IN A CLOUD COMPUTING ENVIRONMENT, A SECURE MULTI-TIER AUTHENTICATION SYSTEM

Abstract

Owing to its safe service, remote storage solutions are becoming more and more popular. Yet, it's likely that eventually, information regarding the security measures put in place on the server side will become public. New security dangers and assaults are consequently occurring in the cloud. Thus, a modernised and secure authentication technique is needed. Any cloud service that transfers sensitive or private information cannot be secured with single-tier authentication. When compared to single-tier authentication, multi-tier authentication is substantially more secure. Many multi-tier authentication methods are employed in computer settings, but none of them provide security against virtualization or insider attacks. In a cloud context, the server handles all aspects of authentication control. With a cloud system, it is therefore challenging to believe in a third party server. This study recommends a more advanced and secure multi-tier authentication method for getting access to cloud services.

Keyword: ***Cloud Computing, Insider Attacks, Authentication Scheme, Multi-tier Authentication***

Authors

Shiwani tiwari
Shri Ram Group of Institution Jabalpur.

I. INTRODUCTION

As it has become a widely accepted paradigm in a variety of disciplines, including business and academics, cloud computing [1] technology has been embraced by several organisations and individuals in recent years. These systems provide virtual on-demand services using a substantial amount of shared resources, such as network, server, storage, etc. They have excellent scalability. The field of computers has been developing fast every day. The system has to be updated in order to obtain a large virtual capacity and high computational power as a result of developments in computing technology [1]. Internet technology has advanced quickly in recent years. People use the internet to communicate, do business, play games, obtain information, and more. For all of this online activity, some kind of identification is required to verify that the individual is who they say they are. Financial transactions require additional security data, such as personal information and other account-related data.

Username-password authentication, biometric face recognition, Kerberos authentication, Public Key Infrastructure authentication [3] [4], Hybrid Text Image Based authentication, and Symmetric Key Based authentication are some of the several processes used by the authentication approach [2]. A mechanism for ensuring the veracity of the participants in communication is an authentication scheme.

The most common type of authentication is username-password. In the case that the server is compromised, this technique poses a variety of security issues related to the exposing of sensitive user data. Users never have complete faith in the third-party server as a result. Password-based authentication is also vulnerable to dictionary attacks, Man-in-the-Middle attacks, and other security weaknesses [5, 6].

In a cloud computing environment, a third party is in charge of providing storage space, processing power, etc. [7]. The process of authenticating becomes quite difficult as a result. The cloud [8], which is owned and maintained by a cloud storage provider, is where users store their data. So, many are reluctant to save their data in this form of cloud database. Each user wishing to access cloud resources must first provide identification documentation that certifies their identity and grants them access.

This article recommends a multi-tier authentication method to gain access to the services since single-tier authentication is insufficient. The authentication process involves two phases (two-level). On the initial stage, the user enters a simple username and password. At the second step, the user follows a specified series or sequence. The advantage of this two-tier authentication strategy is that no new hardware or software is needed.

This work develops and implements a secure multi-tier authentication approach for cloud computing. The next section discusses literature reviews. Section III discusses the limitations of various authentication systems. The suggested authentication technique is described in Section IV. It concludes the piece by discussing the work's prospects for the future. It also summarises the results of the suggested system.

II. A REVIEW OF THE LITERATURE

The different online apps that employ authentication are covered in this section. Also, it presents the benefits and drawbacks of each authentication method. The majority of applications employ username-password authentication techniques [9]. Online, different password-cracking tools are accessible for free. As a result, the password cracking process takes a little while [10]. NIST and the Federal Financial Institutions Examination are responsible for maintaining security against these types of threats.

The Council (FFIEC) issues directives for conducting financial transactions [11].

Some of the research on two-tier authentication or two-factor authentication systems may be found in papers [11] [12] [13]. According to the study [12] [13], apps employ many forms of authentication and that a single-tier login password is insufficient for accessing services. The login password comes first, followed by the secret code that's transmitted to their cell phone. The user always needs a mobile phone and SIM card to access the services, which is a drawback of this arrangement.

