Designing a Novel Two-Tier Authentication Algorithm for Web Service Architecture

M. Milton Joe[1], B. Ramakrishnan[2] and Resul Das[3]
[1]*Department of Computer Science, St. Jerome's College, Nagercoil, Tamilnadu, India.*
[2]*Department of Computer Science and Research Centre, S.T. Hindu College, Nagercoil, Tamilnadu, India.*
[3]*Department of Software Engineering, Firat University, 23119, Elazig, Turkey.*
m.miltonjoe@gmail.com

***Abstract*—Web pages are secured by one-tier security constraints based on username and password. This one-tier security module is the only way to protect the web pages from hackers. However, the one-tier security constraints on web service architecture have several flaws. It can be hacked from outside without notifying the authorised user. Further, hackers can easily obtain the username and password, which is entered on the web pages to login for further actions. When the system is connected to the internet, the system can be hacked to get the username and password that can be done by monitoring user's keystroke from a remote location. Considering the credentials of one-tier security constraints are being hacked, this paper analyses these flaws and modelled two-tier security constraints to secure the web service. In the proposed architecture pattern, recognition mechanism is used to authenticate the user. The pattern recognition architecture displays at dynamic locations for each refresh of the web page, which prevents the proposed two-tier architecture from being hacked as well as guessing attack.**

***Index Terms*—Web; Hacking; Authentication; Security; One-Tier; Two-Tier; Internet; QoS; Pattern Recognition.**

I. INTRODUCTION

Web service is a way of communication established among electronic devices with the help of World Wide Web [1]. The functionality of web service architecture is similar to a software function used with a network address to access the service over the World Wide Web [1]. Web 2.0 yields a better communication performance in the web service architecture after much research development [2, 3]. The development of Web 2.0 results in internet users to post review contents, blogs, social networks, photos and videos in the web pages [4, 5, 6]. Additionally, useful user-friendly applications added in the web sites along with the development of Web 2.0 have become famous among the hackers to avail the content in an unauthorized manner [4]. All the applications developed under the Web 2.0 technologies are easily accessible, hence they can be hacked with malicious code easily [4].

In response to the rapid development in Information Technology (IT), wireless communication is now able to transfer message from anywhere, anytime using any device [7, 8]. This technological growth in wireless communication [9] laid the foundation for the growth of Service Oriented Computing [7, 10]. Web service architecture makes possible programs written in different languages on different platforms to communicate with the standard protocols [10]. Web pages are designed with HTML, JavaScript, and CSS. These languages impose the addition of multiple elements with no restrictions, making it easy for unrelated web pages to access the contents [11, 12]. These design strategies have led to possible vulnerabilities of well-known securities, such as cross-site scripting (XSS) [13], client-side attacks [14] and cross-site request forgery [15, 16]. Security plays a vital role in the web service architecture and the architecture must be capable of providing efficient security mechanism to the users.

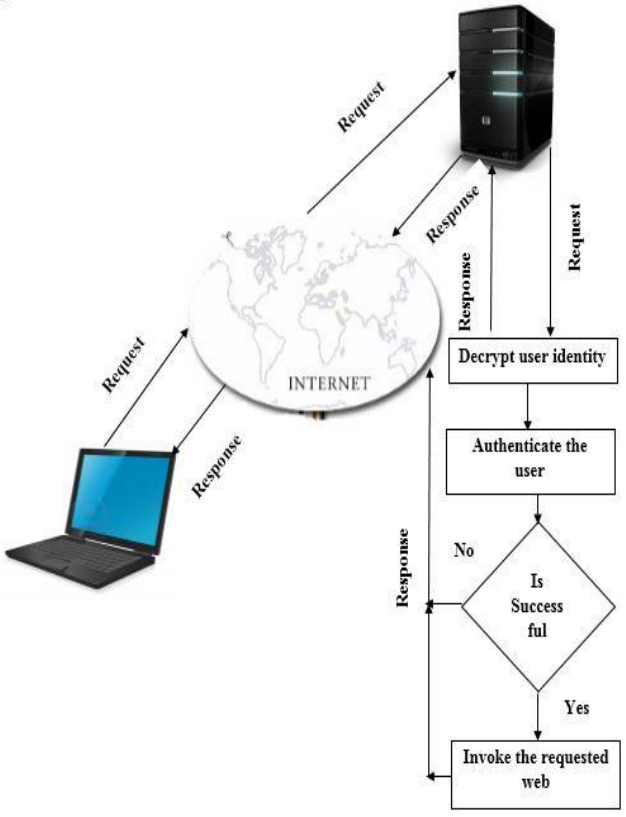

Figure 1: Web Service Architecture

Figure 1 represents the concept of web service architecture. As depicted in Figure 1, the client machine sends the request for a web page to the web server through the internet. As soon as the request is received by the web server, it decrypts the user credentials for authentication. Once the user is authenticated successfully, the requested web page is invoked and the response is sent to the client. If authentication fails, the

error message is sent as a response to the client. This web service architecture must be secured for any transaction between client and server.

This web service architecture is protected with only a username and password, and these username and passwords can be captured through various malicious software, such as key loggers and mouse loggers. The real problem for researcher is the prevention of hacking the username and password of a user. This paper implements a novel authentication mechanism, which prevents the unauthorised access in web service architecture. In the proposed modelling, pattern recognition was used to authenticate the user after a successful authentication with valid username and password. The proposed pattern mechanism was displayed at dynamic locations for each login to a particular web account. Mouse over event was used to draw the pattern in order to prevent the authentication process from being hacked. The detailed description of the proposed mechanism is illustrated in the following sections.

