International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 04 Issue: 04 | Apr -2017 www.irjet.net p-ISSN: 2395-0072

A Two Factor Authentication System for Touchscreen Mobile Devices Using Static Keystroke Dynamics and Password

Javed Miya[1], Mayur Bhatt[2], Mayank Gupta[3], Mohammad Anas[4]

[1]*Assistant Professor, Dept. of IT, Galgotias College of Engg. & Tech., Greater Noida – 201306, U. P., India*
[2, 3, 4] *Student, Dept. of IT, Galgotias College of Engg. & Tech., Greater Noida – 201306, U. P, India*

Abstract - *The number of touchscreen mobile devices are rapidly increasing each day and so are the number of people that use them. Username-password combination is the most common method of authentication but has many vulnerabilities like shoulder surfing, social engineering, brute force attacks, key-loggers, etc. Keystroke Dynamics provides a novel approach to strengthen this existing method. Typing rhythm or keystroke of every user is distinct in the same way as signatures of users are distinct. Keystroke Dynamics is a biometric but unlike other biometrics it does not need any special hardware. In this paper, we have proposed a simple yet elegant method for touchscreen mobile devices which uses keystroke dynamics as first factor authentication and a secret password as second factor authentication to grant access to the mobile device.*

Key Words: **Keystroke Dynamics, Typing Rhythm, Two Factor Authentication, Biometric Authentication, Static Keystroke**

1. INTRODUCTION

The number of touch screen mobile devices has increased rapidly over the past few years which has contributed to the rise of access and storage of confidential information in these devices. This rapid increase has also led to more sophisticated attacks like shoulder surfing, social engineering, etc [1]. Hence, a need for a better secured authentication process arises. Most access systems these days prompt users to authenticate themselves with a username and password combination. This method of authentication relies on the password's secretiveness and in some cases, even the username's secretiveness. If this secretiveness is not breeched, the assertion is that this pair uniquely identifies a user. Number of problems are associated with maintaining password secrecy like passwords having birthdays, anniversaries, common words or terms associated with a particular user which are universally considered vulnerable since it is easy for an attacker to guess them or find via social engineering attacks. Some systems require users to remember complicated passwords or usernames - the more complicated, the better. Of course, complicated also usually means "difficult to retain" which is a usability liability [2]. This is challenging for the users because not only must they choose complicated passwords, but they also must choose them repeatedly, since different systems have different rules for username as well as password generation. A user is left to have many username-password pair. This is done to ensure that the compromise of a single password doesn't compromise them all. In practice though many users are overwhelmed by remembering so many unique and complicated username-password pairs that they don't fully comply with guidelines provided by the system – typically, using the same or very similar password for all accounts. However, even if they do follow the best-recommended practices, passwords are still easily stolen by hackers, whether transferred inadvertently or not: users sometimes write passwords down on a piece of paper, store them in easily recognizable text files, and accidentally expose them by entering them in the username field and so on.

A new class within biometric authentication is getting traction these days called behaviometrics [3]. Behaviometrics are the characteristics of a person that are related to the pattern of behavior of that person [4], including but not limited to their typing rhythm, gait, handwriting and voice. In this paper, our main focus is on keystroke dynamics which is a behaviometrics. Keystroke Dynamics can be defined as a process of monitoring and analyzing the user's typing rhythm through the on-screen touch keyboard in order to authenticate the users [5]. It is considered as an interesting behavioral authentication solution for many reasons. First, contrary to other biometric features, keystroke dynamics is inexpensive to implement as it requires only the standard touch screen keyboard available on all mobile devices, which means unlike other biometric methods, keystroke does not require any additional specialized hardware. Second, it is user-friendly and nonintrusive [6]. Third, the typing rhythm of the person cannot be lost or forgotten. If the pattern is stolen or guessed, the user can produce another one easily. So it is the only resettable biometric. Researchers are continually trying to implement something new in this field since last two decades. Plenty of researchers have come up with novel approaches to overcome the vulnerabilities that are present in current authentication systems. A two factor authentication approach has been applied in this paper, namely, keystroke or typing rhythm and traditional password.

2. KEYSTROKE DYNAMICS

Of late, with the rise of smartphones and number of applications that require sign-ins to access them, biometric authentication has gained traction. Finger print scanner is the most used biometric [7] nowadays but there is need for a less expensive biometric measure that can be universally used with all smartphones. Keystroke dynamics is the process of recording ones typing rhythm and then using it to authenticate users. The typing rhythm or pattern of a user is extracted using the time values (usually in milliseconds) of key presses by the user. Many more features like pressure, orientation, etc are also used to define a person's typing rhythm but time values of key pressed are the most important feature [8]. Time values like duration of time a particular key is pressed called 'down-up' time, the idle duration of time between two consecutive key presses called 'up-down' are recorded and saved to help recognize a user using their typing rhythm. There are mainly two types of keystrokes, static keystroke recognition and dynamic keystroke recognition.

