Strengthening Authentication System Using Mindmetrics

[1] Abhishek Karkamkar [2] Pawnesh Kumar [3] Rohit Sharma [4] Saurav Modi
[1][2][3][4] Department of Computer Engineering, Rajarshi Shahu College of Engineering
[1]abhishek.karkamkar@gmail.com, [2] pawnesh.kevin@gmail.com, [3]mannurohit07@gmail.com,
[4] sauravmodi03@gmail.com

Abstract: **Authentication is a two step process, identification and verification. Earlier system didn't give any stress on identification process; our system focuses on whether the user is legitimate user or not by using a personal secret data of the user instead of using his login id for the identification process. This is achieved with the help of Mind metric token. Our system will ask the user to enter the Mindmetric token at the time of registration. User has to use his unique Mind metric token along with the password at the time of login. This helps in adding extra security for identification step. Mind metrics is what resides in user's brain. User uses this token to pass first identification process step, after this a set of login IDs are displayed to user in partially obscured form. A legitimate user can easily select his ID and can successfully complete the identification process. To begin verification process user has to pass OTP process and a picture based question process. The password of user is divided into two halves and encrypted and then stored in two servers. This adds more security to verification process. At the time of verification of password the encrypted password which is stored on the two servers are decrypted and then merged. After decryption the password is matched with the user password which he has entered at the time of login along with the Mind metric token. If both the password matches with each other, then only the user is a fully legitimate user and has a full access to his account. Thus system not only enhances identification process but also enhances verification process. Thus in turn authentication is strengthened. This scheme is used where biometrics scheme cannot be used cost-effectively.**

Keywords**: Authentication, Verification, Mind metric, Identification, OTP**

I. INTRODUCTION

Computer systems use an authentication mechanism to allow access only to legitimate users. The authentication process is consisting of two parts, identification and verification. Identification process is used to verify "who the user is?" and verification process is used to verify if the user is legitimate or not. Traditionally the identification was performed by a "username" or login ID and the "password" for verification. In a password based system, the plaintext passwords are transferred into hash values is generated from the newly entered password, and compared with the stored hash values in the password hash file. If the hash value matches, access is granted. This password verification process is the heart of the most authentication systems.

There are number of ways to stole the user's password for illegal access. Plaintext passwords can be hacked from the network, by malware or by key logging software. When the plaintext password is not available, the attackers can try password-guessing attack where they try possible values for the victim user. In the password cracking attack, the attackers obtain a password hash file, and tries different inputs to find an input that produces the same hash values as the victim user's hash value.

While passwords are supposed to be a random characters, login IDs are not random. They are used for communication or accounting purpose, and must carry a meaningful pattern. It may be part of user's first and/or last names, part of social security number, combination of names and numbers, account number or email addresses. Thus login ID's are publicly known or can be guessed easily. In other words, obtaining the login ID is generally not a barrier for the attackers, and the success of an attack depends on the difficulty of the password.

The term "Mind metrics" is coined with the concept of Biometrics as it is similar to biometrics.
Biometrics is a field of study which aims to identify or recognize people based on traits they have. Given these traits, a system can be trained to recognize certain people, with a certain probability. Biometrics refers to metrics related to human characteristics. Biometrics authentication is used in computer science as a form of identification and access control.

Mind metrics uses some secret data instead of human characteristics as a token to identify the user. It utilizes personal secret data instead of a login ID to identify a user uniquely, hence mind metrics.

The concept of biometrics or mind metrics is used in authentication schemes to identify a user with legitimate ID holder.

Comparison between biometrics and mind metrics:

1] Some special hardware device (e.g.: thumb scanner) is required in biometrics. On the other hand no specialized hardware is required in mind metrics and can be easily implemented. So mind metrics can be used for accessing local or remote computing systems from any conventional private or public computers.

2] Biometrics is costly and cannot be easily implemented on public e-commerce web sites. Mind metrics is cost effective and can used for public e-commerce web sites.