II. Related Work

A. Password Based Authentication

Every user in web service architecture is authenticated based on a password [17, 18, 19]. However, this password or PIN type of validation mechanisms has many limitations [20]. Passwords can be modelled with characters or alphanumerical characters, and this password could be used during the authentication process. Password guessing attack and password cracker programs are used to identify the password of a particular user [17]. The modulation of a password with purely characters is obviously less secure and it can be identified through Brute force attack or Dictionary attack [21].

B. Privacy Question based Authentication

This authentication mechanism is the process of authenticating a user through privacy question [15, 18, 19]. Users of web service architecture are allowed to set the privacy-based question and answer at the time of account creation. However, this question and answers cannot be remembered after several years [22]. This privacy questions are ambiguous, guessable and easily attackable through various attacks [22].

C. Virtual Keyboard Authentication

Another type of authentication mechanism used to prevent the hacking of the username and password of a user is virtual keyboard authentication. This authentication mechanism is used by most of the financial organisations to prevent username and password from being stolen. This authentication model replaces the physical keyboard exists in a system and provides the virtual keyboard on the screen [23]. On this keyboard, the alignment of keys is randomly arranged during each refresh and the location of the keyboard is static. Users can enter the username and password with this virtual keyboard by clicking the appropriate keys through the mouse clicks [23]. Though this authentication mechanism replaces the physical keyboard, it does not prevent the hacking of user's username and password. This authentication mechanism has a number of flaws as listed below [23]:

- The location of the keyboard and the alignment of keys are captured in screen shots.
- Mouse clicks are used to enter the username and password, which can be identified through mouse logger
- The display location of the virtual keyboard is static.
- The mouse movements are stored in the mouse logger, which can be matched with the keyboard to identify which keys are clicked, since the keyboard remains in the static location.

The observations obtained from the previous authentication mechanisms have shown that present authentication mechanisms have limitations and a strong authentication mechanism must be developed. Hence, high level security model can be applied in all possible domains of web service architecture. The following section discusses the efficient security deployment at various domains.

The following sections are categorised as follows: Section III represents security levels at various domains in web service architecture. Section IV illustrates the existing one-tier security architecture and its various limitations. Section V defines the proposed two-tier security architecture and how it prevents the hacking of username and password.

III. Web Service Security Model

Security mechanism in web service architecture can be applied at three different levels [7, 24].

- Platform/ transport-level (point-to-point) security.
- Application-level (customer) security.
- Message-level (end-to-end) security.

A. Platform/ Transport-level (point-to-point) security

Platform/transport-level security provided between web service clients and web service server is illustrated in the Figure 2. In the point-to-point (Platform/Transport-level) model, the client requests for a service in XML format to the web service architecture and the requested service is not encrypted by the client. However, when the entire message or data reach the transport layer, the network encrypts the message to provide a secured transport at the platform/ transport-level. The Internet Information Server (IIS) provides the basics of the authentication and the message integrity, while the message confidentiality is provided by the Secure Sockets Layer (SSL).

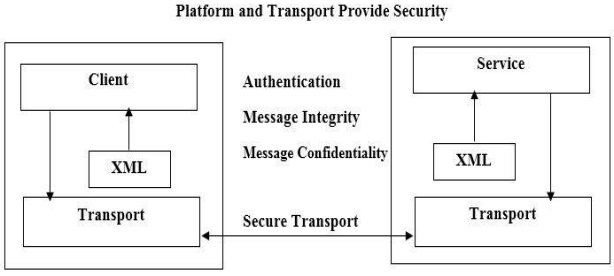

Figure 2: Platform/ Transport-level (point-to-point) Security

B. Application-level Security

In the application-level security model, the security constraints are controlled by the application domain itself and the detailed description is shown in Figure 3. Application-level security model makes use of simple object access protocol (SOAP) to pass the user's identities to the web service architecture for authentication. The most common way is to pass the username as a ticket in the SOAP packet header. This application level security domain can encrypt the data when it is necessary, and it requires a secured key storage. The programmer or the developer must have detailed knowledge of the cryptographic system to provide the security constraints at the application domain.

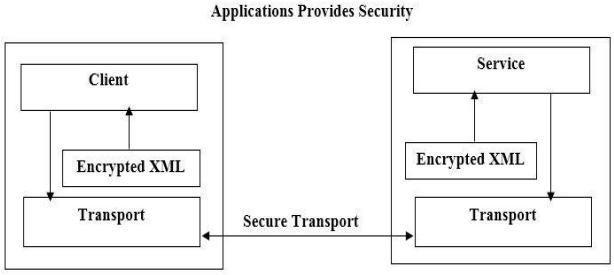

Figure 3: Application-level Security

Figure 4 provides the working principle of the message-level security in web services. The most powerful and flexible message-level security model is availed with the XML for the web service architecture [25]. SOAP messaging service provides data integrity, confidentiality and message authentication. In message-level security, any transport medium can be chosen for transmitting the message and within the transport of the credentials, digital signatures, and the messages can be encrypted for a secure transmission [26].

