SECURITY AND PERFORMANCE OF WEB BASED APPLICATION BY USING INTEGRATED FRAMEWORK

[1] Lukesh R.kadu, [2] Dr.Jagdish W.Bakal
[1] *Student of ME (Computer), Pillai Institute of Technology , New panvel, Navi Mumbai, India*
[2] *Principal,S.S.Jondhale Coe,Dombivili (E),Mumbai,India*

[1] lukesh_kadu@rediffmail.com
[2] jwbakal@gmail.com

Abstract: **-To address the challenges in Web services security, the we firstly analyzed threats facing Web services and related security standards, presented integrated security framework based on use of authentication, authorization, confidentiality, and integrity mechanisms for Web services, and proposed how to integrate and implement these security mechanisms in order to make Web services more secure against the attacks. Also the performance of web application is become critical in recent era. so proposed here general purpose integrated framework for seure web based application.**

Keywords: ***web services; security; authentication; authorization; Confidentiality;Role based access control; secure web application project***

I. INTRODUCTION

Due to the evolution of Internet technology and application popularization, security and performance have become key issues for implementing Web-based applications. Presently, most of the Web-based applications design only consider security issue, but ignore performance problem. According to these two issues, we propose a new approach to integrate security and performance aspects for Web-based applications.

The advance of Web services technologies promises to have far-reaching effects on the Internet and enterprise networks. Web services based on the eXtensible Markup Language (XML), SOAP, and related open standards, and deployed in Service Oriented Architectures (SOA) allow data and applications to interact without human intruption through dynamic and ad hoc connections. Web services technology can be implemented in a wide variety of architectures, can co-exist with other technologies and software design approaches, and can be adopted in an evolutionary manner without requiring major transformations to support applications and databases. The security challenges [1] presented by the Web services approach are critical and unavoidable. Many of the features that make Web services attractive, including greater accessibility of data, dynamic application-to-application connections, and relative autonomy (lack of human intervention) are at odds with traditional security models and controls. Difficult issues and unsolved problems exist, such as protecting the following: 1) Confidentiality and integrity of data that is transmitted via Web services protocols in service to- service transactions, including data which is traverses in intermediary services.

2) Functional integrity of the Web services that requires the establishment of trust between services on a transaction-by-transaction basis.

Current security mechanisms on the application layer are not enough to prevent any attacks from hackers. The enterprise makes use of information technology (IT) to create its competition and value. Due to advanced IT, it can change business operation and living style, and exchange information transparently via Internet, Intranet, or Extranet. The Web-based applications of enterprise make internal employee exchange and retrieve Web pages that stored in different application systems on the Intranet. Also, they can strengthen the competitiveness and reduce the cost. Since the Internet environment is open and global, the attacks of the hacker are omnipresent. Hence, the security problems of Web-Based applications are more and more important.

Though firewall can efficiently prevent attacks on the Intranet, most intrusion attacks are still occurred in business operation or data management on the application layer. These events may include illegal Web pages access, theft of sensitive data, purchasing goods on low price, stealing user's Cookie via JavaScript. Currently, we can find that most of the Web-based applications design only consider security issue and ignore performance problem. Hence, we need to consider performance problem in designing Web-based applications. Therefore, the purpose of this paper is to propose the general purpose framework for solving security events and improving processing performance issues of Web-based applications on the Intranet

Finally, our experimental results in section VII indicate that the propose general purpose integrated framework can solve related security holes, obtain better processing performance, and reduce development time and cost.

II. ATTACKS FACING WEB SERVICE AND RELATED SECURITY PARAMETERS

There are a wealth of security standards and technologies available for securing Web services, they may not be sufficient or necessary for a particular organization or an individual service. for that reason, it is important to understand the attacks that face Web services so that

organizations can determine which threats their Web services must be secured against. The common threats facing Web services are [2]

1) Message alteration. An attacker inserts, removes or modifies information within a message to deceive the receiver

2)Loss of confidentiality. Information within a message is disclosed to an unauthorized individual

3)Falsified messages. Fictitious messages that an attacker intends the receiver to believe are sent from a valid sender

4)Man in the middle. A third party sits between the sender and provider and forwards messages such that the two participants are unaware, allowing the attacker to view and modify all messages

5)Principal spoofing. An attacker constructs and sends a message with credentials such that it appears to be from a different, authorized principal.