There are some general rules that concentrate on managing security risks and assaults that may be used to access various Internet-based financial services [11] [12]. Since 2001, the rules have given more importance to safeguarding consumer data, minimising fraud, and maintaining identity safety. Also, these suggestions offer a few actions that may be taken to advance authentication technology.

Financial institutions have a variety of methods they can use to authenticate customers when they access financial services, including customer passwords, Personal Identification Numbers (PINs), Digital Certificates, Public Key Infrastructure (PKI) [14], Smartcards, One-Time Passwords (OTP), some security tokens, Profile Scripts, and biometric identification.

There are several different multi-factor authentication systems listed in [11], including:

1. **Shared Secrets:** The information element (Key) that is shared by the client and the reliable third party is known as a shared secret [15].
2. **Tokens:** A token is a piece of digital information that is distributed around the internet, created using XML, and contained one or more claims [16, 17].
3. **Biometrics:** Based on a person's bodily characteristics, biometric technology verifies a person's authenticity [18] [19].
   Non-Hardware-Based One-Time Password: This method involves giving the consumer a scratch card. A certain number that appears in a specific location on the scratch card serves as the one-time password.
4. **Hardware-Based One-Time Password:** This approach requires two-tier authentication; after entering a username and password, the user must input a secret password that has been received to their cell phone.
5. **Internet Protocol Address (IPA) Location and Geo-Location:** Using this method, the service provider verifies the user's physical condition by determining his location. For instance, if a consumer lives in India and completed his initial transaction there, his subsequent transactions would only be carried out in India.

The paper [20] offers a single sign-on mechanism (SSO). A security broker can use the single sign-on approach to help authenticate a cloud service user across numerous cloud service environments. This authentication will last as the user uses additional cloud services. The benefit of SSO is that as a consequence, customers of cloud services do not need to re-authenticate themselves with each new request. When a consumer of cloud services wants to access cloud services hosted on separate clouds, the security broker of the SSO method is most helpful.

The drawback of SSO is that if the SSO server is compromised, the entire cloud environment is compromised as well.

A mechanism for authentication based on many layers is provided in the paper [21]. It produces a password at level one and concatenates the resulting password at further levels. The user enters the password to access the cloud services at each level. The benefit of this method is that the user inputs a different password again at each level. Hence, it is challenging to circumvent layered security. This method's drawback is that new passwords are used at each level, making them difficult to remember.

There is a concern when a user stores their data on a third party cloud. The issue is how the customer will trust on the third party provider's statements [22] about its security capabilities. The user must maintain a private cloud that is secure and has the necessary security features. The security capabilities of this private cloud will be evaluated using third-party security capabilities [23]. Access will be provided if the data complies with the private cloud policy; otherwise, it won't. The benefit of this method is that users may apply their own security policies to private clouds. This method's drawbacks include additional overhead, maintenance difficulties, and higher initial private cloud costs.

The methodology discussed in the papers [24, 25] focuses on the establishment of a session key for identity management and mutual authentication between a user and a cloud server. This system uses two-tier verification, which is based on OTP, smartcards, or other techniques, to confirm user validity. This scheme's benefit is that it gives the client side additional security and control to fend off assaults. This solution has the drawback of requiring additional gear and software from the user in order to access the services.

The papers [18] [19] [26] provide other methods of authentication that make use of the user's physical attributes or a biometric. This technique has the advantage of using multi-tier authentication, however the user needs some additional hardware for authentication.

III. LIMITATIONS OF CURRENT TECHNIQUES

We explore numerous authentication methods in the literature review section, along with their benefits and drawbacks. These approaches' drawbacks may be broken down into four categories: protection from insider attacks, presence of client- or server-side authentication controls, additional software and hardware requirements, and number of security levels needed. Based on the aforementioned factors, Table 1 compares several methodologies [27].