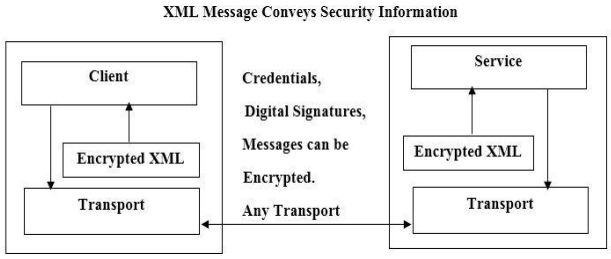

Figure 4: Message-level (end-tot-end) Security

The application level security is considered as the research parameter. All the applications in the web service architecture are protected with two user credentials. The hackers can obtain these user credentials in a numerous way. However, the security at the application level must be tightened to enhance the Quality of Service (QoS) in the web service architecture. The detail description of how the credentials is hacked, and the novel mechanism to protect the web service architecture from the hackers is presented in the following section.

IV. ONE-TIER SECURITY ARCHITECTURE

At present, all web service technologies adopted by the web service architecture comprise of One-tier security architecture for logging into a web page. One-tier security architecture is just a username and password for the particular web page.

Figure 5 illustrates the detailed description of One-tier security architecture in web service technologies. One-tier security mechanism protects the web page with just a username and password. When the requested web page is loaded on the client machine, the user will be requested to enter the login details for successful login to the corresponding web page. If the user enters a wrong username and password, an error message will be displayed to the user. If the user enters the exact username and password, the authentication becomes successful and the corresponding web page will be loaded for further action.

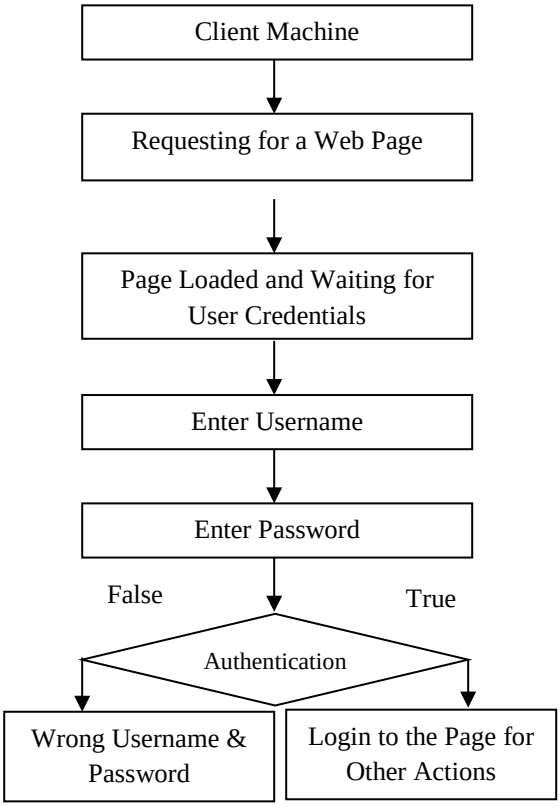

Figure 5: One-tier Security Architecture

A. Limitations of One-tier Security Architecture

A detailed study of the web service mechanism indicates clearly that all web page accounts are protected with only a small piece of information, such as the username and password. All web pages, such as Net banking, E-Mail, Social Networks and so on are exposed to the one-tier security constraints. The web pages must be securable from being hacked by the hackers. The present cryptography technology maintains the securable data transmission in the network environment. That is, the encryption and decryption mechanism protects the data, even though it has been hacked by the hackers.

However, it is possible for the hackers to hack the username and password of a particular user from a remote location. When a user loads a web page, it makes sure that the particular user is connected to the network through internet. Hackers from a remote location, search for the victim or vulnerable user in the network through the internet access point. As soon as the hacker identifies the vulnerable client, the hacker is able to control the client machine by just installing a software from a remote location. Once the software is installed on the client machine, that software identifies and sends the user's login details (username and password) to the hackers. The software that records the user's actions in a machine as listed below:

- Keylogger (Detects the Keystroke)
- Mouse Logger (Detects the Mouse actions & Records it)

B. Keylogger (Detects the Keystrokes)

Keylogger or Keystroke detector is a software, which needs to be installed on a machine. The aim of this software programme is to detect each and every key pressed on the particular machine in a hidden manner. Keylogger monitors the system and records the keystrokes of the system and reports the keys pressed. The report generated by the software clearly notifies what keys are pressed at a specific time, and even the software generates the screenshot of the web pages on a particular machine.

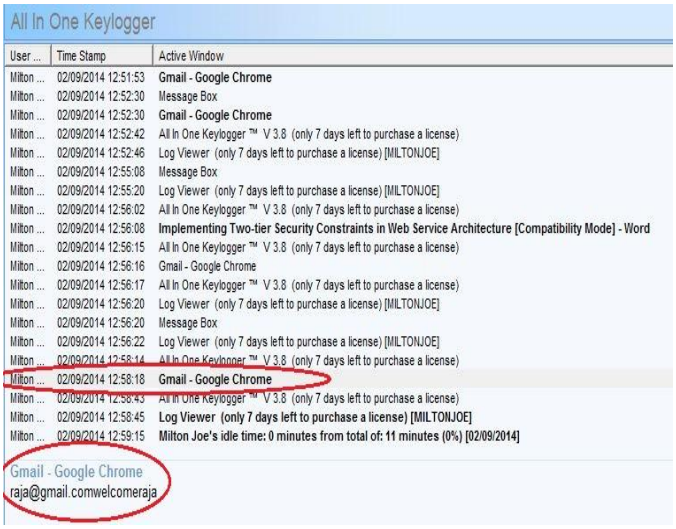

Figure 6: Keylogger Software Records the Keystrokes

Figure 6 represents the keystrokes being monitored by the keylogger software. As shown in Figure 6, when the user opens his Gmail account, the username and the password are recorded. The username (raja@gmail.com) and the password (welcomeraja) entered in the Gmail web application page is clearly recorded in the keylogger software. Once the username and the password of the particular user is being hacked, the hacker can easily get access in the account and the account can be misused widely as the hacker wishes. All the present web technology web accounts provide the one-tier security constraints, such as username and passwords. This one-tier security mechanism has the limitations of being hacked. However, a novel security mechanism is ultimately needed to protect the user account from being hacked in the web service architecture.

