INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH & DEVELOPMENT

ISSN 2278 – 0211 (Online)

Keystroke Dynamics Authentication System Using Neural Network

Sonali B. Wankhede
4th Sem. ME (Computer Engineering)
Thadomalshahani Engineering College, Mumbai, India
Shilpa Verma
Associate Professor, Computer Engineering Department
Thadomalshahani Engineering College, Mumbai, India

Abstract:
The fact that computers regularly store private, sensitive and classified information makes it very important that we can confidently identify their users. Traditionally, this has been achieved through password authentication systems. However, these systems are far from perfect. For instance, if a password becomes compromised, it is no longer adequate for authenticating its rightful owner. In the hope of improving on this, there exists ongoing research into utilising the idiosyncrasies of a user's interaction with a computer as a form of authentication. So far in this field the most promising techniques focus on patterns in the timing of a user's typing. We shall refer to this as 'biometric keystroke authentication'. This paper focuses on the time interval between keystrokes as a feature of individual's typing patterns to recognize authentic users and reject imposters. A Multilayer Perceptron (MLP) neural network is used to train and validate the features. The classifier is used to analyse the features of the user. Authentication of a user is accomplished using a classifier and appropriate adaptation of the user sample is introduced upon successive authentication.

***Key words**: Keystroke Biometrics, user authentication, neural network, key time intervals*

1. Introduction

Biometrics measure individuals' unique physical or behavioural characteristics to recognise or authenticate their identities. Biometrics offer to inextricably link the authenticator to its owner, something passwords and tokens cannot do, since they can be lent or stolen. In terms of combining with existing systems, much research has gone into investigating the viability using the typing behaviour present upon the entry of password credentials as an additional layer of authentication. This technique could potentially overcome the shortcomings of passwords, as not only must the password be known, but it must be entered in the manner of the legitimate user. While this technique is promising, there has been little work that takes into account how a user learns to type credentials. Yet acknowledging this learning process could be a very important feature of a viable biometric keystroke authentication system [1].

Keystroke dynamics is a class of behavioural biometrics that captures the typing style of a user. Typing style includes such factors as the length of time it takes to type the login id/password, how long we depress a key and how long we take to type successive keys. By collecting all possible digraphs (two-letter combinations) from the login Id/password – one can develop a model of how the person types these credentials for example. In addition to this static information, one can investigate how a person's typing style evolves with continued practice. This practice effect – or learning curve – can be quantified and used as a metric directly. In addition, any attributes collected for the authentication process must be updated over time. In addition to the static direct attributes mentioned above – secondary or derived attributes should be acquired. These include typing speed, edit distance and entropy to name a few. These attributes provide at the very least an additional range of attributes that can be used in the classification process. In addition, they may provide useful classification information not found in primary attributes [4].

In the biometrics literature – there are two primary objective metrics used to quantify the efficacy of the authentication process: False Rejection Rate (FRR) and False Acceptance Rate (FAR). The former is usually reported as a measure of false rejection – a type I error and the later a false acceptance or type II error. Another measure – called the Cross-over Error Rate (CER) – sometimes referred to as the Equal Error Rate (EER) is also reported – they provide a measure of how sensitive the biometric is at balancing ease of use for the authentic user while at the same time reducing the imposter access rate. All extant biometric systems yield a trade-off between these two measures – those that reject imposters effectively (low FAR) are usually accompanied by a high FRR and vice versa [2].

Keystroke verification techniques can be classified as either static or continuous. Static verification approaches analyse keystroke verification characteristics only at specific times, for example, during the login sequence. Static approaches provide more robust user verification than simple passwords, but do not provide continuous security — they cannot detect a substitution of the user after the initial verification. Continuous verification, on the contrary, monitors the user's typing behaviour throughout the course of the interaction. Keystroke dynamics can be described by several features which are extracted from the typing rhythm of the user. These features are extracted from data which are recorded by the event recording module [3].
According to [1], Keystroke solutions are usually measured in three ways:
Dwell time – how long a key is pressed, Flight time – how long it takes to move from one key to another, and key code. Keystroke dynamics is one of the novel and creative biometric techniques. Usually, each keystroke is represented by two timestamps: the moment that the key was pressed and the moment that it was released. Dwell time refers to a single keystroke and it is defined as the time that passed between the moment the key was pressed and the moment that it was released.

