User Authentication using Keystroke Dynamics

Vishesh Mishra
School of Computer Science and Engineering(SoCSE)
University of Petroleum and Energy Studies
Dehradun, Uttarakhand, India

Raghav Gupta
School of Computer Science and Engineering(SoCSE)
University of Petroleum and Energy Studies
Dehradun, Uttarakhand, India

Gautam Sood
School of Computer Science and Engineering(SoCSE)
University of Petroleum and Energy Studies
Dehradun, Uttarakhand, India

J. C. Patni
School of Computer Science and Engineering(SoCSE)
University of Petroleum and Energy Studies
Dehradun, Uttarakhand, India

ABSTRACT

“User Authentication Using Keystroke Dynamics”. It is a method to get the user authentication on an android application by using the keystroke dynamics of the user using Artificial Neural Networks with the help of Error Back Propagation algorithm. In this application the user enters the password 30 times and databases are used to record 45 factors that describes a user’s keystroke patterns like di-graph, dwell time, tri-graph, flight time, finger size, button pressure, coordinate values which can be seen by the user in real time. Once this is done the data is taken and put in an Artificial Neural Network and trained using Error back propagation Algorithm. This process done over time produces trained set off weights that would produces an already calculated value in the output layer. This data from the network is again stored in a separate table which is then used to check the authentication of the user typing the password.

Keywords

Artificial Neural Networks, Error Back Propagation Algorithm, Keystroke Dynamics, Database Systems, Mobile Phones, Passwords, PINS (Personal Identification Numbers), Android, Biometric Authentication.

1. NOMENCLATURE

Di-graphTime elapsed between releasing the key and pressing the next key.Dwell time how long we press a key. Tri-graph latency between three keys. Flight time how long we take to type successive keys Finger size- Amount of space occupied by finger touch. Button pressure Finger pressure applied while typing Coordinate values Location of Button pressed.

2. INTRODUCTION

As mobile technology is becoming more powerful and sophisticated it is very clear that in the coming time it would be the most universally owned device in the world. Mobile phone is becoming more and more personalized and are collecting dozens of parameters that pertain to our personal data. In this case it is very important to ensure that systems are built so that this data is not stolen or misused by someone. Our project deals with this problem. There are basically three types of authentication system that can be put to use to ensure safety over the internet. The first method in to use Passwords and PINS (Personal Identification Numbers). This method is not as successful as one might think. There are several ways to go around this system as hackers can use social as well as cyber hacking to penetrate this system can steal the passwords. The second method is the Token authentication which is basically an object possessed by the intended user which is passed hand to hand and is more likely to be stolen on the network. The third and the most secure method is the Biometric authentication, this system is better because it uses characteristics that are unique to an individual and cannot be stolen very easily. Fingerprints are from this category but are not a perfect solution as although with relative difficulty, it is still possible to steal the fingerprints of the user and use it to gain access to the system. Hence, we decided to use keystroke dynamics. In this process the hacker cannot copy the individual characteristic of the user as the system is relying on the keystroke pattern which is net to impossible to steal as various factors to measure keystroke pattern of the user like keystroke time, pressure is kept encrypted within the device. This information is captured using various sensors like pressure pads on the mobile device.

3. LITERATURE REVIEW

Authentication via Keystroke Dynamics is quite old. The earliest studies on this method were performed in the years 1985-1990 for desktop keyboards. D. Umphress and G Williams are among the first to write on the subject of using keystroke patterns to identify a user. Their work explains the concept and introduces the idea of latency (also called digraphs), the amount of time between two keystrokes. Much of the later work on the subject used the idea of latency and it is used even today in many Keystroke Dynamics Analyzers (KDAs). KDAs were also deployed for pre-touchscreen mobiles (the ones with 12 keys). With the mainstream adaptation of capacitive touch screens and consequently convenient touch keyboards on mobile phones in 2007, this method has been applied to touchscreen phones as well [1].