2.1 Static Keystroke

When typing rhythm of the user is only matched before entering the system and only on few particular tokens it is termed as static keystroke recognition [9]. It is termed static because there is no continuous monitoring of the user. No authentication takes place after the user has been granted access to the system. Static keystroke recognition is much more simple and straight forward than dynamic.

2.2 Dynamic Keystroke

When the typing rhythm of the user is continuously recorded and analyzed to make sure that the same user is using the system whom access was granted is called dynamic keystroke recognition [10]. It is much more complicated and sophisticated than static and needs lot more resources to give decent results. Dynamic keystroke recognition is best suited for high security system that have critical information.

3. PROPOSED SYSTEM

In this section we discuss the proposed methodology for the two factor authentication system. The objective of the system is to authenticate users based on the combination of habitual patterns of their typing rhythm on their mobile device and a simple secret text password. We have applied straightforward approach for this purpose. The two phases, 'create account' and 'login' are used by the user to authenticate themselves by the system. In 'create account' phase, the primary functions are data capture and feature extraction. Android has key events ACTION_DOWN and ACTION_UP which are used to store timing values of key pressed [11]. We will use three features namely consecutive key pairing, pair vector and tolerance to store and later recognize the typing pattern of each individual user. System will store the keystroke times in username, email id and mobile number, with the user's other credential details like first name, last name and password in a database. Login phase takes place whenever a user needs to access the system. The proposed method has been organized in the following three sections. Section 4.1 describes the create account phase, section 4.2 deals with the login phase and section 4.3 explains the background typing comparison process. Following are the proposed parameters:

1. Consecutive Key Pairing (CKP): time between press down of two consecutive keys in milliseconds.

2. Pair Vector: A dynamic array data structure which stores the values of CKP

3. Tolerance: A predefined tolerance value for each CKP.

Below is the flow chart of the entire system:

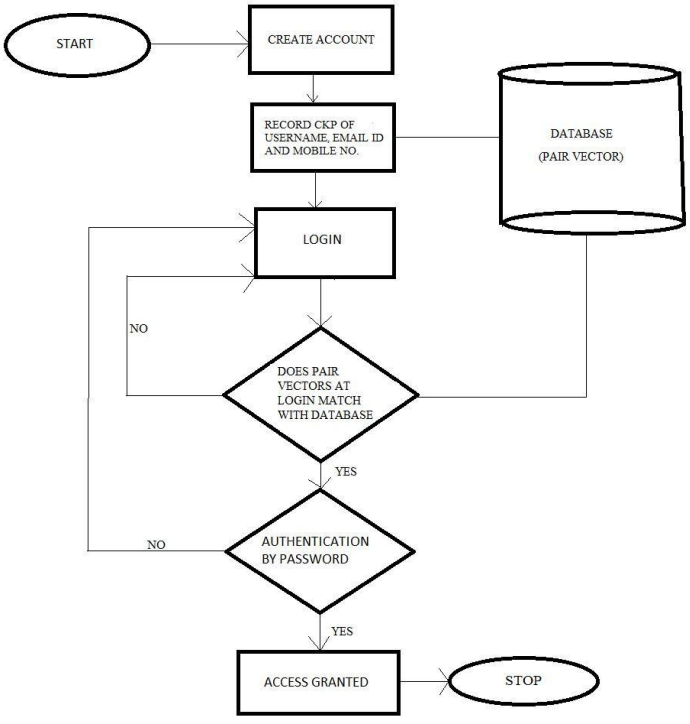

Fig -3.1: Flowchart of proposed system

4. IMPLEMENTATION

4.1 Create Account Phase

A user first creates his/her account by providing First name, Last name, Username, Password, Email Id, and Mobile Number. Typing pattern is registered in Username, Email Id

and Mobile number via CKP and stored as Pair Vector. User then needs to submit this data if all fields are filled correctly, his/her data is saved.

