Journal of Theoretical and Applied Information Technology
20th June 2014. Vol. 64 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

AN ENHANCED TRUST AUTHORIZATION BASED WEB SERVICES ACCESS CONTROL MODEL

R.JOSEPH MANOJ[1], A.CHANDRASEKAR[2]

[1]Research Scholar, Manonmaniam Sundaranar University, Tirunelveli, India &
Associate Professor, St.Joseph's College of Engineering, Chennai, India.
[2]Professor, St.Joseph's College of Engineering, Chennai, India.
Email: rjmanoj79@gmail.com , drchandrucse@gmail.com

ABSTRACT

With the increasing tendency among business organizations to move around the web services platform, Web services paradigm creates new security challenges which can only be realized by developing effective access control models. Now a day service provider's big challenge is detecting and preventing malicious requesters or behaviours. In order to evade malicious requesters or behaviors, the service providers who allows service requester to access the web services, requires development of access control models that can capture relevant information about a service requester at the time of access request and incorporate this information for making effective access control decisions. This proposed system provides an enhanced approach to detect and prevent IP address spoofing, SQL injection to avoid unauthorized users and capture relevant information such as network conditions, frequency of access, timeout, success rate, failure rate etc., about a service requester and establish trust value. Based on the trust value the honest and active users will be allowed to access the service otherwise their trust value will be decreased and not be allowed to access the service. So this method would control malicious requesters to access the web services and make the authorization process effective. This paper also compares existing system and proposed system and verifies the performance and correctness of the proposed work based on simulation results from a prototype implementation.

Keyword- *Web Services, Trust based Access Control, Access model for service requester, Trust Authorization Model*

1. INTRODUCTION

Web Services are becoming a popular technology which brings great economic benefits to people in the development of complex web applications. Web services describes a standard way of integrating Web applications using the eXtensible Markup Language (XML), Simple Object Access Protocol (SOAP), Web Services Description Language (WSDL) and Universal Description, Discovery and Integration (UDDI) open standards over an Internet protocol backbone.

1.1 An Overview of Access Control Model

An Access control model for a web service is to restrict the set of clients who invoke the operations offered by the service. Access control policies define rules stating that only subjects with certain credentials satisfying specific conditions can interact with a web service. Thus it is required to cross the border of security domains and to address the movement of unknown users across borders so that access to services can be granted. Few of the traditional access control models [1] are given as follows:

Role-Based Access Control (RBAC) is the one of the most widely used Web Service access control schemes. In such access control schemes, clients are assigned roles that contain permissions in order to gain a secure access to specific Web Services.

Attribute-Based Access Control (ABAC) models make use of attributes owned by the clients, the providers, and some other attributes related to the environment. Decisions are be made to allow or deny the request based on all these attributes.

Trust-Based Access Control systems (TBAC) are different from the previous access control schemes since the client trust level is dynamically calculated based on some statistical analysis of behaviors, activities and previous access attempts. Thus, service violations and bad client behavior lead to a decrease of the trust level, whereas good behavior leads to an increase in the trust level.

The rest of the paper is organized as follows. In Section 2, related work of trust based

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

access model has been discussed. In Section 3, the proposed model is discussed. In Section 4, proposed system's results are analyzed. In Section 5 system performance is analyzed and Section 6 concludes the proposed system and future work of system.

2. RELATED WORK

There are number of people have done their research on trust based models in web services. Some of them are as follows.

Bayesian Network based trust and reputation model [Hien et al 2010] for web service selection approach has three sources for trust calculation such as reputation, QoS monitoring and direct experience of consumer. In this model, author tried to overcome some earlier limitations by integrating the mentioned sources to find the trust value.

Stefania Galizia et.al. (2007) [3] presented a trust model for accessing web service. It follows Trusted Third Party based approach for the classification of the web services with the help of Internet Reasoning Service tool.