3] Mind metrics is a deterministic process, and thus there is no uncertainty. Thus mind metrics is more practical, cheaper, and can be used by any public web sites such as e-commerce web sites.

II. RELATED WORKS

In "Mind metrics: Identifying User's Without their Login ID's" [6] the main focus was only on Identification process. The Verification process was same as old one. Say, the password was stored in a hash table using a cryptographic hash value of the password over a public channel which makes hash value accessible to an attacker. this was a drawback of this system. Because the password was stored in a single server in hash table, it is not very difficult for a attacker to get the password from a hash value.

III. PROPOSED WORK

Passwords are commonly used by people during a login process that controls access to protected computer operating systems, mobile phones etc. A computer user may require password for many purposes: retrieving e-mails from web servers, logging into computers, databases, accessing programs, networks, websites and social networking websites.

In our Mind metrics system, we are using "Efficient Two-Server Password-Only Authentication key Exchange Protocol" [2] for storing password. The password will be divided into two parts and stored in two different servers. This adds extra security to our Mind metrics system. Consider a scenario where a password is stored in two servers, and if one server is compromised, the attacker still cannot get the access because of partial information.

Also we are using "Time and Location Based One-Time Password Authentication Scheme" [1] in our system at the time of login. This adds extra security to our system. This OTP scheme is used with "SMS-Based Authentication Through Usability" [4] to send the OTP Password as a text SMS on user's Mobile Device.

1. Algorithms

1.1 Diffie-Hellman key Exchange Protocol

The Diffie-Hellman key exchange protocol [5] was invented by Diffie and Hellman in 1976. It was the first practical method for two users to establish a shared secret key over an unprotected communication channel.

Consider two users Alice and Bob, who are totally unaware of each other, but want to establish a secure communications between them. Diffie-Hellman key exchange protocol can be used as follows:

1. Alice and Bob agree on a cyclic group G of large prime order q with a generator g.
2. Alice randomly chooses an integer a from Z^*_q and computes $X=g^{a,}$ while Bob randomly chooses an integer b from Z^*_q and computes $Y=g^b$. Then Alice and Bob exchange X and Y.
3. Alice computes the secret key $k_1 = Y^a = g^{ba}$, while Bob computes the secret key $k_2=X^b=g^{ab.}$

It is obvious that $k_1=k_2$ and thus Alice and Bob have agreed on the same secret key, by which the subsequent communication between them can be protected.

Diffie-Hellman key exchange protocol is secure against any passive adversary, who cannot interact with Alice and Bob, attempting to determine the secret key solely based upon observed data.

1.2 ElGamal Encryption Scheme

ElGamal encryption scheme was invented by ElGamal in 1985 [3] on basis of Diffie-Hellman key exchange protocol. It consist of key generation, encryption, and decryption algorithm.

ElGamal encryption consists of three components: the key generator, the encryption algorithm, and the decryption algorithm.

1.2.1 Key Generation

The key generator works as follows:

- ❖ Alice generates an efficient description of a cyclic group **G** of order **q** with generator **g**. See below for a discussion on the required properties of this group.
- ❖ Alice chooses an x randomly from . **{1,….,q-1}**

- Alice computes .**h : =** $\mathbf{g^x}$
- Alice publishes **h**, along with the description of **G,q,g**, as her **public key**. Alice retains x as her **private key**, which must be kept secret.

1.2.2 Encryption

The encryption algorithm works as follows: to encrypt a message **m** to Alice under her public key (**G,q,g,h**).

- Bob chooses a random **y** from **{1,…..,q-1}**, then calculates **c1 :=** $\mathbf{g^y}$.
- Bob calculates the shared secret **s :=** $\mathbf{h^y}$.
- Bob maps his secret message **m** onto an element **m'** of **G**.
- Bob calculates $\mathbf{c_2 := m' . s}$.
- Bob sends the ciphertext $\mathbf{(c_1,c_2)=(g^y,m'.h^y)=(g^y,m'.(g^x)^y)}$ to Alice.