This is a succinct overview of the comparison using these criteria:

1. **Security against insider attacks:** This criterion is predicated on the idea that it is simple for an insider to have access to first-tier login and password. This cannot be adopted. Multi-tier authentication is therefore necessary.
2. **The existence of authentication controls directed at the client or server:** In a cloud computing environment, users save their data on remote cloud storage (third party storage), which is managed and regulated by third party services.
3. Provider. Thus, there should be a method for client-side user authentication.
4. **More hardware and software required:** Certain authentication methods call for additional TABLES from the user I: Evaluation of Several Authentication Methods

Scheme/ Parameter	Security from insider attacks	Presence of authentication control towards	Additional hardware and software needed	No. of security tiers
Authentication using Single Sign-on	No	Server	No	1
Multi-level Authentication technique	Yes	Server	No	More than one
Two-tier	No	Client/Server	No	2
Strong user authentication	Yes	Server	No	2
PAS	No	Server	Yes	2
AQP	No	Server	Yes	2
Architecture based on proactive model	No	Client/Server	Yes	1
SOA	No	Server	No	2
Multi-tier authentication	Yes	Client/Server	No	2

Both software and hardware. The system's performance suffers as a result, and overhead goes up. The system as a whole fails if extra hardware malfunctions.

5. Necessary Security Tier Numbers One-tier authentication is substantially less secure than multi-tier authentication. Single-tier cannot protest against insider assaults, according to papers [12, 13]. Multi-tier authentication is therefore necessary.

IV. PROPOSED SCHEME: TWO-TIER AUTHENTICATION

Plan and their restrictions. We suggested a more sophisticated and effective multi-tier authentication method in this section. By employing this method, we may lessen the drawbacks of the current strategy that were covered in the preceding part. The benefit of the suggested plan is that it does not call for additional hardware or software.

Second-tier authentication is carried out on the client side, making it more secure.

Multi-tier authentication is the foundation of the suggested authentication strategy. There are two levels to this plan. The user inputs their login and password in the first layer, which uses a straightforward authentication process. The second tier

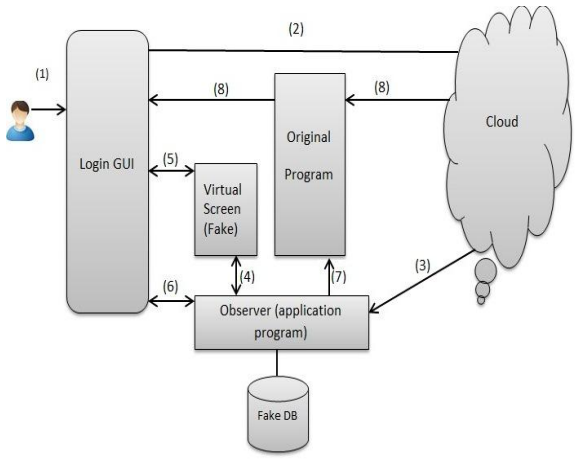


Figure 1: shows the suggested authentication scheme's architecture.

On a series of specified activities on the virtual screen, authentication is predicated. This sequence needs to match what the user did while registering. This virtual display has been loaded by the viewer (This is application programme run at client side).

The stages that followed detailed the general procedure seen in fig. 1:

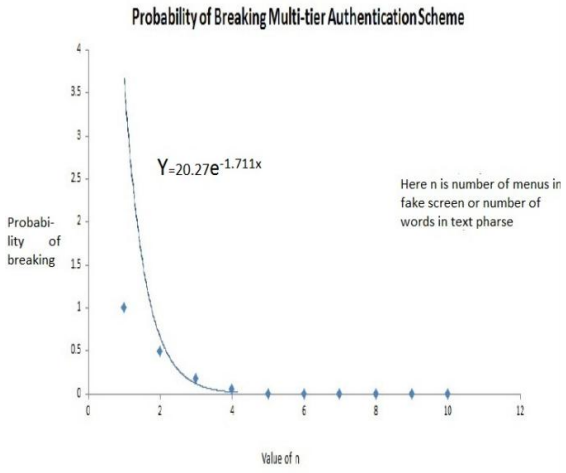
Step 1: The user types in the browser the URL of the cloud application site. The browser has loaded a login GUI screen.
Step 2: User provides username and password (primary credentials) in step two. As seen in fig.1, these credentials are sent to the cloud server for verification.
Step 3: The cloud server verifies the login information. If both credentials are accurate, the cloud server sends a validation request.
Step 4: The observer retrieves the information from a fictitious database and starts the code to load a fictitious screen after getting the cloud server's validation reply.

Step 5: The false screen is loaded client-side in the browser when the data fetching procedure is complete.