Surya Nepal et al. (2010) [4] developed a fuzzy based trust management framework for web service. Initially, they developed a data model based on consumer views on QoS attributes that evaluates the reputation of services. Secondly, they proposed the fuzzy based linguistic query model to parse the requested query to evaluate by different query processing algorithm. They have not addressed some issues such as trust bootstrapping, propagation, retaliation, reciprocation and dishonest or biased ratings.

Wang Meng et al. (2009) [5] proposed a Dynamic Trust Model which is based on recommendation credibility. They suggested a method to differentiate honest and dishonest recommendation and adjust the trust values dynamically.

Kai Wei Shaohua Tang proposed a multi layer trust computation model based on direct search in which service providers need to compute and control the trust of users

Vivekananth et al (2010) [7] proposed a trust evaluation model based on behavior which describes the consistency of behavior and focused on behavior of entities in different contexts. A penalty factor is levied for malicious behavior. The trust factor of two entities may be depending on penalty, time and context. The penalty factor ranges between 0 and 1. A threshold value is used and if the total trust value is greater than threshold value then the resource is will be allocated.

Shangzhu Jin et al. (2010) [8] proposed a model in which service requester trust value is calculated based on based feedback and time decay. It fails to calculate the new user trust value.

Tie-Yan Li et al. (2003) [9] proposed a trust model and trust metrics evaluation algorithms. There are two levels the upper level specifies the trust relationships among Virtual Organizations in a distributed manner. The lower level specifies the trust values in a grid domain.

Wu Xiaonian et al. (2009) [10] tried to quantify the entities trust according to the entity's behaviors. This behavior trust computation model is based on risk evaluation. This model also features identification of asset, threat and trust.

Shashi Bhanwar et al. (2009) [11] proposed an access control model based on trust by determining reputation and trust of the domain on the basis history of past transactions and rated feedback value.

Srivaramangai et al. (2010) [12] proposed an access control model discussed systems based on reputation can be used in web services to enhance the reliability of transactions. Reliability can be achieved by establishing mutual trust between the requester and the provider.

Shunan Ma et al. (2005) [13] proposed multifactor based access control model for calculating trust. This model also includes multi-factors trust computation, permission mapping and feedback module.

Sapna singh et al. (2010) [14] proposed trust based model, the privilege for defining the access levels are given to the publisher where certain constraints will be defined on each information object being published by the publisher in order to establish a desired trust level for the subscriber to get access to the information of his interest.

Cesar Ali et al. (2010) 15] proposed a new trust model to access the web services based on context and role of the services requester. Here too they failed to handle new user trust value effectively.

The Existing trust based models are not handled new users trust level effectively and consider very few factors to compute the trust for service requester and the service requester transaction would be considered as failed if server errors occur. To avoid these kinds of issues, proposed work compute trust value using multifactor such as such as network conditions, frequency of access, timeout, success rate and

Journal of Theoretical and Applied Information Technology
20th June 2014. Vol. 64 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

failure rate. Also proposed system detects and prevents IP address spoofing and SQL Injection attack to strengthen the initial authentication system. Also proposed system needs to focus on categorization of service requesters according to the trust value. Then only those who have high trust value will be served by the system effectively.

3. PROPOSED WORK

The following Figure 1 shows proposed access control model components and its relationship with each other.