Note that one can easily find $\mathbf{h^y}$ if one knows **m'**. Therefore, a new **y** is generated for every message to improve security. For this reason, **y** is also called an ephemeral key.

1.2.3 Decryption

The decryption algorithm works as follows: to decrypt a ciphertext $\mathbf{(c_1,c_2)}$ with her private key **x**,

- Alice calculates the shared secret $\mathbf{S := c_1{}^x}$ and then computes $\mathbf{m' := c_2.s^{-1}}$ which she then converts back into the plaintext message **m**, where $\mathbf{S^{-1}}$ is the inverse of S in the group **G**. (E.g. modular multiplicative inverse if **G** is a subgroup of a multiplicative group of integers modulo *n*).

The decryption algorithm produces the intended message, since

$$\mathbf{C_2 . s^{-1} = m' . h^y . (g^{xy})^{-1} = m' . g^{xy} . g^{-xy} = m'.}$$

2 *Detailed Description Of Mindmerics System*

Step 1: Mind metrics token registration

In this step, the user will submit his Mind metrics token in the system. For ex:- "This is my secret token #4", and specifies a desired login ID and password . And other user credentials like email id and mobile number.

A generalized image is provided to the user with a question, for ex:- "what do you see in the given pic?". After the user has entered them, the user selects "create account". Then the authentication server validates the entered information, for example, to ensure that the login ID is unique among all login IDs and that the password satisfies the password requirements (e.g. length of the password). If there is an error, it may prompt the user to enter another login ID or password.

Step 2: Multiple login IDs

A number of fake IDs are created together and will be displayed during normal login process. IDs shown to user will be in obscured from, for ex:- login ID "fake1234" will be shown as "fa****34".This is not a problem for a legitimate user, but it makes difficult for the attackers to recover the full login ID. The authentication server identifies the associated fake login IDs from the index table.

Step 3: OTP verification

After the user will pass the identification stage. He will be directed for verification stage. In this step, he will be provided by a randomly generated OTP code through SMS on his/her mobile. He needs to enter the OTP code in provided OTP field for verification.

Step 4: Preventing inference attack with fixed choices

The displayed list of partially-obscured login IDs is the same and appears in the same order every time. An attacker cannot simply supply the same token with a different computing system in order to identify which partially-obscured login ID does not change.

Step 5: Handling tokens which not exist

If the token does not match any token in the hash file, the identification system creates random login IDs and show it to the user. This prevents an attacker from knowing whether the entered token matches any user accounts. To prevent an inference attack, the choice of displayed login Ids do not change when the attacker types the non-existing token again.

Step 6: Image question verification

After getting the authentication from the OTP stage the user will be provided by a Image with one question. The answer given by the user is compared form the answer stored in sever at the time of registration

Step 7: Password verification

After the Image question verification the entered password will be send to verification server. The verification server retrieve the partially strored password from the two servers, decrypt it and merge it for password verification. It they matches the credentials assigned to the user account, the verification server grants an access.

If the user have entered the credentials that did not match all the information assigned to the user account, the serverd indicates that the user had entered wrong credentials and try it again from the begning.

2.1 SYSTEM ARCHITECTURE

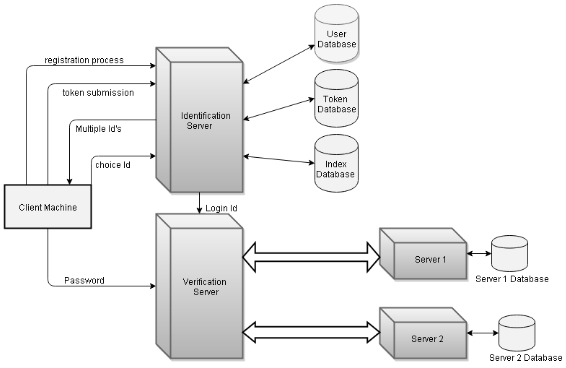

Fig 1: System Architecture

IV. MATHEMATICAL MODEL

Let S be the system describing authentication process
S= {Input, Output, Function, success, failure}