Most of the keystroke mechanism depends on password authentication via calculating the time interval between the single keystroke press. This parameter is one of many that can be used for keystroke dynamics and measuring and setting the security parameters. But in addition to this we can also take the record of other parameters like time interval between two key presses. In such calculation four timestamps are associated namely first key press (P1), first key release (R1), second key press (P2) and second key release (R2). Thus, leading to six combinations of time differences of the above four timestamps, namely, d1: R1-P1, d2: P2-R1, d3: R2-P2, d4: P2-P1, d5: R2-R1, d6: R2-P1. Every user has his unique typing style. Typing styles are the most efficient way of collecting the data between user and system [2]. A lot of work has been done in terms of making the biometric authentication or reading the keystroke dynamics less costly for the end user. Over the years the study has advanced as technology has and the research has gone from reading keystroke dynamics from computer keyboards to reading it from the mobile devices using several pattern classifiers like artificial neural network. But there has never been a study or a case method to classify the results from the effectiveness of different models. The presence of such models would allow us to test the effectiveness of such algorithms and then improve them [3].

With the rise of computing and machinery handling data of people in the day to day life, the lives of people have been increasingly exposed to hacker who can penetrate system. One of the methods to make the system secure is to use biometric authentication. In ord7er to guide the research of this technology in the future there needs to be a systematic review, to make the correct protocols to assess these researches as they move forward [4]. More then ever before the internet is changing computing and with it are all types of problems being faced by users across the globe. One of them is the stealing of personal data. One of the methods is to authenticate the user using biometric authentication. The method can access the user's information on keystroke dynamics and use artificial neural networks to train the data to authenticate the user [5].

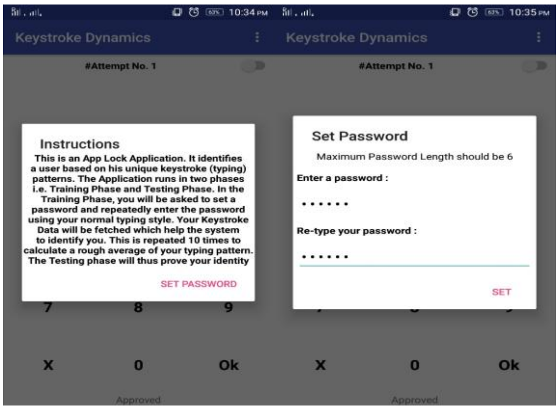
In [6] the proposed approach diminishes security danger by a host based client profiling system where a key stroke dynamics is utilized for investigating the client conduct and a retraining approach is additionally proposed as the imposter patterns are absent at the session of enrollment. Retraining supports the general framework execution in moderating the insider risk. In short it is giving security to cloud from insiders attack. The proposed work utilizes a Support Vector Machine (SVM) which is a standout amongst other known characterizations. The two primary strategies that utilized here are FRR and FAR.[7] have four measure class for grouping, which are, Statistical Algorithm, Artificial neural Network, Pattern Recognition, and learning Algorithm. For measurable or statistical approach, Bayes order Classification, in light of back likelihood of events is utilized. For Pattern recognition, closest neighbor strategy is utilized. For machine learning, SVM and K-Mean strategy is utilized. In this three models are utilized for keystroke progression. To begin with is channel show, it requires qualities of information to assess and select the element without including any learning calculation. Second model wrapper display, it utilizes learning calculation for the execution and assessment of the information. Third model is hybrid model, it takes the benefits of channel and wrapper model.

In keystroke dynamic based confirmation, curiosity discovery strategies are utilized since just the substantial clients design are accessible. At the point when a classifier is first built sooner or later, in any case, artificial keystroke designs wind up accessible from fizzled login endeavors. [8] propose to utilize the retraining system where a curiosity locator is retrained while the already given examples to improve verification exactness.

4. ARCHITECTURAL DESIGN

Here we have designed an android application that consists of the ANN and various other components to both read keystroke values as well as authenticate the user. There are three functionalities that are provided to the user when using this android Application.

The first one is the training part. Here the user gets to type in his password to set it first and then is asked thirty times the same password in order of the system to capture the information on the user typing pattern (Keystroke Dynamics) and store that information in database.