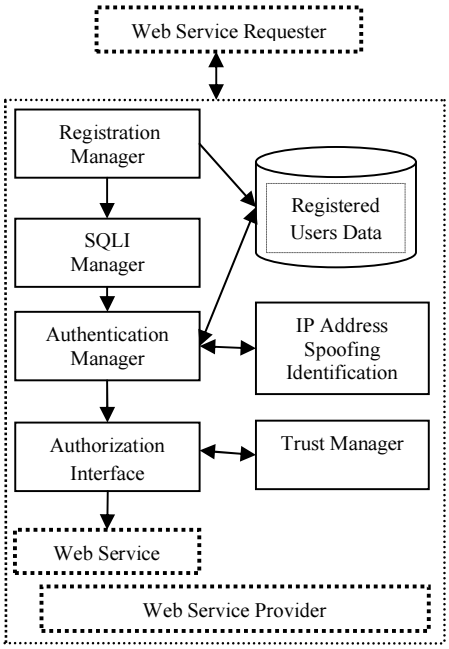

Figure 1: Overview of Proposed System

This system architecture has been divided as two major parts 1. Web service requester who accesses the service 2.Web service provider provides the service and manages the trust value. In the web service provider part SQL Injection (SQLI) Manager, Authentication Manager, Authorization Interface and Trust Manager are the components which are involved in the service access control process.

3.1 Registration Manager

It has the following components.

3.1.1. Registered Users Data: It is a data storage area of the authentication system which maintains the details such as user id, password, Remote IP address of last request, date and time of last request, previously requested page, number of hits, Number of status codes such as '200','400' and '408' requested URL and other personal details of registered service requesters. These data will be used by later authentication processes.

3.1.2. User Registration: If there is request for registration then new requester of service has to register their personal details with their IP Address with service provider. During the registration process, requesters will be providing unique user id and password to access the web service. Consequently new requesters' details such as user id, password, and IP Address will be stored in the database. If the requester is already registered, it pass request to SQL Injection Manager.

3.2 SQL Injection (SQLI) Manager

User registration module passes the SQL request to the SQLI Manager [18] which detects and prevents SQL Injection at the provider area using following two components.

3.2.1. XML Generator: In this process, the intercepted SQL Query is analyzed and XML file is created by obtaining input parameters from SQL Query to detect possible injections [20]. This XML file is then further used for validation in order to detect vulnerabilities.

Consider the SQL query for login mode: *select * from user_Info where username='John' OR 1=1 - -'password=''*. The below Figure 2 shows the sample XML file that would be generated after the above SQL query is intercepted which consist only the input parameters that were given as user inputs from the client application. This file will be used for validation in order to find if any injection is present.

```
<? xml version="1.0" encoding= "utf-16"?>
<sqlxper>
<inp1>John</inp1>
<inp2>'1' ='1</inp2>
</sqlxper>
</xml>
```

Figure 2: Sample XML file for the SQL Query

3.2.1. XSchema Validator: This validation process is to identify the injected parameters with the help of XSchema and the schema is a generalized metadata which define structure and type of user input.

```
<?xml version="1.0" encoding= "utf-16"?>
<xs:element name="inp1">
<xs:simpletype>
<xs:restriction base="xs:string">
<xs:pattern value="[a-zA-Z0-9]>
<xs:restriction>
```

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

```
<xs:simpletype>
<xs:element>
<xs:element name="inp2">
<xs:simpletype>
<xs:restriction base="xs:string">
<xs:pattern value="[a-zA-Z0-9]{8}>
<xs:restriction>
<xs:simpletype>
<xs:element>
<?xml>
```

Figure 3: Sample XSchema for XML Validation

So in this approach, a well defined XSchema is defined for detecting possible injection characters in the input provided by the user as in Figure 3.

The XML file generated by the XML generator, which consists of the user inputs, is now validated with a well defined XSchema. If the validation succeeds then injection is not present in the SQL Query and the operation is allowed to move on next authentication process with passing IP address and user id of requester. In case of failure, process confirms the SQL injection through query.

3.3 Authentication Manager

3.3.1. Data Filtration: After validating SQL Injection, data filtration [21] is the process in which data such as user id, password, Requested URL, IP Address of service requester who currently claims the service from service provider are filtered from the server log file and pass the data to the user authentication process.

3.3.2. IP Spoofing and origin Identification: This process supports the user authentication to identify IP address spoofing by using Ingress packet filtering. Ingress packet filtering checks close to the source whether the address belongs within an assigned network range [22]. If the IP address spoofing is not present then access model proceed to user authentication process.