6) Denial of service. An attacker cause the system to allocate resources disproportionately such that valid requests cannot be met.

III. LITERATURE REVIEW

The solutions about security problems in designing Web-Based applications contain SWAP (Secure Web Applications Project), RBAC (Role-Based Access Control), Language-Based Systems, Commercial System, and others. First, we examine SWAP, RBAC, and Language-Based Systems approaches

Researchers have proposed many kinds of security model, such as Discretionary Access Control (DAC) and Mandatory Access Control (MAC)[3]. These models apply to different applications and application environments, and are usually based on user-group. While DAC and MAC can hardly be enforced in open networks, RBAC has the advantages of simplifying the management, and it is one of the more promising approaches to handle sets of access rights. The concept of RBAC began with multi-user and multi-application on-line systems pioneered in the 1970s.the basic idea of the RBAC model is that permissions are assigned to roles rather than users [4]. The users acquire permissions by acting members of roles as shown in Fig 1.

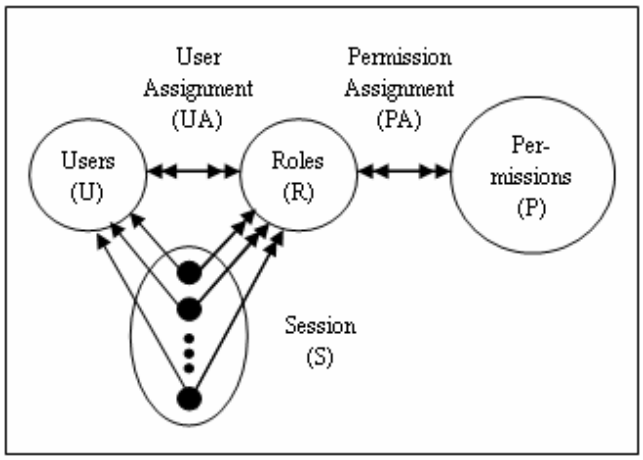

Fig.1. RBAC ARCHITECTURE IN WEB BASED APPLICATION

A typical Web-based application using the SWAP tool as shown in Fig 2. has source code written in three different languages, as well as some third-party component in which we assume that the source code is inaccessible. Spectre protects Language C or third-party component of the application by dynamically transforming HTTP requests and responses in accordance with the refined security policy of SPDL for this application. As you can see, Spectre does not protect the parts of the application written in Languages A and B. Instead, our secure IDL compiler has folded that the relevant parts of the SPDL policy are directly mapped into the source code. Also, the IDL compiler supports multiple languages, allowing us to convert the SPDL into Language A and Language B [5].

In order to run un trusted code in the same process as trusted code, there must be a mechanism to allow dangerous calls to determine if their caller is authorized to exercise the privilege of using the dangerous routine. Java systems have adopted a technique called stack inspection to address this concern. Stack inspection is an algorithm for preventing un trusted code from using sensitive system resources

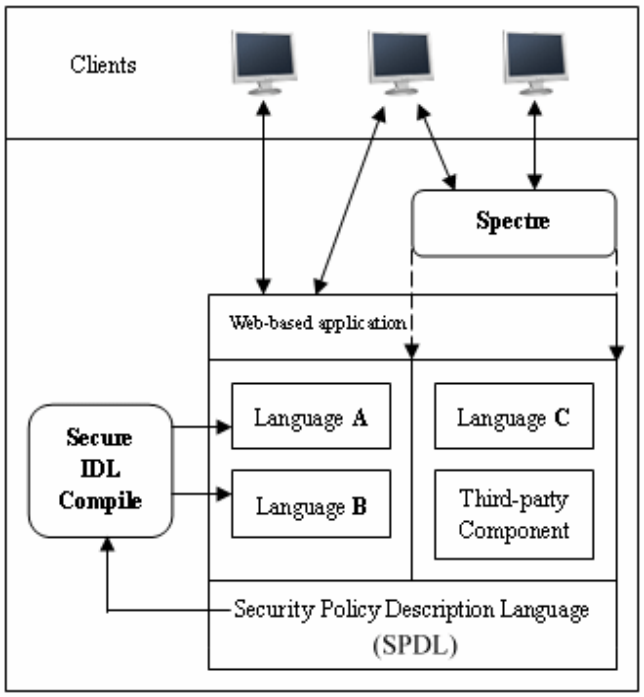

Fig.2. SWAP ARCHITECTURE IN WEB BASED APPLICATION

Some Web services and HTTP standards can protect against many of these threats

1)W3C XML Encryption. Used by WS-Security to encrypt messages and provide confidentiality of part or all of SOAP message

2)W3C XML Signature. Used by WS-Security to digitally sign messages and provide message integrity and sender authentication

3)WS-Security Tokens. Allows messages to include credentials to aid receivers in determining whether or not the message sender is authorized to perform the requested action.