Instructions

This is an App Lock Application. It identifies a user based on his unique keystroke (typing) patterns. The Application runs in two phases i.e. Training Phase and Testing Phase. In the Training Phase, you will be asked to set a password and repeatedly enter the password using your normal typing style. Your Keystroke Data will be fetched which help the system to identify you. This is repeated 10 times to calculate a rough average of your typing pattern. The Testing phase will thus prove your identity

Set Password

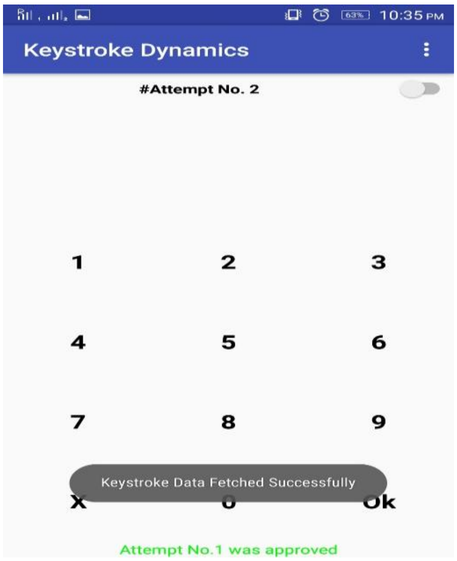
Maximum Password Length should be 6

Enter a password : ••••••

Re-type your password : ••••••

Fig 1: Enter Password screen

This information is basically stored as 45 different datasets that mimics the users typing pattern. As the user is doing this he/she can decide to look at the values that are being recorded by opening the concerned page at the click of a button in the user interface.

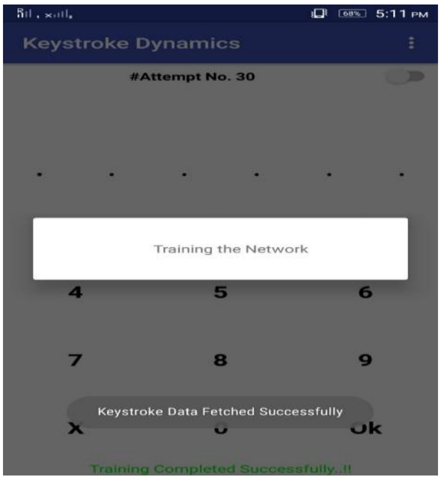


Keystroke Data Fetched Successfully

Attempt No.1 was approved

Fig 2: Keystroke Dynamics

The second functionality the user has is to access this information which is displayed in a separate page that shows all the values that are being recorded like pressure points, coordinates of the button, time taken to press single or multiple buttons and many more. The third functionality would be the ability for the android application to authenticate the user. Here the user enters the password and if the previous keystroke data saved matches the data recorded when the user types the password a token is released saying the user is legitimate. If the data is not matched then the user is deemed illegitimate. To do so there needs to be some training to be done on the typing pattern of the user. This is done using our artificial neural network and an algorithm that is applied on the network called Error Back Propagation algorithm. The data from the user is fed into our artificial neural network and then weight function is applied on the network.



Keystroke Dynamics

#Attempt No. 30

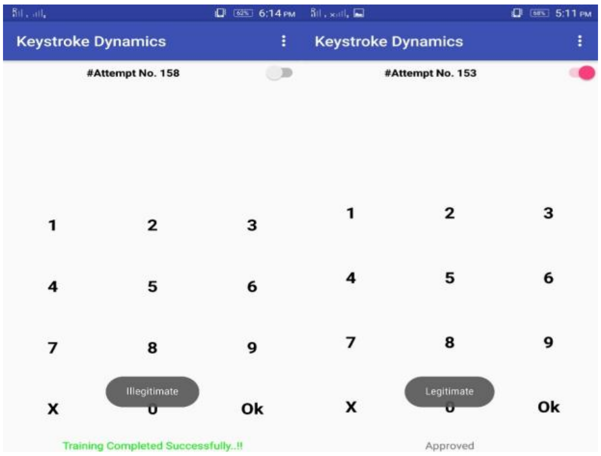
Training the Network

Keystroke Data Fetched Successfully

Training Completed Successfully..!!

Fig 3: Training the pattern

Each time new value is generated on the single node in the output layer, it is matched with the already calculated value presented by the Error Back Propagation algorithm. Every time the data is sent back to the network the weights are self-adjusted till the output value is same as the calculated value. At this point the values of the trained weights are sent to another database system to be stored and when the user types his password the pattern in registered and the trained weight values are applied on it over the network and if the output value is same as the calculated value then the user is authenticated otherwise not.



Keystroke Dynamics

#Attempt No. 158

Illegitimate

Training Completed Successfully..!!