3.3.3. User Authentication: In this process, actual authentication is takes place by analyzing and retrieving details from the web server log. Once it receives server log details of requesters, it retrieves the requesters' previous history from the registered user data based on user id. Authentication process [19] is described as three cases in the algorithm given in Figure 4.

```
UserAuth (uid, pwd, ipaddr, reqpage)
{
Case 1: If uid and pwd are valid & IP address
                is registered and not spoofed
   Step1: Allow to access authorization
          Interface.

Case 2: If uid or pwd is invalid & IP address
                is registered & not spoofed
   Step 1:Check if requested page is visited &
          Number of hits > threshold value &
       Previous bad request < threshold value.
          If yes
             print to enter correct username,
             password
          Else
             Deny to access web service.

Case 3: If uid or pwd is invalid & IP address
                is not registered
   Step 1: Deny to access web service.
}
```

Figure 4: User Authentication Algorithm

Once the user is authenticated by checking SQL Injection and IP Address spoofing, authorization interface will be called by passing user id.

3.4 Authorization Interface and Trust Manager

Authorization interface component decides whether to allow or deny the accessing of web service by evaluating the trust of web service requester. It uses trust manager [21] to evaluate the trust value of requesters dynamically. In the proposed access model, the trust value range is assigned from 0 to 10 and threshold value of trust is 0. If registered requesters trust level is greater than threshold value then requester will be allowed to access the web services otherwise service access will be ignored. The working principles and components of trust manager are explained as follows.

3.4.1. Trust Decision Point (TDP): Trust decision point is the part of trust manager. Based on current trust value TDP takes initial decision of whether service requester allowed or denied to access the web services. This initial trust level verification makes system more efficient by avoiding calculation of trust level for the requester unnecessarily.

In the case of new requester, system assigns initial trust value of 0. If the requester is

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

new to the system, TDP will not send request to calculate the trust value to TMP and TNP but allows the requester to avail the service from the provider and stores the transaction details.

3.4.2. Trust Management Point (TMP): Trust management Point (TMP) is another part of trust manager. The basic idea is that after the initial trust level verification done by TDP, Trust Management Point (TMP) dynamically evaluates trust value using the following factors which are retrieved from the registered user database.

1. Success Rate (*St),* this specifies number of successful transactions of a service requester in access control for a specified time period. This time period can be assumed by service provider.

2. Failure Rate (Ft), this Specifies number of failure transactions of a service requester in access control for a specified time period.

3. Frequency of Access (Af), it has the value of frequency of accessing the service by the requester. It has specified threshold value which can be assigned by service providers. If the requester's Af is lower than threshold level, requester can be defined as lazy requester.

4. *Time-out (To),* this value specifies number of time-out occurs during resource access. This factor is used to recognize honesties in request based on its threshold value.

5. *Average Time (At)*, average time spent during service access in access control model. This factor also used to recognize honesties in request.

This part of trust manager also classified service requesters into four types according to their history of behavior in the system. They are 1.Honest and active 2.Honest and inactive 3.Dishonest and active 4.Dishonest and inactive.

Here is an example for the four types of user introduced above; if Time-out (To) and Average time (At) are greater than its defined threshold value (Vo), then requester is considered as honest else dishonest requester. If frequency of access (Af) is greater than its threshold level, then the requester is considered as active requester else inactive requester. Service provider can assign their own threshold value for the above mentioned factors such as To, At and Af. Trust value calculation will be varying according to user's type. The proposed model has the following trust calculation techniques.