4)W3C WS-Addressing IDs. Allows the message sender

to supply a unique identifier for the message
5)IETF SSL/TLS. Secures the HTTP protocol over which SOAP messages are sent and received SSL/TLS with client authentication. Requires both the sender and receiver to authenticate with one another before securing the HTTP protocol.
SWAP approach only protects interactions among Web pages. In contrast, RBAC approach is only restricted user to access Web pages, but not protect interactions among Web pages. In addition, the Web-based application development process is a multi-stage process consisting of: design, coding, testing and development. Usually, securing a Web-based application is difficult, but the authors have proposed to solve the security of the applications using Language-Based systems such like J2ME programming tool. Also some scientist have proposed the specific domain modeling and the service-oriented design frameworks for designing the secure Web-based applications [6,7]So far as now, most of the scientist only focus on designing the secure Web-based applications. However, a full-fledged Web-based applications general purpose integrated framework to support both security and performance feature.

IV. ANALYSIS

In designing Web-based applications, there are some essential design issues for implementing the system. However, we will propose following design issues according to two main aspects: security and performance.

Design issue for security

The rise of the Internet with opportunities to connect computes anywhere in the world has increased the threat of the organization's Web-based applications. Hence, the Web-based applications are required to concern with security of accessing and storing information. In particular, following design issues are to solve the security problems in a Web-based application:
(1) Access control: Different users and roles can have different interfaces or permissions.
(2) SQL injection: Stream SQL statement to attack database or records in database.
(3) Session hijacking: Grab session Id of Web user and enter user's Session.
(4) Information disclosure: Let user look the detail information while occurs exception.
(5) Hidden-field tampering: Update contents of hidden-field in the Web pages.

Design issue for performance

Besides security, the second most important characteristics of Web-based applications is performance. In general, following performance solutions are concerned with programming in a Web-based application:
(1). Time for accessing Web Page: It will suffer from the effect of programming structure.
(2). Time for manipulating Database:

. Data Query: Time for selecting data from database.

. Data Insert: Time for inserting data into database.

. Data Update: Time for updating data from database.

. Data Delete: Time for deleting from database.

So above two demands as stated ,are driving the need for designing Web-based applications. Therefore, we will develop general purpose integrated framework to support security and performance features on the Web-based applications.

V.DESIGN

ALGORITHM OF GENERAL PURPOSE INTEGRATED FRAMEWORK

As, already stated that general purpose framework for secure web based application combines RBAC and SWAP methods and enhances performance at the same time during in designing Web-based applications. Actually, the user will start to access Web pages using the RBAC and then make some interactions between any two Web pages using SWAP with tuning performance under general purpose integrated framework.
Based on design and performance issues here is algorithm

MAIN FUNCTION( )
{
Input:
Create a database WBA
Create a table Tb_User (U_Id, U_Name, Gender)
Create a table Tb_Login (R_Id, U_Id, Uid, Upswd)
Create a table Tb_Role (R_Id, R_Name)
Create a table Tb_Permission (Per_Id, Per_Name)
Create a table Tb_AccessRight (A_Id, R_Id, Per_Id)
Create another tables that user will to use.
Dim flag as int
Dim uid as user input
Dim upswd as user input

Method:
START
If user want to access the web page that is not login page
To navigate 'login' page.
Else
{
Display 'login' page.
User input uid in field.
User input upswd in field.
If 'id' field or 'password' field is null then
To navigate 'login' page.
Else
Do
{
Query table Tb_Login all field.
If 'id' field is equal to table Tb_Login field Uid then
If 'upswd' field equal to table Tb_Login field Upswd then
{
Set flag to '1'.
Access_Right(R_Id)
Break;
}
}
while (not end of table)
if flag is not equal to '1' then

To navigate 'login' page.
}
}
END
Access_Right (R_Id)
{
Query table Tb_ Permission field Per_Id via R_Id.
Query table Tb_Permission field Per_Name via Per_Id.
Display user interface and then user executes permission.
User_Operation( )
}
User_Operation ( )
{
If user want to insert data into database WBA then
{
Input all fields values
Argument_Check (field*1*, field*2*,…,field*n*)
}
If update data from database WBA then
{
Input all fields values
}

The key performance tuning issues of general purpose framework are mainly focused on accessing Web pages on the Web Server and manipulating tables within a Database.