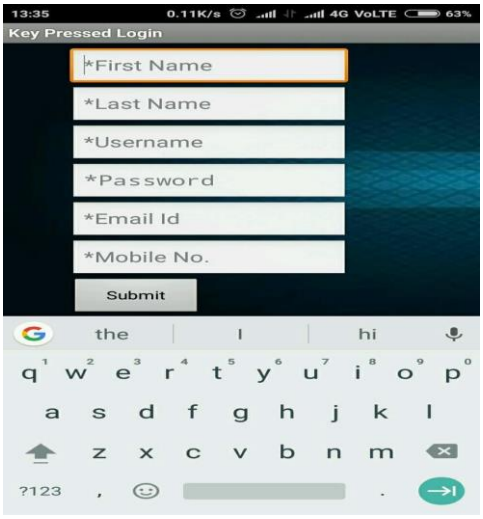

Fig -4.1: Screenshot of screen account phase

4.2 Login Phase

The login phase realizes the authentication procedure. Correct username and password does not ensure authentication of a user. The parallel typing verification is the main concern of our proposal. In the first step, a user is asked for either username/ email id/ mobile number, which is chosen at random. If the character by character rhythm is not correct, the user is denied access, and the application closes itself for better security otherwise the user is led to the next phase of authentication. In the next phase, a password is input. The password is a string input and the password string in the databases are matched. If both match, access granted else denied. When both the factors are correctly entered by the user, system access is granted. If even one factor is incorrect, the system denies access to the user.

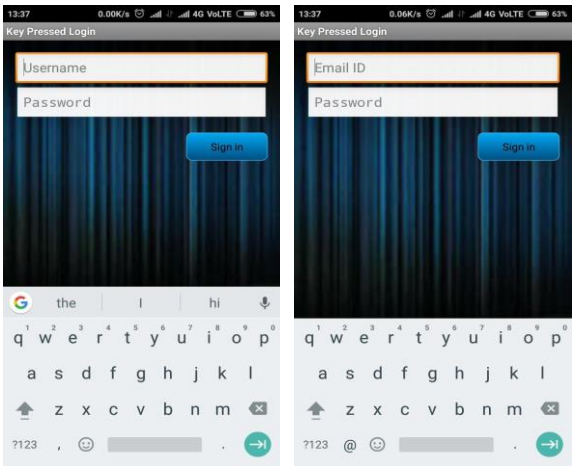

Fig -4.2: Screenshot of Login Phase

4.3 Background Typing Comparison Process

Let us suppose that the user types "RAMADHIR" as his username for authentication.

Step 1: During Login phase, record Consecutive Key Pair (CKP) values for Username/ Mobile No. / Email. In our example, user types "RAMADHIR". CKP for this are shown in figure below

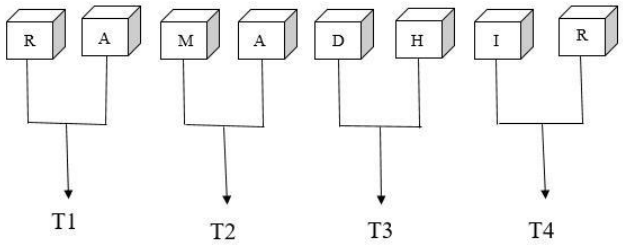

Fig -4.3: Consecutive Key Pair values for "RAMADHIR"

Step 2: Each Vector Pair, i.e., T1, T2, T3, T4 ±tolerance value is checked with the vector pair in database. For example if vector pair for RAMADHIR is [212, 180, 192, 184] and tolerance is set at 100 in database then this pair vector range [212 ±100, 180 ±100, 192 ±100, 184 ±100] in login phase will successfully pass first factor of authentication and move onto second authentication. If pair vectors are outside this range then the application closes and user needs to re-login.

Step 3: Second factor authentication is traditional password. The user needs to enter their password, if it matches with the one in database, access is granted else denied and user needs to re-login.

5. DISCUSSION OF RESULTS

For our system, we have used java for coding since it contains in-built functions for reading keyboard events like key-press, key -release, etc. and used it to make an android application. We have used two standard parameters False Acceptance Rate (FAR) and False Rejection Rate (FRR) to measure the efficiency of our proposed system. The FAR and FRR can be defined as the following:

FAR = (Number of incorrect attempts accepted by the system / Number of incorrect attempts)*100

FRR = (Number of correct attempts rejected by the system / Number of correct attempts)*100

We always expect very low value of these parameters to make the system highly efficient [12]. Both FAR and FRR directly depend on the tolerance value considered to verify a user's typing pattern. The performance result reported here is based on the database created for 1020 attempts. We asked 120 persons to register themselves to our application

on their personal mobile devices. Participants created their profile and attempted to login 9 times each from their own machine at their convenience. All of the participants were informed about the purpose of the experiment. It was necessary for the participants to focus on their keystroke. The overall result analysis of our experiment is shown in table 5.1.