Type 1: If the requester is identified as honest and active, then requester's decayed trust value (dt) will be increased as follows:

$$dt^{t+1} = (dt^t + (St+Af)*dt^t) / (St+Ft) \quad (1)$$

Type 2: If the requester is identified as honest and inactive or dishonest and active, then requester's decayed trust value (dt) will be decreased as follows:

$$dt^{t+1} = (dt^t - (St+Af)*dt^t) / (St+Ft) \quad (2)$$

Type 3: If the requester is identified as dishonest and inactive, then requester's decayed trust value (dt) will be decreased by subtracting penalty factor as follows:

$$dt^{t+1} = (dt^t - (St+Af)*dt^t) / (St+Ft) - pf \quad (3)$$

In the above formula, service provider can assign penalty factor. This system assumes penalty factor as 0.5. The algorithm for Trust Management Point (TMP) expressed in Figure 5.

```
dt (St, Ft, Af, dt, To, At)
{   If (To>Vo and At> Vo and Af > Vo)
      //Honest and Active Users
       Calculate dt^{t+1} by using formulae (1)
  Else if  (To>Vo and At> Vo and Af < Vo)
     // honest and inactive /dishonest and
       active Users
       Calculate dt^{t+1} by using formulae (2)
  Else if  (To<Vo and At< Vo and Af < Vo)
     //User is dishonest and inactive Users
       Calculate dt^{t+1} by using formulae (3)
}
```

Figure 5: Algorithm -Trust Management Point (TMP)

After calculating trust value, TMP checks for server failures during the previous transaction and user category. If there are server failures or user category is platinum then TMP sends request to TNP for negotiating the trust value so that honest users will not be punished. To reach the platinum user class, service requesters should maintain maximum value of 10 for 1 month.

3.4.3. Trust Negotiation Point (TNP): TNP negotiates the trust value of the eligible service requesters. TNP uses the following factors to negotiate the trust level.

1. Server Error rate (Se), this value uses in the trust negotiation point also to know about number of time-out or discontinuation of request occurs due to Server problem. Due to server error sometimes the trust level of honest service requester may be decreased or punished. To avoid this issue TNP checks whether the previous service accesses has any communication failure due to server or not.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

2. *User class (Uc)*, used to identify the class of user is platinum. This factor is used to reward the honest users.

So if there is server error or user maintains platinum user class, TNP negotiates the value by adding Negotiate Factor (Nf) of 0.5 with the value send by the TMP. Negotiation process will be negotiating honest and active users for 3 times to encourage their honest and activeness. Finally TNP send final trust value to TDP which sends the trust value to authorization interface.

Finally Authorization Interface decides whether to allow or deny the requester to access the web services based on the trust value return from the Trust Manager.

4. RESULT ANALYSIS

In order to prepare experimental data sets and to implement all the functionality needed by the proposed system from client registration to service request, an application-independent framework was developed. On top of this framework, an e-library web application was implemented to test proposed system operability and performance. At low level of framework a web service for search and download the books was developed. The e-library application was simulated in LAN environment for more than 1 month and about 200 registered users were involved in the experiment. Finally extract of experimental result was tabulated and performance was analyzed as follows.

Experiment 1

In the experiment 1, system was checked for how effectively it handles fraudulent victims based on Ingress filtering technique for specific days. So the system was deliberately assigned specific number of attackers to spoof IP address. The above experiment resultant data have been illustrated in Table 1 and concluded that ingress filtering technique effectively identified all IP address spoofing and prevented the victims correctly.

TABLE 1: Denial of Fraudulent victims

Access Frequency (Af)	Trust Value Based on Af	Trust Value Based on Af with To, At	Trust Value Based on Af with To, At, Pf
10	0.33	0.32	0.27
20	0.37	0.36	0.31
30	0.4	0.39	0.34
40	0.42	0.41	0.36
50	0.47	0.47	0.42
60	0.51	0.5	0.45
70	0.55	0.54	0.49
80	0.59	0.58	0.53
90	0.62	0.61	0.56
100	0.67	0.66	0.61

Experiment 2

In experiment 2, consider the values [12] of dt=0.66, Ft=100, St = 80 and frequency of access (Af) ranges from 10 to 100. Frequency of access (Af) threshold value for a specific period is assumed as 100. Here To, Af are assumed that they are less than their threshold value, Penalty factor (Pf) =0.5. Resultant trust values based on frequency of access (Af), Time out (To), Access Time (At) and Penalty Factor (Pf) are listed in Table 2.