C. Mouse Logger (Detects the Mouse actions & Records it)

Mouse logger or mouse recorder is a software that records the mouse actions that are performed on a machine. This software records the clicks of the mouse, such as right click, left click, and mouse movements. The recorded actions can be viewed by the hacker at any time. Even the software, such as mouse logger and keylogger takes the screenshot of the monitor in a periodical interval time.

The above study makes clear that the keystroke and the mouse actions of a system can be recorded. The present web service architecture has one-tier security mechanism, such as the username and the password. The one-tier security mechanism can be easily recorded by the hacker since the mechanism uses the keyboard of the system to enter user's credentials. Hence, the present security mechanism is not securable and it can be hacked by hackers easily. A novel security mechanism must be adopted to prevent the web service applications from hackers. This paper considered the limitation of the present one-tier security mechanism and modelled a novel two-tier security architecture. The detailed representation and working principle of the proposed two-tier security architecture are illustrated in the following section.

D. Identification of Victim in the Network

When the systems are connected in the network, it can be hacked by hackers. Hackers will scan for the vulnerable machine in the network by using various scanning strategies. Once the vulnerable machine is identified, the hackers will install the infection codes in the target machine to take the control over the vulnerable machine. The various scanning methods used are explained below:

- Random Scanning
  - Uniform Scanning
  - Hit-list Scanning
  - Routable Scanning
- Localised Scanning
  - Local Preference Scanning
  - Local Preference Sequential Scanning
  - Selective Scanning

a. Random Scanning

Random scanning is the process of identifying vulnerable machine in the network environment at randomly [27]. The hackers will search for all the IP addresses connected to the network, and then they will choose any machine as the victim. The various scanning strategies come under random scanning are illustrated below:

i. Uniform Scanning

Uniform scanning process is used when the hackers have no knowledge where the vulnerable host resides. That is, no preference is given to the victim machine. All the systems are scanned within the IPV4 address space, and then the target machine is chosen without any preference [27]. This uniform scanning process needs a perfect random number generated to identify the target IP address at random [27].

ii. Hit-list Scanning

The Hit-list scanning process first searches for the all the vulnerable machine within the hit list [27]. Once the hit list victims are infected and then the propagation moves towards random scanning strategy. This hit-list scanning process was introduced by Staniford et.al. [28].

iii. Routable Scanning

Routable scanning strategy within the IPV4 address space needs to enquire each IP address within the whole routable address space [27]. Hence, it has to determine which are the IP addresses are routable within the address space [27].

b. Localised Scanning

The localised scanning process employs the scanning activity within the nearby locations instead of going for random scanning [27]. Here, scanning activity is encountered within the defined locations to identify the victim. The scan strategies come under localised scanning process are listed below:

i. Local Preference Scanning

Vulnerable machines reside in the network anywhere in the world [27]. Hackers may be willing to infect the systems that are available within the hacker's location. In this situation, local preference scanning methodology is used to identify the victims residing within the local area of the hacker.

ii. Local Preference Sequential Scanning

The local preference sequential scanning strategy is completely different from random scanning process. Here all the IP addresses are scanned in the sequential manner, which is one by one from the starting IP address [27].

iii. Selective Scanning

When the hacker wants to infect certain IP address rather than the entire internet, this selective scanning strategy is used [27]. The scanning address space is obviously reduced with those selected IP addresses [27]. All the above described scanning methods are used to identify the vulnerable machines in the network to affect them. There are various ways for a particular system to be hacked in the web service architecture. Hence, a novel procedure is needed to prevent the web service architecture from being hacked.

V. TWO-TIER SECURITY ARCHITECTURE

As discussed in the previous section, a system can be hacked in the network by installing malicious software to identify the user credentials. One-tier security architecture, such as the username and password is the only way to protect the user's account in the web service architecture. However, there are many ways to identify the vulnerable host on the network by employing various scanning strategies discussed in the previous section.

A novel web service architecture is ultimately needed to prevent the host from being hacked. A security mechanism is needed to prevent a complete user account, even though malicious software is installed on a host to identify the user authentication details. That is, although the hacker gets the authentication credentials of a user, such as username and password, the hacker could not login to the particular account with those credentials. Such a strong architecture is needed to enhance the security mechanism in web service architecture.

This paper modelled and implemented a novel Two-tier Security architecture for web service applications, which prevents the user account from being hacked by the hacker in the networking environment. The detailed illustration of the proposed Two-tier Security architecture for web service applications is illustrated below using a neat block diagram.

Figure 7 describes the working data flow of the proposed Two-tier Security architecture. When the web application is loaded, the user will input the user credentials for successful login. Once the one-tier (Username and password) authentication is successful, the user will be authenticated for the two-tier security authentication. If the two-tier authentication is successful, the user will have access to their accounts.