Table -5.1: Results of the experiment

Input Type	No. of attempts	Accepted	Rejected	FAR	FRR
Valid	1080	1046	34	---	3.15 %
Invalid	1080	40	1040	3.70 %	---

We can see that, out of total (1080+1080) = 2160 predictions made by the system, (1046+1040) = 2086 decisions are correct. So the accuracy of the system is (2086÷2160) ×100 = 96.5%.

Table -5.2: Results of "Authentication through Keystrokes: What You Type and How You Type" [13]

Input Type	No. of attempts	Accepted	Rejected	FAR	FRR
Valid	1000	920	80	---	8%
Invalid	1000	50	950	5%	---

As we can see FAR, FRR values are lower and overall accuracy is higher for table 5.1 than table 5.2. Hence we conclude that the proposed methodology is better.

6. CONCLUSIONS

We can conclude from the study that our method is quite simple since it is based on statistical approach and provides interesting results with more than 96.5% accuracy. We believe that this methodology will successfully defeat many attacks, which conventional password protection mechanisms fail to vanquish. But there are still some potential problems in our scheme which need to be mentioned. First, we have studied the characteristics of keystroke dynamics for users in their own machines. So, typing speed in other devices may differ for the same user. Second, the registration process is time consuming and boring for the users. Our experiment does not deal with these situations. However, there are many scopes in the proposed systems which can be considered for further improvements. One of the ways of improving the system's performance is to update the user typing profile with time. Another way to improve the result is to find the reasonable threshold by using some kind of mathematical analysis. The use of smartphones, tablets and other touch screen devices are gaining popularity with an unbelievable pace in recent times. It will be very useful if we are able to make a single application for all these devices.

REFERENCES

[1] Ramzi Saifan, Asma Salem, Dema Zaidan, Andraws Swidan, "A Survey of behavioral authentication using keystroke dynamics: Touch screens and mobile devices", Journal of Social Sciences (COES&RJ-JSS), 5(1), Pg 29, Jan 2016.

[2] [2] A. Peacock, X. Ke, and M. Wilkerson, "Typing patterns: a key to user identification", IEEE, Security Privacy, 2(5): Pg 40, Sep. 2004.

[3] Jiang Zhu, Haol Hun, Sky Hiu, "Mobile Behaviometrics: Models and applications", Communications in China Journal, 2013.

[4] D.Y. Liliana, D. Satrinia, "Adaptive Behaviometrics using Dynamic Keystroke for Authentication System", International Conference on Future Information Technology IPCSIT vol.13 IACSIT Press, Singapore – 2011.

[5] Fabian Monrose, Aviel D. Rubin, "Keystroke Dynamics as a Biometric for Authetication", Future generation computer systems, Elsevier – 2000.

[6] H. Saevanee, P. Bhattarakosol, "Authenticating User Using Keystroke Dynamics and Finger Pressure", IEEE, Networking Conference, Issue 14, Vol 3- 2009.

[7] [7] G. Aishwariya, S. Kokilapriya, S. Adhithya, Dr. A. Grace Selvarani, "Fingerprint Recognition for Android Application Data Retrieval", IJSRST, Volume 3, Issue 1, Print ISSN: 2395-6011, Online ISSN: 2395-602X – 2017.

[8] E A Kochegurova, E S Gorokhova, A I Mozgaleva, "Development of the Keystroke Dynamics Recognition System", Journal of Physics: Conference Series, Volume 803, Number 1, Pg 264 – 2017.

[9] Baljit Singh Saini, Navdeep Kaur and Kamaljit Singh Bhatia, "Keystroke Dynamics for Mobile Phones: A Survey", Indian Journal of Science and Technology, Vol 9(6), DOI: 10.17485/ijst/2016/v9i6/82084 – Pg 32, 34 - February 2016.

[10] Bergadano, F., Gunetti, D., & Picardi, C, "User authentication through Keystroke Dynamics", ACM Transactions on Information and System Security (TISSEC), 5(4), 367-397, 2002.

[11] Ed Burnette, "Hello, Android Introducing Google's Mobile Development Platform, 3rdEdition" – Pg 227, 2004

[12] Md. Asraful Haque, Namra Zia Khan, Gulnar Khatoon, "Authentication through Keystrokes: What You Type and How You Type", IEEE International Conference of Research in Computer Intelligence and Communication Network, Pg 258 – 2015.

[13] Md. Asraful Haque, Namra Zia Khan, Gulnar Khatoon, "Authentication through Keystrokes: What You Type and How You Type", IEEE International Conference of Research in Computer Intelligence and Communication Network, Pg 260 – 2015.