TABLE2: Effect of multifactor on Trust values

Day(s)	No. of Victims	Day(s)	No. of Victims
1	22	7	20
2	24	8	24
3	26	9	23
4	30	10	28
5	41	11	38
6	41	12	38
7	35	13	33

From this table, we can conclude that trust value is increased as Af value is increased. It means that if the user is dishonest and inactive, trust value will be affected. The values from this table have been taken for the further analysis done below and proved this proposed system is restricting malicious user efficiently. The performance graph of above values is shown below:

Journal of Theoretical and Applied Information Technology
20th June 2014. Vol. 64 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

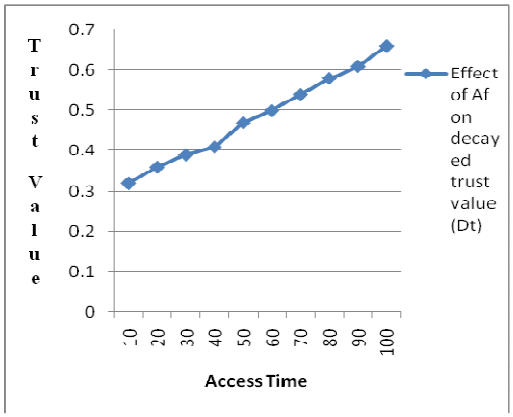

Figure 6: Effect of Af on decayed trust value (Dt)

From the graph in Figure 6, we can see that the trust value of malicious requesters will be steadily decreased if the requester's frequency of access (Af) value is lower than threshold value otherwise trust value will be increased. Hence we can conclude that the lower the value of Af is the faster decline for the value of dt, which effectively encourages the requester to provide honest access and avoid malicious access. However, those malicious behaviors cannot be forbidden simply by Access frequency.

Experiment 3

In experiment 3, Trust values were calculated using the formula and listed in Table 3 based on values of success rate (St) and failure rate (Ft) of the user. Also values of frequency of access (Af), Time out (To), Access Time (At) and Penalty Factor (Pf) are considered to calculate the value.

TABLE 3: Effect of St, Ft and multifactor on trust value

St	Ft	Af	dt Based on St,Ft, Af,To	Dt Based on Af ,To, At	Dt Based on Af,To, At, Pf
10	50	10	0.28	0.26	-0.23
20	40	20	0.56	0.54	0.04
30	30	30	0.84	0.82	0.32
40	20	40	1.12	0.92	0.42
50	10	50	1.39	1.19	0.69

In Table 3, Trust value (dt) based on St, Ft, Af, To and Pf was calculated and listed above. From this we can conclude that when success rate is increased, trust value also increased otherwise trust value will be decreased.

The graph in Figure 7 shows the changes in trust value according to multifactor such as St, At, Af, To and At. This graph has been drawn based on 3 sets of data 1. Based on St, Ft, Af and To 2. Based on St, Ft, Af with To, At 3. Based on St, Ft, Af with To, At, Pf. This graph concludes that dishonest and inactive requester's trust value is steadily decreased when trust value is calculated using the multifactor such as St, Ft, Af with To, At, Pf. Also finds that if requester is honest and active value grows gets increased. After including Penalty factor, dishonest and inactive user trust value will be degraded vastly.