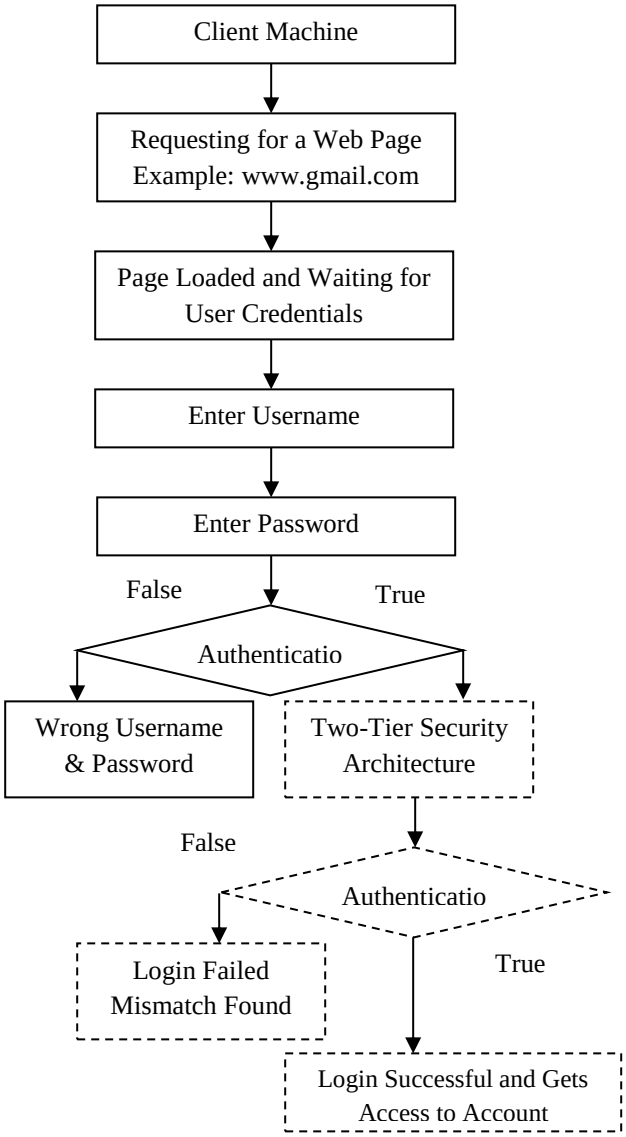

Figure 7: Two-tier Security Architecture

A. Modelling Two-tier Security Architecture

The proposed Two-tier Security Architecture is modelled in the format of 4 * 4 matrix as represented in the Figure 8.

Once the username and the password of the user are authenticated successfully on a web page, the proposed two-

tier security authentication page is displayed on the web page. The user has to fulfil the security parameter to login to the particular account. When the user creates the account in a website for the first time, the user is asked to create the username and password for the particular account. In the same way, the user must be asked to draw the pattern of his / her own. That is, the user has to draw a line to connect the points to make a design, and the design is stored for validating that the user has successfully login to the web account. The mouse over event is used to create the design. When mouse over event is used, the user can draw the pattern, although the pattern is not visible on the web page.

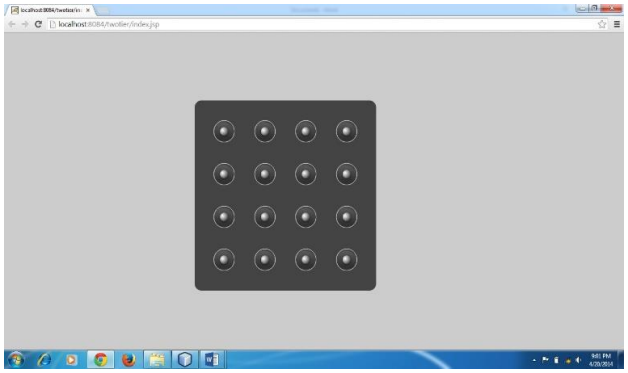

Figure 8: Two-tier Security Architecture

B. Forming the pattern

- Users are advised to create the pattern with a minimum of 10 points.
- It is not wise to draw the pattern in the same row or column.
- Pattern is drawn on the web page to authenticate the user with the mouse action.

The above mentioned security constraints are framed to make it difficult to the hacker, who tries to hack the account using a valid username and password and a guess to the proposed Two-tier security architecture pattern.

C. Mouse Over Event

It is our knowledge that input can be given by either the Keyboard or Mouse. The keylogger software is used to identify the keystroke of a machine, which shows the username and password of a user. In the same way, mouse logger or mouse recorder software is used to identify the right click, left click and mouse movements performed on a system. These hacking software even produce the screen shot of the current desktop of a user's system at a periodical interval time.

The above study shows that keystrokes and mouse clicks, as well as mouse movements can be captured by the hackers. The fundamental aim of this paper is to protect the web service architecture, although the hacker has user's valid username and password. The security is achieved by adopting two-tier security architecture modelled with keystroke and mouse clicks that can be identified by the hackers.

This paper proposes to make use of the mouse over event to authenticate the web service architecture, which cannot be identified by the hackers. The mouse over event does not make any line or any design visible on the web page, when the pattern is drawn. If we make the pattern visible on the web page, then the pattern can be captured in the screen shot.

Once the pattern is visible, the hacker can easily get access into the user account. The aim of the research is to prevent the pattern from being captured by the hackers, and the hacker should not get access into an unauthorised account. As we know that, when screen shots are taken, the mouse pointers cannot be captured in the screen shot image. Hence, the mouse over event is a perfect procedure to authenticate the user in web service architecture. The working principle of the proposed model is: The user has to draw or connect the points, which is created by the user earlier and to ensure that the pattern drawn by the user matches with the database. If the authentication is successful, the user is allowed to login to the particular account. Figure 9 illustrates the working methodology of the proposed model. At the time of the authentication, when the user places the mouse over the container (pattern recognition area), the proposed model becomes active. All the points or dots defined in the model will have predefined numerical values. The predefined numerical values of the dots or points are captured as the user moves the mouse over the pattern dots. Once the user keeps the mouse away from the container, the captured values are matched with the database to check the user's originality.