Second-tier authentication begins at step six. User enters specified sequence after registering. The activities on the false screen are continually monitored by the observer.

Step 7: The original screen is loaded in the browser if the user's behaviour is proper and they follow the registered sequence.

Step 8: When the preceding step has been successfully completed, it creates a direct line of communication between the cloud server and the client.



The following three second-tier authentication procedures might be used in the method described above:

1. **Menu Activity:** Following the successful completion of first-tier registration, the user logs a series of menu item clicks. This sequence is kept in a fictitious database. The user follows the same click order while performing second-tier authentication. The original screen loads if the click sequence is right; otherwise, it fails and is not displayed on the original screen.
2. **Text Field Activity:** The cloud server will ask the user a series of security questions when they register on the platform. The user enters the responses to these queries, which are then saved in the fictitious database. If the user responds correctly to this question upon login, the original screen loads. is loaded in the browser otherwise user is not valid and original screen is not loaded.
3. **Mouse Activity:** A mouse event occurs when a user clicks their mouse at a certain location while registering. Also, it involves moving the mouse from one coordinate to another. The observer records in the fictitious database the mouse event activity and the quantity of clicks at certain positions. If the mouse event during login is accurate, the second-tier authenticated is verified.

V. EVALUATION OF PERFORMANCE

We covered the proposed effective multi-tier authentication technique in the preceding section. Google App Engine [33] and Eclipse Integrated Development Environment were used to develop this strategy (IDE). The document [29] [30] provides a variety of essential information for utilising Google App Engine to construct a cloud application. forming a cloud

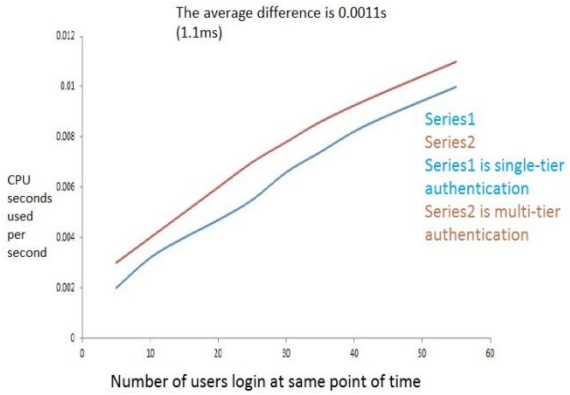


Figure 3: Performance of the system [27]

Eclipse IDE has been used to create an application for the cloud server by installing the Google App Engine plug-in package [33] [34]. The contact between a user and a cloud server is established via Remote Procedure Call (RPC) [31, 32]. We examine numerous results factors on the Google Dashboard [28] screen.

While evaluating the system's performance, the following variables are regarded as crucial:

System Security: The fundamental flaw with the prior authentication method is that it is vulnerable to insider assaults. So, we are concerned about this form of assault in the proposed method. The cloud storage server stores data in the form of hashed data. Hence, there is extremely little chance of insider assaults. There are two processes for authentication in the suggested approach. Let's assume that the two results of the authentication are S and F, respectively, for success and failure. The results of the first two levels are SS, SF, FF, and FS. There were four events overall, or n(S). If P represents the success at each level, then P(E)=P2 represents the likelihood of multilayer authentication being broken.A failed authentication attempt at any level denoted by $1-P(E)=1-P^2$. Let we assume that probability of If each level is successful, there is a P2=0.001 chance that multilevel authentication will be compromised. If the key (password) is 128 bits long, there are 2128 distinct key combinations in the first layer. As a result, key security also improves.

We primarily deal with second-tier authentication strength as follows in the suggested scheme:

o Menu Activity: If there are n menu items on the menu, n!different menu combinations are possible. If a user only provides one chance for authentication at a time, the likelihood of success is 1/n!

o Text Field Activity: The quantity of characters in the response field is taken into consideration. The likelihood of success is 1/k! if the response has k characters since there are k! different possible combinations of replies.