•For Register operation
Input = {User Details}
User Details D = {D1, D2… ,Dn}
D = {Username, Firstname, Lastname, Password, Email Id, Contact, Token}

•For Login operation
Input = {Login Credentials}
Login Credentials C = {C1, C2…, Cn}
C = {Username, Token, Password}
Output:

•For Register operation:-
Output={Account Creation Summary}
Account Creation Summary = {Username, Password, Token, Correct User Id, Fake User Id1, Fake User Id2, Fake User Id3}

•For Login operation
Output ={Authentication Notification}
Constraint:
Constraint C = {C1, C2, C3}

Where,
C1 = "All Servers and Client Machines should be connected in one network"
C2 = "User should enter the token that fulfills all the validation norms during registration"
C3 = "Both the servers connected to the verification server should be in active mode while authenticating the user"
Identify data structures, classes, divide and conquer strategies to exploit distributed /parallel/concurrent processing, constraints.

Functions: Identify Objects, Morphisms, Overloading in functions, Functional relations

Success Conditions: User Mindmetric token, OTP, Password match successfully.

Failure Conditions: Even after entering the correct credentials of the user system is not providing authentication to the user.

V. CONCLUSIONS

User authentication is done in two steps, identification and verification. The traditional password-base verification system are been challenged by sophisticated attacks, but the new schemes are being made to cover the weakness of the password-based systems. The mind metrics scheme was proposed to strengthen the identification process with personal secret information. We concentrated on adding extra security to identification as well as verification processes. In identification process user had to submit a mind-metric token. Token helps system to identify whether user is legitimate or not. On submitting token user is provided with partially obscured login id, legitimate user can easily select his/her id.

To strengthen the process of verification, the concept of Two-server password is used. User password is encrypted and split into two parts, and stored in two different servers. Thus if attacker is able to hack one server, then also he will not be able to access user's account.

We had also provided a OTP facility in verification process. After the identification process user will be provided by a OTP code which will be used for moving further.

After the OTP, the user will be provided by a generalized image, and a question will be asked to the user, the answer of this question has to be already submitted by the user at the time of registration. So, for the verification, the answer given by user at the time of login, must be matched with one which is already stored in server.

Thus in turn by enhancing identification and verification process authentication of system is strengthened.

REFERENCES

[1] Wen-Bin Hsieh Dept. of Electron. Eng., Nat. Taiwan Univ. of Sci. & Technol., Taipei, Taiwan Jenq-Shiou Leu, " Design of a time and location

based One-Time Password authentication scheme ,“ Wireless Communications and Mobile Computing Conference (IWCMC), 2011 7th International on July 2011.

[2] Xun Yi, San Ling, and Huaxiong Wang, ”Efficient Two-Server Password-Only Authenticated Key Exchange,” IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 24, NO. 9, SEPTEMBER 2013.

[3] Flonta, S. Tech. Univ. of Cluj-Napoca, Cluj-Napoca Miclea, L., “An extension of the El Gamal encryption algorithm,”http://en.wikipedia.org/wiki/ElGamal_encryption, Automation, Quality and Testing, Robotics, 2008. AQTR 2008. IEEE International Conference on may 2008.

[4] Alzomai, M.; Queensland Univ. of Technol., Brisbane, QLD; Josang, A.; McCullough, A ; Foo,E.” Strengthening SMS-Based Authentication through Usability,” Parallel and Distributed Processing with Applications, 2008. ISPA '08. International Symposium on Dec 2008.

[5] W. Diffie and M.E. Hellman, “New Directions in Cryptogra- phy,” IEEE Trans. Information Theory, IT-22, no. 6, pp. 644-654, Nov. 1976.

[6] Juyeon Jo; Dept. of Comput. Sci., Univ. of Nevada, Las Vegas, NV, USA ; Yoohwan Kim ; Sungchul Lee ” Mind metrics: Identifying users without their login Ids” Systems, Man and Cybernetics (SMC), 2014 IEEE International Conference on Oct 2014