Keystroke Dynamics

#Attempt No. 153

Legitimate

Approved

Fig 4: User Attempts

We have used here three separate databases. One for the training values to be stored. Other for saving the password and a third for the purpose of storing the trained weights from the artificial neural network. The database was connected to the program using the SQLite database connectivity. The entire program was developed on the Android Studio. The program is divided into two parts, one is to implement the user interface part of the system and the other is to take the data, store it and then apply ANN and the algorithm to generate the trained weights. The major part of the program and almost all the logic was written in JAVA and the user interface was written in both JAVA as well as XML.

5. PROPOSED SYSTEM

We have used Neural Networks here for the training of datasets. SQLite databases are used in this application for storing keystroke values, training data and calculating weights. The database was connected to the program using the SQLite database connectivity API. The entire program was developed on the Android Studio. The program is divided into two parts, one is the user interface part of the system and the other is to take the data, store it and then apply ANN and the algorithm to generate the trained weights. The major part of the program and almost all the logic was written in JAVA and the user interface was written in both JAVA as well as XML. The application requires an Android 5.0 or latest. Integrated Development Environment (IDE) used here is the Android Studio with Nexus 5x API 24 emulator. You need at least 10MB of hard disk space and about 256MB of RAM.

6. WORKFLOW

The following workflow describes the functioning of the application:

1. Access the application on an android mobile device

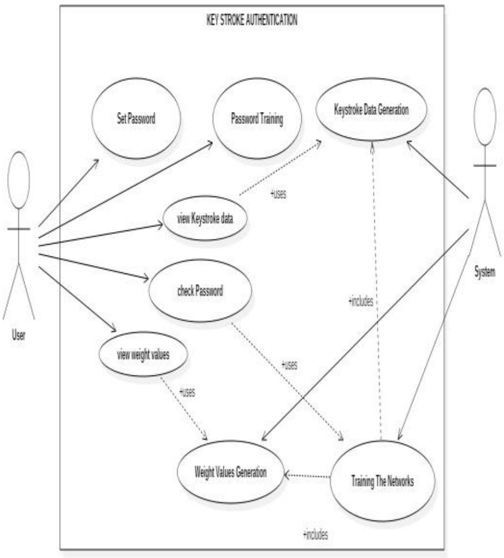


Fig. 5. Use Case Diagram

2. On starting the android application user is shown the Instruction Layout with all the basic details on the usage of the application.

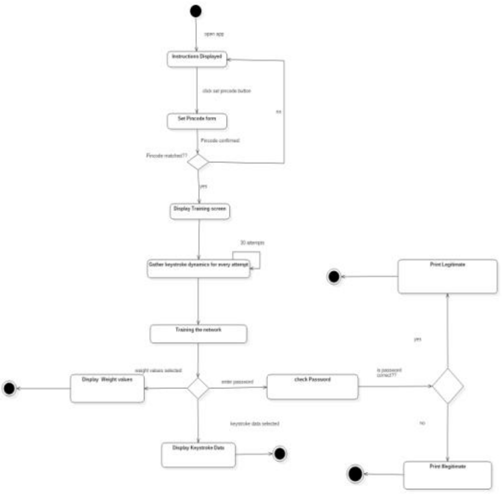


Fig. 6. Activity Diagram

3. Now the user is asked to enter the passcode of 6 digits that will be further used in training the data.

4. This is done thirty times in order to get the keystroke style and the typing pattern of the user.

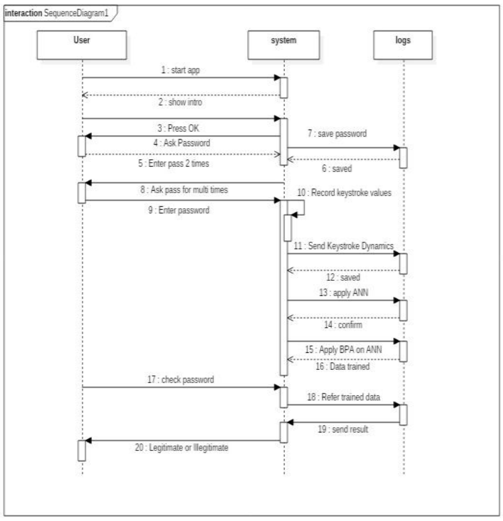


Fig. 7. Sequence Diagram

5. The Keystroke data from above 30 attempts is calculated.

6. User can now choose to view the Keystroke data that is being calculated on every attempt made on setting the password as Test Cases.