2. Related Work

In 1994, Obaidat and D.T Maccahairolo [18] achieved 97.5% correct classification by using a combination of multilayer feedforward with the BP algorithm (MFN/BP) and sum of product (SOP) network with keystroke time interval. In 1999, Monrose et al [20] proved that Structured Text is more appropriate than free text by working on asset of 63 users and using Euclidean Measure, non weighted Probability, weighted Probability Measure as metrics. They argued that Keystroke patterns depend not only on the user but also on the environment. They were able to achieve a combined FAR and FRR of 7.9 through their experiment. John A. Robinson et al[22] in 1998 used the Minimum Intra Class Distance Classifier, Non Linear Classifier and Inductive Learning Classifier as a tool and realized a FAR : 1) 23 2) 31 3)10 FRR : 1)24 2) 31 3) 62. They worked on 137 users and claimed to have got the Best performance by using inductive learning classifier. The main disadvantage found out was that of the Typographical errors. Sajjad Haider et al [19] discussed a variety of techniques for user authentication. Some of the techniques compared are NN, FuzzyLogic, Statistical Methods and also a hybrid combination of these techniques. FAR and FRR errors are calculated both for single try and two tries. The FAR for NN is the highest at 0.20 and the FRR is highest at 0.22 again for the NN. Also the FAR decreased after two tries but the number of tries does not have much impact on FRR. Kenneth Revett et al [6] used approximately 100 users in an experiment. Decision rules were generated from the rough sets and 97.5 of classification accuracy was achieved. Also digraph time was proved to be an efficient metric to identify the user. Statistical models and diagraph latencies were found to be the first techniques used to analyze keystroke biometrics. Then the neural network (NN) approach was developed by Brown and Rogers [15], they used a simple MLP with BP. Their work was extended by D.T. Lin [24], who considered the deviation on the architecture and parameters of the neural network with customized keystroke latency and gave a 1.1% FAR and 0% impostor pass rate (IPR). N. Capuano [12] used the MLP with RBF as a transfer function, rather than a sigmoid one used previously by others. It resulted in 97% correct authentication with 0% intrusions.

3. Neural Network

An Artificial Neural Network (ANN) is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. The key element of this paradigm is the novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements (neurons) working in unison to solve specific problems. ANNs, like people, learn by example. An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurons. This is true of ANNs as well. In this section, the explaination about the two neural networks : Multilayer perceptron (MLP) and classifier is mentioned.

3.1. Multilayer Perceptron (MLP) network

A multilayer perceptron (MLP) is a feed forward artificial neural network model that maps sets of input data onto a set of appropriate outputs. Figure 1 shows the structure of the MLP network used in this paper. A MLP consists of multiple layers of nodes in a directed graph, with each layer fully connected to the next one. MLP utilizes a supervised learning technique called back propagation for training the network. MLP is a modification of the standard linear perceptron and can distinguish data that are not linearly separable. It consists of three main parts: an input layer, one or more hidden layers, and an output layer. The input layer distributes the input data to the processing elements in the next layer. The second stage is the hidden layer which incorporates the nonlinearity behaviour and the last stage shows the output layer. Input and output are directly accessible, while the hidden layers are not. Each layer consists of several neurons. The goal of this type of network is to create a model that correctly maps the input to the output using historical data so that the model can then be used to produce the output when the desired output is unknown.

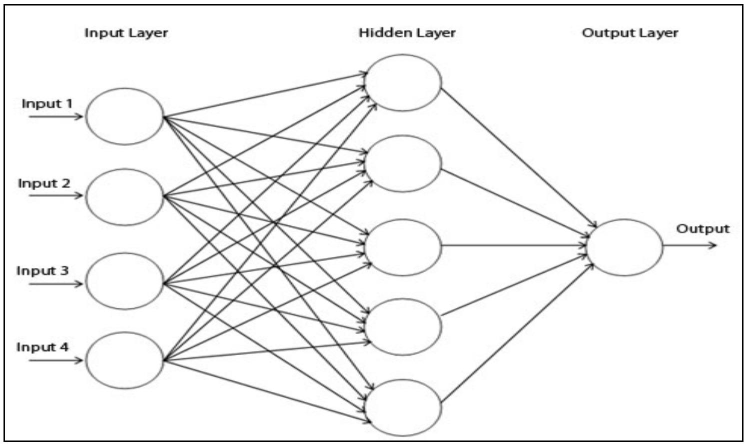

Figure 1: Architecture of an MLP Neural Network

Learning occurs in the perceptron by changing connection weights after each piece of data is processed, based on the amount of error in the output compared to the expected result. This is an example of supervised learning, and is carried out through back propagation, a generalization of the least mean squares algorithm in the linear perceptron.