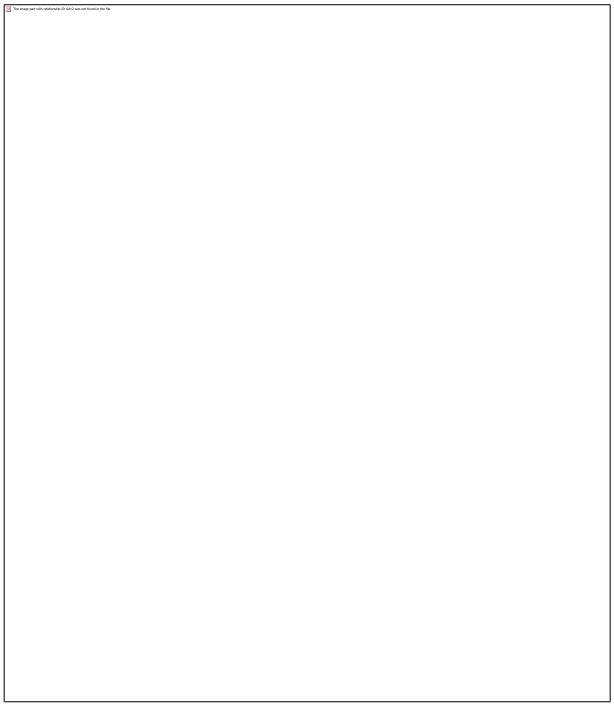

Fig.3.RBAC FLOWCHART IN GENERAL PURPOSE INTEGRATED FRAMEWORK

V. PRACTICAL CASE IMPLEMENTATION

Actually, the implementation procedure is divided into security and performance instances.

PART 1: Security Instance
(1)Access control: Different user's logins will have different interfaces based on user defined role.

(2) SQL injection : It can prevent data leak from this feature

(3) Coding for seession hijacking
Session.Add("id", "u"+ uid.Text + "s");
Session.Add("pswd", "u" + upswd.Text + "u");

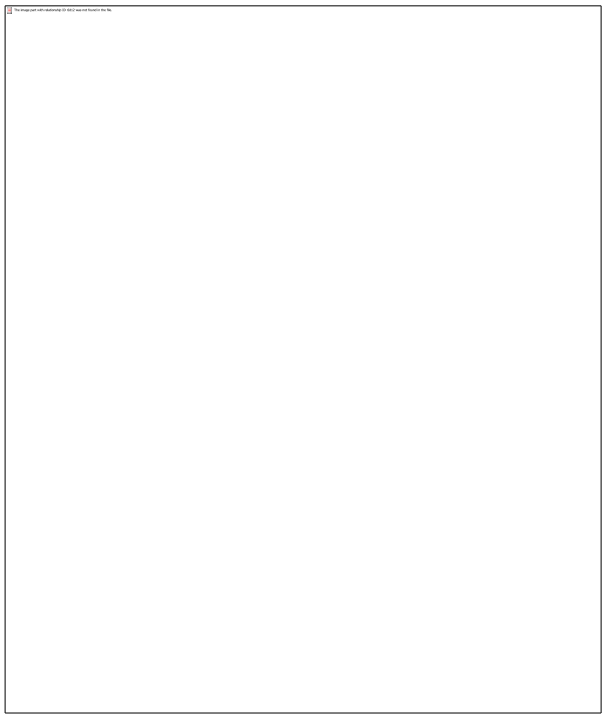

Fig.4.SWAP FLOWCHART IN GENERAL PURPOSE INTEGRATED FRAMEWORK

(4) Coding for Information disclosure:
try
{
myCommand.ExecuteNonQuery();
Message.InnerHtml = "insertion O.K.
";
}
catch (SqlException e)
{
if (e.Number == 2627)
Message.InnerHtml = "The Same Index KEY!!";
else
Message.InnerHtml = "Is all data stuffing?";
Message.Style["color"] = "red";