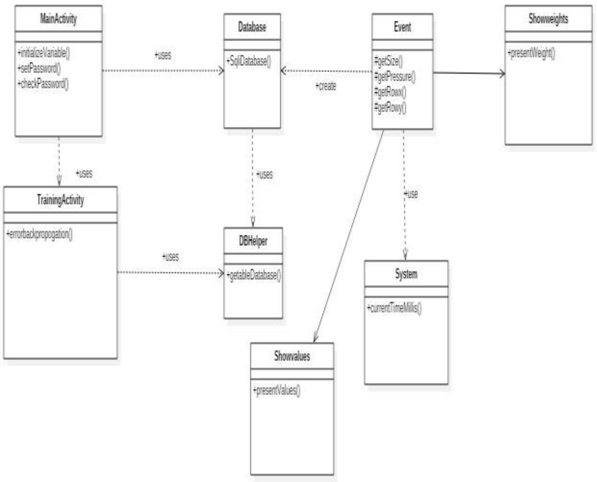


Fig. 8. Class Diagram

7. After all the successful attempts weight values will be calculated for every attribute and will be feeding the data into the neural networks to train the data.

8. Data will be processed for the training and thus validation can be performed.

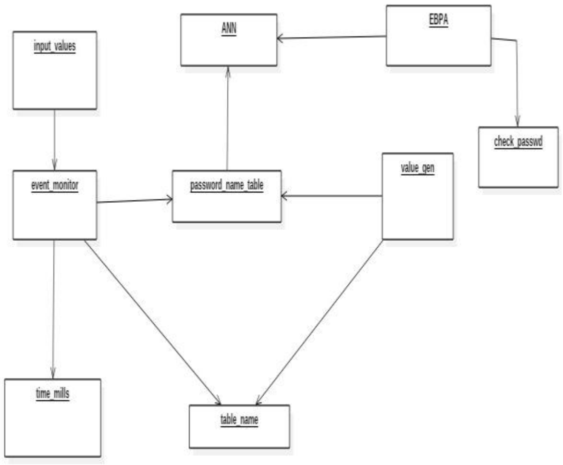


Fig. 9. Object Diagram

7. MATH AND EQUATIONS

Range Upper Limit = Mean + (X * SD) (1)

Range Lower Limit = Mean – (X * SD) (2)

Deviation = (Test value of the feature-Mean value of the feature) ÷ Mean value of feature (3)

Absolute Distance = 1 – Number of similar diagraphs ÷ Total common diagraphs (4)

Calculation in ANN:

$Z_{in} = v_{0j} + \sum^{n}_{i=1} x_i v_{ij}$ (5) (input layer function)

$Z_j = f(z_{in\ j})$ (6) (activation function)

$f = 1 \div (1+e^{-x})$ (7) (sigmoid function)

$Y_{ink} = w_{0k} + \sum_{j=1}^{p} z_j w_{jk}$ (8) (output layer function)

SD – STANDARD DEVIATION

8. RESULTS

The result of the final development is an android application that has the ability to use keystroke dynamics the be able to authenticate the user. The user here has to enter the password once this is done they are asked for their more times to record in their dataset the keystroke of the person, then the user's dataset is trained accordingly and once this set of work is done the user is able to log into their system very easily

9. FUTURE WORK

Suggested below are some suggestions as to what other features can be added to the system to make it more efficient:

i. **Encryption:** There are ways to steal the trained weights of the user from the application. This can be prevented by encrypting the data and only decrypting them when in use this allows us to make the system more secure.

ii. **Hand written authentication:** The user is in this case using PINS or passwords to authenticate themselves but we can also use the handwriting of the person to make the system more complex for them to understand and use. This means that now we have to take more into account not just the existing parameters but also other parameters like the handwriting and the way in which the user makes strokes or how much pressure is applied by them while writing the password on the screen pad.

iii. **Better Algorithm:** One of the things that can be done is to use better algorithms to train the Artificial Neural Networks thus allowing us to improve our accuracy.

iv. **Multiple User Authentication:** There are sometimes when parents leave their mobiles for children to call them, other times you may want to access multiple devices without setting up the password. The problems here are different but the concept is the same, that is the ability of the user to be able to authenticate themselves over multiple devices just the same way google account allows us to do so over our laptop and desktop at the same times. Also to authenticate multiple users on the same device requires us to save the images of several persons thus resulting in more databases and more training time and also the ability to differentiate one dataset from another.

v. **Better Accuracy:** There are cases where the user might be the person but due to certain reasons pertaining to their current situation like they might be in a hurry or something else might lead to then using a very different way of typing thus not authenticating them at all. This problem can only be solved with increase number of dataset and more and more training required.