We represent the error in output node j in the n^{th} data point by

$$e_j(n) = d_j(n) - y_j(n), \qquad (1)$$

Where d is the target value and y is the value produced by the perceptron. We then make corrections to the weights of the nodes based on those corrections which minimize the error in the entire output, given by

$$\mathcal{E}(n) = \frac{1}{2}\sum_j e_j^2(n) \qquad (2)$$

.

Using gradient descent, we find our change in each weight to be

$$\Delta w_{ji}(n) = -\eta \frac{\partial \mathcal{E}(n)}{\partial v_j(n)} y_i(n) \qquad (3)$$

Where y_i is the output of the previous neuron and η is the learning rate, which is carefully selected to ensure that the weights converge to a response fast enough, without producing oscillations. In programming applications, this parameter typically ranges from 0.2 to 0.8. The derivative to be calculated depends on the induced local field v_j , which itself varies. It is easy to prove that for an output node this derivative can be simplified to

$$-\frac{\partial \mathcal{E}(n)}{\partial v_j(n)} = e_j(n)\phi'(v_j(n)) \qquad (4)$$

Where ϕ' is the derivative of the activation function described above, which itself does not vary. The analysis is more difficult for the change in weights to a hidden node, but it can be shown that the relevant derivative is

$$-\frac{\partial \mathcal{E}(n)}{\partial v_j(n)} = \phi'(v_j(n))\sum_k -\frac{\partial \mathcal{E}(n)}{\partial v_k(n)} w_{kj}(n) \qquad (5)$$

This depends on the change in weights of the kth nodes, which represent the output layer. So to change the hidden layer weights, we must first change the output layer weights according to the derivative of the activation function, and so this algorithm represents a back propagation of the activation function.

3.2. Classifier

In the field of machine learning, the goal of statistical classification is to use an object's characteristics to identify which class (or group) it belongs to. A classifier is a system that performs a mapping from a feature space X to a set of labels Y. Basically what a classifier does is assign a pre-defined class label to a sample. For example, if you are building a spam classifier then the feature space contains a representation of an email and the label is either "Spam" or "Non-Spam". A linear classifier achieves this by making a classification decision based on the value of a linear combination of the characteristics. An object's characteristics are also known as feature values and are typically presented to the machine in a vector called a feature vector. A hierarchical classifier is a classifier that maps input data into defined subsumptive output categories. The classification occurs first on a low-level with highly specific pieces of input data. The classifications of the individual pieces of data are then combined systematically and classified on a higher level iteratively until one output is produced. This final output is the overall classification of the data. Depending on application-specific details, this output can be one of a set of pre-defined outputs, one of a set of on-line learned outputs, or even a new novel classification that hasn't been seen before. Generally, such systems rely on relatively simple individual units of the hierarchy that have only one universal function to do the classification. In a sense, these machines rely on the power of the hierarchical structure itself instead of the computational abilities of the individual components. This makes them relatively simple, easily expandable, and very powerful.

Performance Measures: Two important error rates are used to determine the performance of a biometric authentication system – False Acceptance Rate (FAR) and False Rejection Rate (FRR).
FAR is the percentage of impostors inaccurately allowed as genuine users.
It is defined as

$$FAR = \frac{\text{Number of false matches}}{\text{Total number of impostor match attempts}} \quad (10)$$

FRR is the number of genuine users rejected from using the system.
It is defined as

$$FRR = \frac{\text{Number of false rejections}}{\text{Total number of genuine match attempts}} \quad (11)$$

Some researchers report the equal error rate (EER) instead of FAR and FRR. EER is defined as the value of FAR/FRR at an operating point on ROC where FAR equals FRR. Higher FAR is generally preferred in systems where security is not of prime importance, whereas higher FRR is preferred in high security applications. The lower the value of EER, the better the system is [1].

4. Proposed Method

Keystroke solutions are usually measured in three ways: dwell time – how long a key is pressed, flight time – how long it takes to move from one key to another, and key code [1].