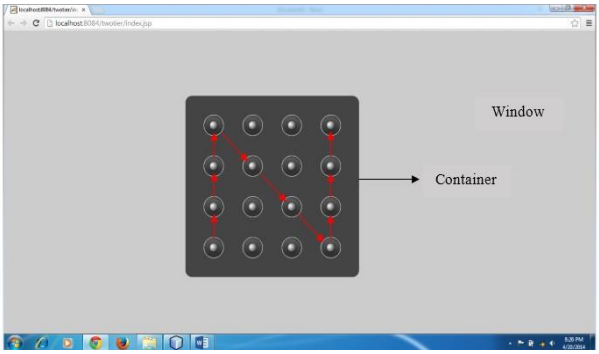

Figure 9 Working Principle of the Two-tier Architecture

D. Dynamic location of the Two-tier Security Architecture

Mouse logger or mouse recorder stores the movements of the mouse, and it can be viewed by the hackers at any time. This software stores only the mouse movements, excluding the web page screen shot, where it has been moved. Knowing the mouse movement action, the hacker can only guess the pattern and the hacker is not sure which row and column the movement has been moved. It is possible for the hacker to identify the pattern by guessing the stored mouse movements again and again.

Hence, to avoid this limitation, the proposed Two-tier Security architecture is displayed on dynamic locations on the web pages for each login. When the authentication matrix design is displayed at various locations for each refresh, the hackers cannot guess the pattern of a particular user. When a webpage is loaded, the pattern is displayed at random location and the position of the pattern's current location is captured. Unless the user attempts to login to a particular account, the pattern will be displayed at the same location regardless of how many times a web page is refreshed. This methodology prevents the proposed architecture from guessing attack.

```
<html>
<style>
  * {
        margin: 0;
        padding: 0;
     }
body{
        background-color:#CCC;
       }
div  {
        float: left;
       }
 #page_Wrapper { }
 .president-table
         {
              border-collapse: separate;
              border-spacing: 40px 40px;
              margin:auto;
              background-color:#444;
              height:260px;
              -webkit-border-radius: 15px;
              -moz-border-radius: 15px;
              border-radius: 15px;
              overflow:hidden;
         }
</style>
</head>
<body onload="dynamic()">
<div id="page_Wrapper">
<%@include file="patternlock.jsp" %>
</div>
<script>
function dynamic()
{
 window_Height = window.innerHeight;
 window_Width = window.innerWidth;
 image_Element=
document.getElementById("image");
 image_Height = image_Element.clientHeight;
 image_Width = image_Element.clientWidth;
 availSpace_V = window_Height - image_Height;
 availSpace_H = window_Width - image_Width;
 moveImage();
 }
function moveImage()
{
 var randNum_V = Math.round(Math.random() *
                 availSpace_V);
 var randNum_H = Math.round(Math.random() *
                 availSpace_H);
 image_Element.style.top = randNum_V + "px";
 image_Element.style.left = randNum_H + "px";
}
      </script>
   </body>
</html>
```

Figure 10 Implementation Code for Displaying Two-tier Model at Dynamic Locations

The malicious software provides the screenshot of the webpage. Further, the movements of the mouse can be captured by the mouse recorder software, excluding the webpage. The hacker may match the screenshot along with the mouse movements to attempt for guessing attack. That is, the hacker can keep on refreshing the webpage to make the pattern appear at a desired location. If the pattern is displayed at the same location, unless the user does not attempt to login regardless of how many times a webpage is refreshed, the hacker cannot attempt the guessing attack. Hence, the proposed Two-tier Security Architecture in web service applications works well with no limitation. The display position of the Two-tier security module is randomly changed for each time when the web page is loaded as depicted in Figure 10. The horizontal co-ordinate and vertical co-ordinate values are changed by random values. When the horizontal and vertical co-ordinate values are changed by subtle values, the display location of the proposed model cannot be guessed by the attacker. Hence, the proposed module prevents the authentication mechanism from guessing attack.

E. Comparison of the Proposed Pattern Modelling with Mobile Pattern Modelling

The proposed Two-tier security architecture makes use of pattern identification to authenticate the user. However, this sort of pattern identification technology is being used in the mobile phones. The proposed modelling is different from the pattern identification used in the mobile phones. The comparison and the advantage of the proposed modelling is illustrated below:

Limitations of Mobile Phone Pattern Modelling:

- The pattern is developed with 3*3 matrix and with 3*3 it is not highly secured.
- The pattern can be easily guessable, since it has 3*3 matrix format only.
- It does not come under the networking environment.
- Display location of the pattern is static.
- It is used to protect the mobile phone access.
- It is useful to individual person only and not to the entire community.
- This pattern modelling can be attacked and hacked by the hackers easily.

Advantage of the Proposed Modelling:

- It uses 4*4 matrix format and it is highly securable.
- Pattern drawing is not visible, so it cannot be hacked.
- Since mouse over event is used, it cannot be hacked in the web applications.
- It comes under the networking architecture.
- Display location of the pattern is dynamic, hence it is prevented from the guessing attack.
- Even though the hacker has the pattern movement, hacker cannot access the account as the mechanism uses dynamic locations to display the pattern.
- Mechanism is used in the web service architecture.
- Entire web service architecture is secured from being hacked by the hackers.