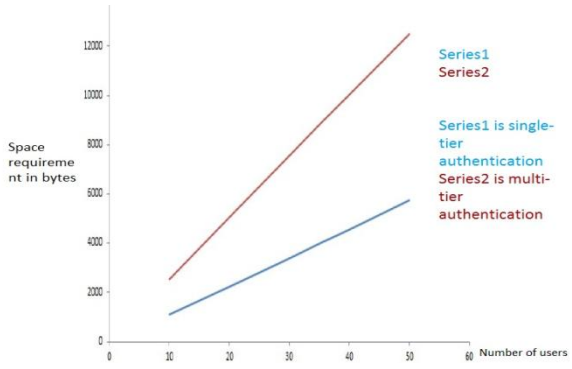
o Mouse Event: With this technique, the user selects the sequence and locations for any number of clicks. Users only get one opportunity to authenticate. So, it is quite difficult for hackers to figure out that click order. Also, it offers a very high degree of security.

The second factor security's success probability is 1/(n! or k!). The likelihood of success will fall in proportionally if the value of the variable (n or k) is increased by a tiny amount. Consider the case where n=5 and n!=120. The current success probability (breaking second factor) is 1/n!=1/120=0.0083. Assuming that n is now 6 and that n!=720, the chance of the second component breaking is now 1/720, or 0.0019. In the case of n=10, the value is 1/3628800=0.00000028. With the help of the easily understood graph in Figure 2, the aforesaid conclusion and likelihood of success are demonstrated.

System Performance: The system's efficiency is measured in terms of CPU processing time. The CPU processing time is a crucial factor in determining how well the suggested multi-tier authentication strategy performs. We require both a single-tier and multi-tier authentication solution in order to assess performance. Both methods need an equal number of successful login attempts throughout the testing period. The graph in Figure 3 shows the experimental values of login attempts against CPU seconds consumed per second.

The graph in figure 3 above clearly demonstrates that multi-tier only requires 0.0011 seconds longer than single-tier. The entire plan was created in a cloud computing environment. We are aware that the cloud enables us to scale our resources indefinitely. Thus, this difference is insignificant. So, based on the findings, it can be concluded that the suggested multi-tier authentication technique requires a little longer processing time than the single-tier scheme. Yet, the additional processing time has no impact on the system's performance.

Need space: This parameter is crucial for determining how efficient this suggested multi-tier authentication solution is in terms of additional storage. first calculating additional storage space



More room for a multi-tier authentication technique is shown in Figure 4.

Storage requirements for single-tier authentication must be determined. The findings for both approaches were combined, and the graph that resulted clearly demonstrates that both are linear functions of data input, i.e., that as the number of data entries (users) increases in a multi-tier authentication strategy, so does the space need. One user's credentials need to be stored in 114 bytes for a single-tier authentication system and 253 bytes for a multi-tier

authentication strategy. The system would require 1000000*(253- 114)=139000000 (about 139 MB) more storage space if ever 1000000 people enrolled. More security is not a major concern in a cloud computing setting. Figure 4 depicts the topic mentioned earlier.

VI. CONCLUSIONS

It is abundantly clear that the effectiveness of any authentication method depends on whether it can be successfully circumvented. The suggested plan improves security by using two-level authentication. The benefits of two-level authentication are enjoyed by the proposed multi-tier authentication in cloud computing, which also protects user privacy. One benefit of multi-tier setup is that second level authentication is solely handled at the client side with no additional hardware or software needs. The suggested plan offers a compromise between security and effectiveness. Also, our approach is built on elements of cloud computing, where jobs are spread effectively to avoid the flaws of earlier systems.

VII. PROJECTED IMPROVEMENTS

The suggested plan does have certain benefits in terms of extra hardware and software needs, but it also has some disadvantages. On both levels, it is not feasible to change the login or password. This is a serious issue that has to be addressed in the future. Future advancements may include more methods of retrieving passwords in multi-tier environments.

REFERENCES

[1] Peter Mell and Tim Grace,*"The NIST definition ofcloud computing"*, NIST Special Publication 800-145 (SP800-145), National Institute of Standards and Technology, Gaithersburg, January 2011, pp. 1-52.

[2] White paper for authentication and authorization, "http://www.cryptocard.com/images/stories/pdfs/Authentication WP.PDF".

[3] B. Goswani, and D. S. Singh, *"Enhancing Security in Cloud computing using Public Key Cryptography with Matrices"*, International Journal of Engineering Research and Applications, vol. 2, no. 4, pp. 339-344, 2012.