10. CONCLUSIONS

Brain is an android application that is used to authenticate a user based on their keystroke dynamics, that is their pattern of typing. Once this information is recorded a certain number of times (thirty), the information can then be used to identify the user. The application has a artificial neural network with forty five neurons in the input layer and one neuron in the output layer that produces the value that is to be compared with an already calculated value that is generated by the Error back algorithm which is applied on the artificial neural network. This is done over and over till we get the right value and then the weights on the network can be used to authenticate the user next time the password is typed in, thus providing a more secure and authentic system to save your data.

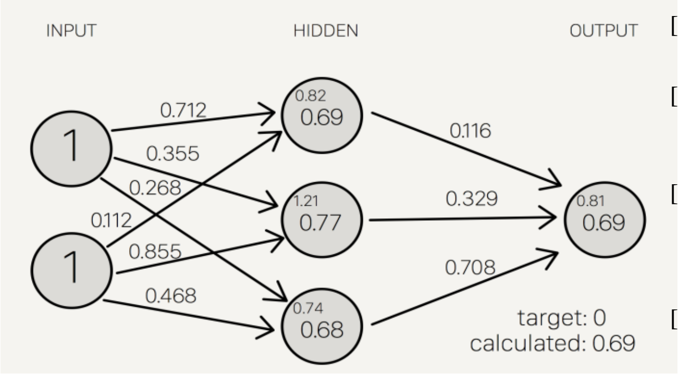


Fig. 10. Artificial neural Network

11. ACKNOWLEDGMENT

vi. We wish to express our deep gratitude to our guide Dr. J.C Patni, for all advice, encouragement and constant support he has given us throughout our project work. This work would not have been possible without his support and valuable suggestions. We sincerely thank to our respected Program Head of the Department, Dr. Ajay Prasad, for his great support in doing our project in Keystroke Dynamics at CIT. We are also grateful to Dr. Kamal Bansal**,** Dean CoES and Dr. Manish Prateek, Director SoCS UPES for giving us the necessary facilities to carry out our project work successfully. We would like to thank all our peers and mentor for their help and constructive criticism during our project work. Finally, we have no words to express our sincere gratitude to our parents who have shown us this world and for every support they have given us.

12. REFERENCES

[1] J. D. Umphress and G Williams, “Mobile Authentication Using Keystroke Dynamics”, International Conference on communication, Jan 2015

[2] Mayur Mahadev Sawant, Yogesh Nagargoje, Darshan Bora, Shrinivas Shelke and Vishal Borate, “Keystroke Dynamics: Review Paper”, International Journal of Advanced Research in Computer and Communication Engineering Vol. 2, Issue 10, October 2013.

[3] Heather Crawford, Keystroke Dynamics : Characteristics and opportunities, Privacy Security and Trust (PST), 2010 Eighth Annual International Conference, Aug 2010

[4] Pauli Henrique Pisani, Ana Carolina Lorena, A systematic review on key stroke Dynamics, Journal of the Brazilian Computer Society, Nov. 2013

[5] Fabian Monrose, Aviel D. Rubin, Keystroke dynamics as a biometric for authentication, Feb 2000

[6] Mahesh Babu B, Mary SairaBhanu UNIVERSITY, “Analyzing User Behaviour Using Keystrokes Dynamics to Protect Cloud from Malicious Insiders”, IEEE 2016

[7] Rohit A Patil,Amar L. Renke, “Keystroke Dynamics for User Authentication and Identification by using Typing Rhythm”,International Journal of Computer Applications vol. 144, nos. 9, June. 2016.

[8] Hyoung-jooLee,Sungzoon Cho, “Retraining a keystroke dynamics-based authenticator with impostor patterns”, in ELSEVIER.