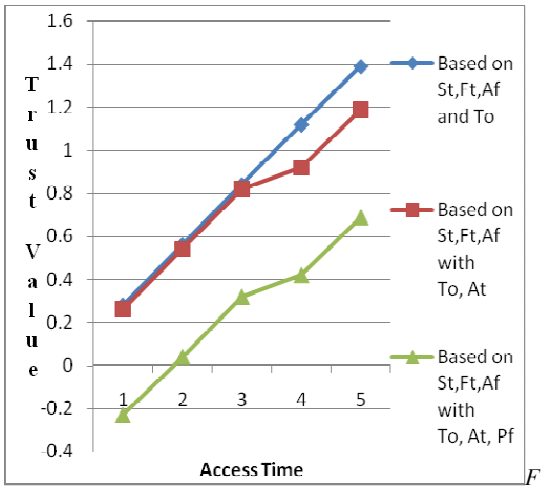

Figure 7: Effect of St, Ft with Af,To and Pf on trust value

However, Honest and active users may be considered dishonest and inactive user because of server failures. To avoid this problem punished requester trust value will be added with Negotiate factor (Nf) so that it will encourage requesters. In our system Negotiate factor (Nf) was set as 0.5. Also if requesters reach maximum value of 10 and maintain the same value for 1 month, they will be considered as platinum user. Negotiation process will also be done for these users for 3 times to encourage their honest and activeness.

5. COMPARISON AND PERFORMANCE ANALYSIS

This section presents a comparison of our approach with existing web services access control models. Table 4 depicts the characteristics and features supported in the proposed approach and existing models. The following comparison reflects that our approach uses a mechanism to assign access percentile to web services and uses it to make access control decisions. Since it detects

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

SQL Injection and IP Address spoofing initially the number of times the validation is to be carried for granting access to resources is reduced which results in reduced authorization work during serving of access requests

Finally the performance analysis of proposed system [13] was conducted with existing traditional access control model such as role based access model and attribute access control model and result is shown in Figure 8.

Table 4: Comparison Of Proposed And Existing Model

Features	RBAC	ABAC	Proposed Approach
Support for distributed environment	Yes	Yes	Yes
Trust level assignment to web services	No	No	Yes
Reduced authorization work for access requests	No	No	Yes
Detect SQL Injection	No	No	Yes
Detect IP Address Spoofing	No	No	Yes
Policies specification & Maintenance	Simple	Complex	Simple

Performance analysis says that how efficiently all the access models are restricting malicious users or behavior. The analysis concluded that proposed model restricts more users than traditional access control models such as role based and attribute models.

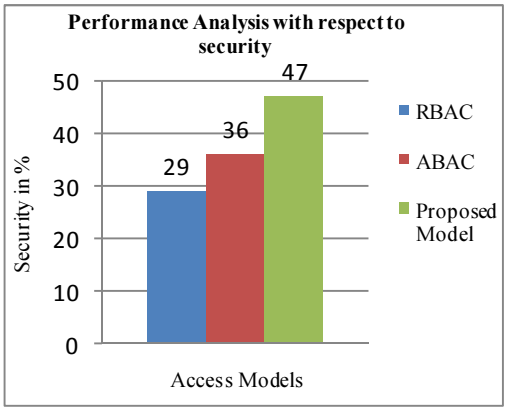

Figure8: Performance Analysis with Traditional Access Mdoels

6. CONCLUSION AND FUTURE WORK

In the proposed system, the impact of multifactor such as Access frequency, Time out, and Access time on trust value to access web service are discussed. Also the system checks SQL Injection and IP address spoofing. As a result, the security of web services gets improved and proposed model is able to encourage requesters to take part in access control actively and honestly. The system results discussion and performance analysis concluded that the model's ability to restrict the malicious users or behavior also better than traditional access models such as role-based model and attribute access model.

In future, system will explore to check the user's authentication more effectively. System will sketch to separate the users according to their trust values so that honest and active users will get more benefits. Further research of the proposed system will lead to complete and more practical solution to manage trust level in access control.

REFERENCES

[1] R. Joseph Manoj , Dr.A.Chandrasekar " An Literature Review on Trust Management in web services access control", International Journal on Web Services Computing, Vol 4; No 3, 2014, pp.1-19.

[2] Hien Trang Nguyen, Weiiang Zhao, Jian Yang, "A Trust and Reputation Model Based on Bayesian Network for Web Services", 2010 IEEE International Conference on Web Services.