}

PART II: Performance Instances
(1) Time for accessing Web Pages
(2) Time for manipulating Database
. *Data Query*
SqlDataAdapter myCommand = new
SqlDataAdapter("select * from Tb_Users", myConnection);
DataSet ds = new DataSet();
myCommand.Fill(ds, " Tb_Users ");

```
MyDataGrid.DataSource = ds.Tables["Tb_Users
"].DefaultView;
MyDataGrid.DataBind();
```

. *Data Insert*

```
String insertCmd = "insert into Tb_Users (name, pho,
addr) values (@Name, @Pho, @Addr)";
SqlCommand myCommand = new
SqlCommand(insertCmd, myConnection);
myCommand.Parameters.Add(new
SqlParameter("@Name", SqlDbType.NVarChar, 20));
myCommand.Parameters["@Name"].Value = name.Value;
…
myCommand.Connection.Open();
myCommand.ExecuteNonQuery();
```

. *Data Update*

```
String updateCmd = "UPDATE Tb_Users SET name =
@Name, pho = @Pho, "addr = @Addr where User_Id =
@Id";
SqlCommand myCommand = new
SqlCommand(updateCmd, myConnection);
myCommand.Parameters.Add(new
SqlParameter("@Name", SqlDbType.NVarChar, 40));
…….

MyDataGrid.DataKeys[(int)e.Item.ItemIndex];
String[] cols = { "@Name", "@Pho", "@Addr" };
```

Approaches Features	Traditional Approaches		General Purpose Integrated Framework	
	performance	security	performance	security
Login	8 * 10-3	2	6 * 10-3	1
Web page accessing	7 * 10-3	2	5 * 10-3	1
DA*-Data Query	9*10 -3	2	6*10 -3	1
DA*-Data Insert	11*10-3	3	8*10-3	2
DA*-Data Update	7*10-3	3	4*10-3	1
DA*-Data Delete	4*10-3	3	2*10-3	2

.

VII .RESULT ANALYSIS

Our models are validated by using proposed and traditional approach for implementing practical ERP system to compare their security level and processing performance. Below given the comparision between traditional integrated approach as shown in Fig.5

Fig.5. COMPARISION BETWEEN TRADITIONAL FRAMEWORK AND GENERAL PURPOSE INTEGRATED FRAMEWORK

VIII.CONCLUSION

In this paper, we propose the general purpose integrated framework and design its algorithm. Also, we illustrate how to implement the secure Web-based applications with tuning performance. Then, we will utilize this framework to set up a practical case – ERP system, and also perform simulations and make results analysis. Finally, our experimental results indicate that the proposed general purpose integrated framework can solve related security holes, and also improve performance.

IX. REFERENCES

[1] E. Pimenidis and C. K. Georgiadis, "Web services security evaluation considerations," International Journal of Electronic Security and Digital Forensics, Vol. 2, pp. 239-252, Mar. 2009

[2] J. Schwarz, B. Hartman, and A. Nadalin, "Security challenges, threats and countermeasures," White Paper, The Web Services-Interoperability Organization, May. 2005.

[3] Guoping Zhang "An Extended Role Based Access Control Model for the Internet of Things" International Conference on Information, Networking and Automation June 2010.

[4] Min Wu, Jiaxun Chen, Yongsheng Ding "Study on RoleBased Access Control Model for Web Services and its Application"

[5] David Scott and Richard Sharp, "Developing Secure Web Applications," *IEEE Internet Computing*, Vol. 6, No. 6, pp. 38-45, November 2002.

[7] Hiroshi Wada, Junichi Suzuki, "A Domain Specific Modeling Framework for Secure Network Applications," In *Proceedings of 30th Annual International Computer Software and Applications Conference* (COMPSAC'06), pp. 353-355, September 2006

[8] Hiroshi Wada, Junichi Suzuki and Katsuya Oba, "A Service-Oriented Design Framework for Secure Network Applications," In *Proceedings of 30th Annual International Computer Software and Applications Conference* (COMPSAC'06), pp. 359-368, September 2006

[9] www.google.com