[4] M. K. Boyarsky,*"Public-key Cryptography and Password Protocols: The Multi-User Case"*, Proc. of the 6th ACM Conference Computer and Communication Security (CCS99), Singapore, Nov. 1999, pp.63-72.

[5] U. Oktay and O.K. Sahingoz, *"Attack Types and Intrusion Detection Systems in Cloud Computing"*, 6th International information Security & Cryptology Conference, pp. 71-76, 20-21September 2013.

[6] Yashpalsinh Jadeja, Kirit Modi, *"Cloud Computing- Concepts, Architecture and Challenges"*, 2012 International Conference on Computing, Electronics and Electrical Technologies [ICCEET].

[7] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, *"A view of cloud computing"*, Communication of the ACM, vol.7, No.4, Apr. 2010, pp. 50-58.

[8] Gurudatt Kulkarni, Rani Waghmare, Rajnikant Palwe, Vidya Waykule, Hemant Bankar, and Kundlik Koli *"Cloud Storage Architecture"*, 2012 7th International Conference on Telecommunication Systems, Services, And Applications(TSSA).

[9] S. Bellovin and M. Merritt,*"Encrypted Key Exchange: Password- Based Protocols Secure against Dictionary Attacks"*, Proc. of the IEEE Sym- posium on Research in Security and Privacy (SRSP92)), May 1992, Oakland, CA, USA, pp.72-84.

[10] Chun-I Fan, Pei-HsiuHo, and Ruei-Hau Hsu, *"Provably Secure Nested One-Time Secret Mechanisms for Fast Mutual Authentication and Key Exchange in Mobile Communications"*, IEEE/ACM Transactions on Networking, Vol. 18, No. 3, JUNE 2010.
[11] "Authentication in an Internet Banking Environment", Federal Financial Institutions Examination Council, Government of USA, 2005.
[12] David Chou,"Strong User Authentication on theWeb", Microsoft Corporation, August 2008 Available at: http://msdn.microsoft.com/en- us/library/cc838351.aspx
[13] Prof. More V.N, *"Authentication and Authorization Models"*, International Journal of Computer Science and Security (IJCSS), Volume 5, Issue 1, 2011.
[14] Y. Zheng, *"Public Key Cryptography for Mobile Cloud"*, Information Security and Privacy, Lecture Notes in Computer Science C. Boyd and L. Simpson, eds., pp. 435- 435: Springer Berlin Heidelberg, 2013.
[15] Prashant et al.,*"An Architecture Based on Proactive model for Security in Cloud"*, International Conference on Recent Trends in IT, IEEE, 3-5 June 2011, pp. 661-666.
[16] Claimstypes:http://blogs.msdn.com/vbertocci/archive/2008/05/05/claimtypes-a-coarsetaxonomy.aspx
[17] Security Assertion Markup Language, A Brief Introduction toSAML, Tom Scavo, NCSA.
[18] S. Jeon , H. S. Kim, and M. S. Kim,*"Enhanced Biometrics-based Remote User Authentication Scheme Using Smart Cards"*, J. of Security Engineering, Vol.8, No.2, Apr. 2011, pp.237-254.
[19] J. Wayman, A. Jain, D. Maltoni, and D. Maio,*"An Introduction to Bio-metric Authentication Systems"*, Biometric systems Technology, Design and Performance Evaluation, Springer, London, 2005, pp.1-20.
[20] Pashalidis, A., Mitchell, C., *"A taxonomy of single sign-on systems"*, Proceedings, volume 2727 of Lecture Notes in Computer Science, Springer-Verlag, Berlin, July 2003, pp.249-264.
[21]]Dinesha et al.,*"Multi-level Authentication Technique for Accessing Cloud Services"*, International Conference on Computing, Communication and Applications (ICCCA), IEEE, 22-24 February 2012, pp. 1-4.
[22] Dominick Baier, Vittorio Bertocci, Keith Brown, Scott Densmore, Eugenio Pace, Matias Woloski, "A GUIDE TO CLAIMS-BASED IDENTITY AND ACCESS CONTROL", Authentication and Authorization for Services and the Web, 2nd ed. http://msdn.microsoft.com/enin/ library/ff423674.aspx.