[3] Stefania Galizia, Alessio Gugliotta and John Domingue," A Trust Based Methodology for Web Service Selection", International Conferences on Semantic Computing, 2007.

[4] Surya Nepal, Wanita Sherchan and Athman Bouguettaya,"A Behaviour-Based Trust Model for Service Web, IEEE international Conference on Service Oriented Computing and Applications,2010

[5] Wang Meng; Hongxia Xia; Huazhu Song, "A Dynamic Trust Model Based on Recommendation Credibility in Grid Domain", International Conference CiSE.

[6] Kai Wei, Shaohua Tang, "A Multi-level Trust Evaluation Model based on D-S Theory for Grid", International Conference CIS '2009.

[7] Vivekananth.P, "A Behavior Based Trust Model for Grid Security", International

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Journal of Computer Applications, Volume 5– No.6, 2010.

[8] Shangzhu Jin, Jun peng Access control for web services based on feedback and time decay" Proc 9th IEEE International Conference on cognitive Informatics, 2010.

[9] Tie-Yan Li, Huafei Zhu, Kwok-Yan Lam, "A Novel Two-Level Trust Model for Grid". ICICS 2003.

[10] Wu Xiaonian, Zhang Runlian, Zhou Shengyuan, Ma Chunbo, "Behavior Trust Computation Model Based on Risk Evaluation in the Grid Environment", WRI World Congress WCSE

[11] Bhanwar, S.; Bawa, S (2009), "Establishing and Evaluating Trust in a Grid Environment", 10th International symposium ISPAN'09.

[12] Renagaramanujam, Srinivasan and Srivaramangai P . "A Comprehensive Trust Model for Improved Reliability in Grid.", International Journal of Computer Applications Volume5- No.7, 2010.

[13] X. Wang, J. Luo, A. Song and T. Ma, "Semantic Access Control in Grid Computing". Proc. 11th International Conference on Parallel and Distributed Systems. 2005.

[14] Sapna Singh, Archana Puri, Shiksha Smreti Singh, Anurika Vaish, S.Venkatesan," A Trust Based approach For Secure Access Control In Information Centric Network".

[15] Cesar Ali, "CATRAC: Context Aware Trust and Role based Access Control for composite web services" 10th IEEE international Conference on computer and information technology (CIT 2010).

[16] Shanshan Song, Kai Hwang, and Mikin Mcwan, "Fuzzy Trust Integration for Security Enforcement in Grid Computing" NPC '04, LNCS 3222, pp. 9-21.

[17] R. Joseph Manoj , Dr.A.Chandrasekar and M.D.Anto Praveena, " An Approach to detect and tautology type SQL injection in web services based on web services based on XSchema Validation", International Journal of Engineering and computer science, Vol 3; No 1, 2014,pp.3695-3699.

[18] R. Joseph Manoj and Dr.A.Chandrasekar, , "An Authentication System of web services based on Web Server Log Analysis" International Journal of Engineering and Technology, Vol 5; No 6, pp.4786-4793.

[19] R. Joseph Manoj and Dr.A.Chandrasekar, "An Access control model of web services based on multifactor trust based management, International Review on Computers and software", Vol 8; No 10, 2013,pp.2460-2466.

[20] Shanmughaneethi.V R, Ravichandran, and S.Swamynathan"PXpathV: Preventing XPath Injection Vulnerabilities in Web Applications" International Journal on Web Service Computing 2.3.

[21] Akram Alkouz and Samir A. El-Seoud,, "Web Services Based Authentication System for E-Learning ",International Journal of Computing & Information Sciences" Vol. 5, No. 2, August 2007 ;PP-74-78

[22] Indrani Balasundaram, Dr. E. Ramaraj "An Approach to Detect and Prevent SQL injection Attacks in Database Using Web Service", IJCSNS International Journal of Computer Science and Network Security, Vol.11. No.1, 2011.