- This mechanism prevents the user profile and account throughout the world.

F. Algorithm of the Two-tier Security Architecture

```
Requesting for a webpage
While (True)
{
Get Username;
Get Password;
If (One-tier Authentication Successful)
Validate Two-tier Security Module
{
Dynamic () // Display the pattern at dynamic location
{
Calculate window height;
Calculate window width;
Obtain the pattern modeling height;
Obtain the pattern modeling width;
Available_space_vertical = window height- pattern modeling height;
Available_space_horizontal = window width-pattern modeling width;
Random_value_vertical= Math.round(math.random()*available_space_vertical;
Random_value_horizontal= Math.round(math.random()*available_space_horizontal;
Display pattern at dynamic location by the values of random_value_vertical & random_value_horizontal;
}
Get the pattern predefined values obtained by mouse over event;
}
If (Pattern matched)
Login to the web application;
Else
Report pattern mismatch;
If (One-tier Authentication unsuccessful)
Report mismatch of username and password;
}
```

Figure 11: Algorithm of Two-tier Security Module

The working principle of the proposed Two-tier Security Architecture is illustrated in Figure 11. As explained in the algorithm, when the one-tier security mechanism is authenticated, the Two-tier security mechanism is authenticated as well. However, the one-tier security credentials can be obtained by the hackers. The proposed Two-tier security prevents the account from being hacked by the hackers, even though the one-tier security architecture credentials are hacked.

In the proposed two-tier authentication model the pattern modelling is displayed at dynamic location for each refresh of the web page. Identification of the dynamic location is generated by random values based on the calculation of the window height, width and pattern modelling height and width, as presented in Figure 11. Modelling the pattern architecture at dynamic location prevents the proposed architecture from being hacked by the hackers and even the hackers cannot guess the pattern of a particular user.

G. Comparison of One-tier Verses Two-tier Architecture

The comparison of one-tier and two-tier security module is explained in Figure 12. As illustrated in Figure 12, the one-tier security credentials, such as the username and password can be hacked easily. However, in the case of two-tier security constraints, although the hacker obtains the one-tier credentials, he/she could not be able to log into the account. Since the mouse over event is implemented in the two-tier mechanism, it cannot be obtained or predicted through keylogger, mouse logger and even through screenshots.

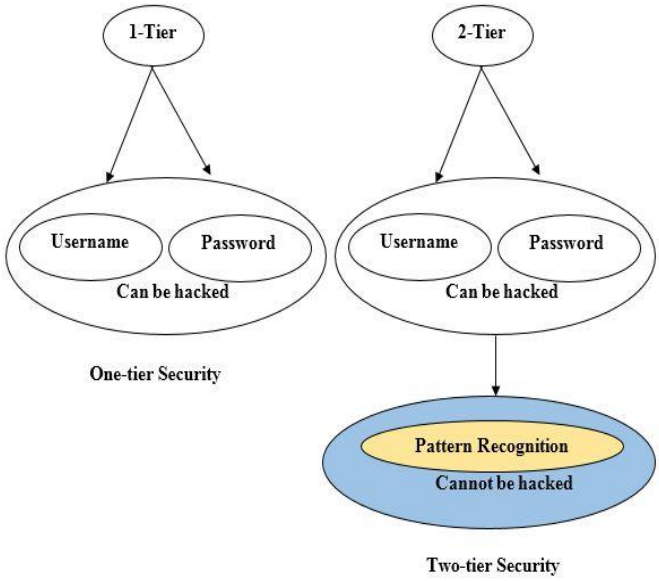

Figure 12: Comparison of One-tier and Two-tier Security Architecture

On the other hand, when additional security mechanism is implemented, a special care must be taken on the usage of CPU. CPU usage (Load) should be very high due to the adaptation of the new security concern. If the CPU load is very high, it is obvious that it will take more time to complete the process. Additionally, when a new security mechanism is implemented, the CPU usage may increase a bit. However, it should not be very high in such a way that it affects the time factor. The CPU usage in the one-tier versus the two-tier architectures are shown below:

CPU usage of one-tier mechanism = 41 %

CPU usage of two-tier mechanism = 41.3%

It can be seen that there is a very subtle difference in the CPU usage, when the proposed model is implemented. Hence, the proposed model can be implemented in the web applications to enhance the security concerns.

VI. CONCLUSION

The complete characteristics of web service architecture are studied in this paper. The study makes clear that all web service architecture applications are protected with one-tier

security module, such as the username and password. However, the keylogger and mouse recorder software could be installed in a machine through the internet to hack client's username and password. Hence, it is crystal clear that one-tier architecture has major limitations, which degrade the quality of service in web service architecture. This paper considered this major limitation and modelled a novel Two-tier security architecture to enhance the security constraints in web service architecture. The proposed model has proven that web service applications cannot be hacked and it prevented attempts of guessing attack by unauthorized users. The implementation of the Two-tier Security Architecture ensures that hackers cannot hack and login to an account of a user even though they obtain the credential of the One-tier Security. By and large, the proposed Two-tier Security model results in effective security constraints in the web service architecture to protect the web service applications.

REFERENCES

[1] Joe, M. Milton, and B. Ramakrishnan. "WVANET: Modelling a novel web based communication architecture for vehicular network." *Wireless personal communications* 85.4 (2015): 1987-2001.

[2] Joe, M. Milton, B. Ramakrishnan, R.S. Shaji "Prevention of Losing User Account by Enhancing Security Module: A Facebook Case", Journal of Emerging Technologies in Web Intelligence, Vol. 5, No. 3, August 2013, Page No: 247-256.

