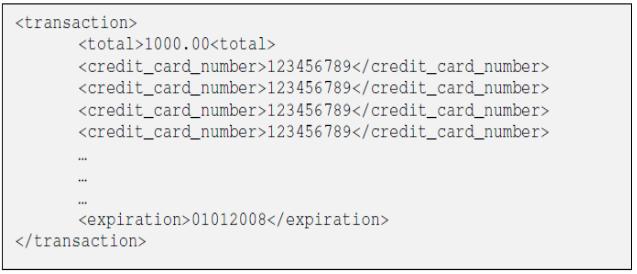
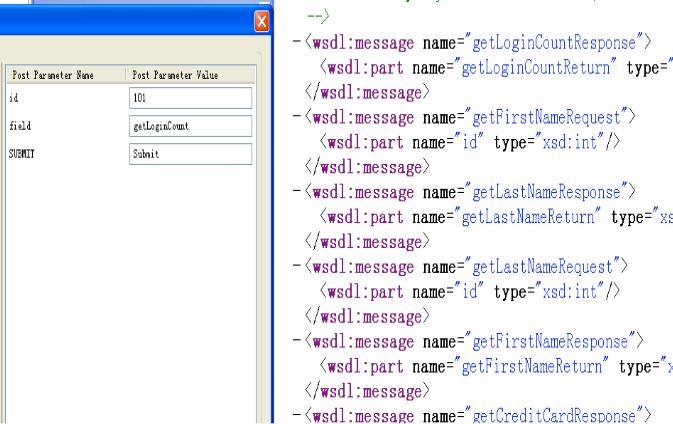
In the IAAS and PAAS layers, normally Cloud Computing service providers would like offer WSDL file for

vendors to design their application. In this case, if the attackers successfully get the WSDL file, they may

ISSN : 0975-5462

Vol. 3 No. 4 April 2011

3475



Danish Jamil et al. / International Journal of Engineering Science and Technology (IJEST)

exploit XML wrapping to bypass authentication.[5][6]. XML wrapping works more or less in the same way as

SQL injection. Because of the message send by user is in XML format, all the input data will be wrapped into

XML tags, called elements. Consequently, if the user input some elements, the SOAP message will treat them as

its inherent elements as well. Under this condition, hackers can modify unauthorized data by injecting relevant

elements. Figure 4 in below is an example of XML wrapping attack. The hacker injected two elements in order

to change the data of user 102. The countermeasure for this weak point also can be referred to the one for SQL

injection. The vendor could prevent this attack by either making a filter for user’s input to remove the key words

or specifying the length of expected input.

Fig.5 XML Wrapping Attack

**B. WSDL Enumeration**

WSDL usually is designed with XML schema. In the face, it is not more than a plain XML file. Thus, if the

XML file is stolen or successfully guessed by attackers, the whole cloud computing system could be taken over

by them. These are many ways to dump the WSDL file from webservice system. The first method, also the

easiest way, is google hacking. Google currently has offered many special operations for users to narrow their

searching range. For example, the keyword “ url: “ means looking to expected information in target’s URL.

Thus, an attacker uses some keywords to indicate expected results containing WSDL, the WSDL may be found

by Google robot. The keywords in below are some

Examples which can be used to search WSDL through Google [7].

a) fieltype:wsdl

b) indexof “/wsdl”

c) inurl:wsdl

d) inurl:asmx

Sometimes, webservice designer would like to arrange the WSDL file under the same directory with webpage

interface. For this reason, attacker may collect the WSDL file by relacing the all the strings, which is behind the

question mark in url, with WSDL.For example: target/webservice?WSDL.

**4. CONCLUSION**

As what has been discussed in above, as an core technology for implementing Cloud Computing System,

websevice play a very important role for its security issues. The security issues are widely deployed in the Saas,

Iaas and Paas layer. In the Saas, where webservice is employed as the substance of cloud computing system to

provide cloud service for customers, Webservice are facing problems from SOAP message, Web Browser, SQL

ISSN : 0975-5462

Vol. 3 No. 4 April 2011

3476



Danish Jamil et al. / International Journal of Engineering Science and Technology (IJEST)

injection and DoS attack. The SOAP message is normally written in plain text and Web Browser will not

provide a particular function to encrypt it. In this case, a new function for encrypting XML message is

expected.In the Paas and Iaas layers, webservice is employed to work as a working environment or data centre

to support the whole system. In this case, many webservice providers will give an WSDL file to Clouding

Computing System developer. However, the WSDL file also can be used to by attackers. In this case, it makes

Cloud Computing System quite vulnerable to WSDL wrapping attack and WSDL enumeration. In this case, the

vendor should think carefully about how to secure their WSDL properly.

**5. ACKNOWLEDGMENT**

I convey my honest thanks to Dr. Syed Faisal Ahmed Bukhari Chairman of Computer Engineering Department

in the Sir Syed University of Engineering and Technology for providing me the leadership and conveniences for

this paper. I expand my truthful gratitude to Mr. Muhammad Numan Ali Khan for his cooperation for presenting

this paper. I also extend my sincere thanks to all other faculty members of Sir Syed University of Engineering

and Technology and my friends for their support and encouragement.

**6. REFERENCES**

[1]

[2]

B.P.R. Enumi Choi and Ian Lumb, *A Taxonomy and Survey of Cloud Computing Systems,* Fifth International Joint Conference on INC,

IMS and IDC, papes 44-45 ,2009.

M.Jensen and N.Gruschka, *On Technical Security Issues in Cloud Computing,* 2009 IEEE International Conference on Clouding

Computing*,* pages *109 -116, 2009.*

[3]

[4]

(2002) The microsoft website, [Online], Available : http://msdn.microsoft.com/en-us/library/ms951257.aspx#wssecuritytoc6120253

C.L. Abad and R.I. Bonilla, *An Analysis on the Schemes for Detecting and Preveting ARP Cache Poisoning Attack,* IEEE.org, Pages 1

– 8, 2007.

[5]

N. Gruschka and L.L. Iacono, *Vulnerable cloud,* 2009 IEEE International Conference on Clouding Computing pages, Pages 625 -631,

2009.

[6]

[7]

(2010)Webgoat Tutorial, [online], Available http://yehg.net/lab/pr0js/training/webgoat.php#Authentication\_Flaws

N. Bhalla and S. Kazerooni. *Web Service Vulnerabilities, Security,* Security Compass, 15 February 2007, pp 6.

ISSN : 0975-5462

Vol. 3 No. 4 April 2011

3477