[3] Mohamed Shehab a, Anna Squicciarini b, Gail-Joon Ahn c, Irini Kokkinou "Access control for online social networks third party applications" Elsevier- Computers & Security 31 (2012) 897 911.

[4] *George Lawton*, Technology News, Published by the IEEE computer society, October 2007, Page No: 13-16.

[5] Joe, M. Milton, B. Ramakrishnan, "Enhancing Security Module to Prevent Data Hacking in Online Social Networks", Journal of Emerging Technologies in Web Intelligence, Vol. 6, No. 2, May 2014, Page No: 184-191.

[6] Joe, M. Milton, B. Ramakrishnan, "A Survey of Various Security Issues in Online Social Networks", International Journal of Computer Networks and Applications, Volume 1, Issue 1, November – December (2014), Page No: 11 – 14.

[7] YIN Hao, FU Qiang, LIN Chuang, LIN Chuang, DING Rong, LIN Yishu, LI Yanxi, FAN Yanfei, "Mobile Police Information System Based on Web Services " TSINGHUA SCIENCE AND TECHNOLOGY - Volume 11, Number 1, February 2006, ISSN 1007-0214 01/21 pp1-7.

[8] Chen, Wen-Shiung, Lili Hsieh, and Ying-Neng Hsieh. "Design and Implementation for SIP-based Push-to-Talk Services over 802.11 Networks." International Journal of Computer Networks and Applications (IJCNA), 2.6 (2015),PP: 261-266.

[9] Altunbey, Feyza, and Bilal Alatas. "Overlapping community detection in social networks using parliamentary optimization algorithm." *International Journal of Computer Networks and Applications* 2.1 (2015): 12-19.

[10] Watkins Demien. Mobile web services technical roadmap. http://www.microsoft.com/serviceprov iders/mobilewebser vices/mws_tech_roadmap.asp. 2003, 11.

[11] Jonathan R. Mayer and John C. Mitchell, "Third-Party Web Tracking: Policy and Technology" 2012 IEEE Symposium on Security and Privacy, DOI 10.1109/SP.2012.47, Page No: 413- 427.

[12] World Wide Web Consortium. Content Security Policy. [Online]. Available: http://w3.org/TR/CSP/

[13] J. Grossman, R. Hansen, P. D. Petkov, A. Rager, and S. Fogie, XSS Attacks: Cross-Site Scripting Exploits and Defense. Burlington, MA: Syngress, 2007.

[14] Chen, P., Desmet, L., Huygens, C., & Joosen, W. (2016, April). Longitudinal Study of the Use of Client-side Security Mechanisms on the European Web. In *Proceedings of the 25th International Conference Companion on World Wide Web* (pp. 457-462).

[15] A. Barth, C. Jackson, and J. C. Mitchell, "Robust defenses for cross-site request forgery," in Proceedings of the 2008 ACM Conference on Computer and Communications Security, October 2008.

[16] W. Zeller and E. W. Felten, "Cross-site request forgeries: Exploitation and prevention," Princeton University, Tech. Rep., September 2008.

[17] P. Venkateswari, T. Purusothaman "A Secure Simple Authenticated Key Exchange Algorithm based Authentication for Social Network" Journal of Computer Science 7 (8): 1152-1156, 2011.

[18] Joe, M. Milton, and B. Ramakrishnan. "Novel authentication procedures for preventing unauthorized access in social networks." Peer-to-Peer Networking and Applications (2016): 1-11

[19] M. Joe, and B. Ramakrishnan, "Review of Vehicular Ad hoc Network Communication Models including WVANET (Web VANET) Model and WVANET Future Research Directions," Wireless Networks, Springer, vol. 22, no. 7, pp. 2369-2386, 2015.

[20] Hsiang, H.C. and W.K. Shih, 2009a. Efficient remote mutual authentication and key agreement with perfect forward secrecy. Inform. Technol. J., 8:366-371. DOI: 10.3923/ITJ.2009.366.371.

[21] Aboud, S.J., 2010. Efficient password-typed key agreement scheme. Int. J. Comput. Sci., 7: 26-31. http://www.doaj.org/doaj?func=abstract&id=495633

[22] Berkeley, A.R.U.C., 2008. Personal knowledge questions for fallback authentication: security questions in the era of Face book. Proceedings of the 4th Symposium on Usable Privacy and Security, (SOUPS'08), ACM, New York, pp: 13-23. DOI: 10.1145/1408664.1408667.

[23] https://hakin9.org/defeating-on-screen-or-virtual-keyboard-protection/

[24] Meier J D, Mackman A, Dunner M, Vasireddy S. Web services security S. http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnnetsec/html/SecNetch10.asp.2002.

[25] Powell Matt. Web services security (WS-security). http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnglobspec/html/ws-security.asp. 2002.

[26] Nishanth, R. Bhagavath, B. Ramakrishnan, and M. Selvi. "Improved signcryption algorithm for information security in networks." *International Journal of Computer Networks and Applications (IJCNA)* 2.3 (2015): 151-157.

[27] Wang, Y., Wen, S., Xiang, Y., & Zhou, W. (2014). Modeling the propagation of worms in networks: A survey. *IEEE Communications Surveys & Tutorials*, *16*(2), 942-960.

[28] S. Staniford, V. Paxson, N. Weaver *et al.*, "How to own the internet in your spare time." in *USENIX Security Symp.*, 2002, pp. 149–167.