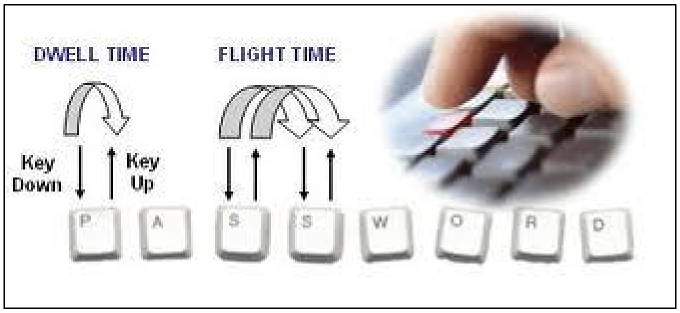

Figure 2: Time Measurement for a Keystroke

In this method we proposed a technique in which, as soon as the string is entered the flight time, dwell time and total time are calculated and then final authentication is done depending upon the required credentials and key time interval values. In this case we calculated the timing parameters for each and every character (i.e. Key) separately.
For example: If the entered string is “JUPITER”
Then timing parameters of each character are calculated
i.e. J, U, P, I, T, E, R respectively.
The approach can be better understood by the following two figures:

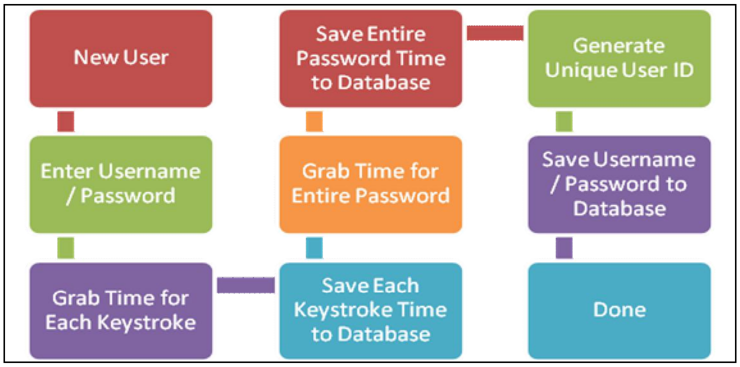
New User
Enter Username / Password
Grab Time for Each Keystroke
Save Each Keystroke Time to Database
Grab Time for Entire Password
Save Entire Password Time to Database
Generate Unique User ID
Save Username / Password to Database
Done

Figure 3: Process for User Registration

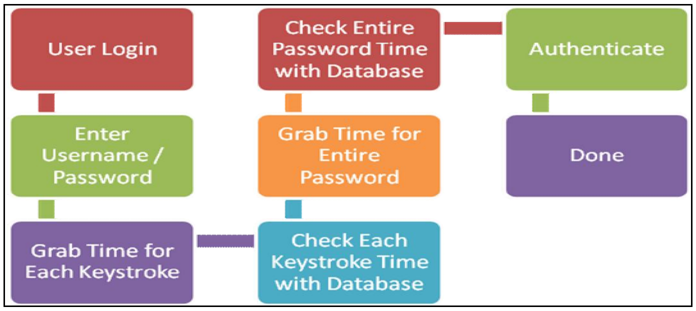
User Login
Check Entire Password Time with Database
Authenticate
Enter Username / Password
Grab Time for Entire Password
Done
Grab Time for Each Keystroke
Check Each Keystroke Time with Database

Figure 4: Process for User Authentication

This approach uses supervised training method in which the learning system is exposed to the environment, which is represented by a measurement vector of features. The measurement vector is also presented to a teacher who determines the desired response. The desired response is then used to create an error signal that adapts the weights of the learning system. Thus, each input-feature vector has an associated desired-output vector, which is then used to train the neural network.

There are several factors which are needed to be considered as follows:

- **Amount of Data**
  In general more data are better than fewer data. A larger number of samples(i.e., a bigger data set) will generally give the network a better representation of the desired problem and will increase the likelihood that the neural network will produce the desired outputs.
- **Features**
  An individual sample is described by a unique set of measurements. In pattern recognition vocabulary these measurements are known as features. Each feature forms a dimension in a space known as feature space. For example, if we wanted to classify different evergreen trees in forest, we would measure different characteristics of each tree. We could measure the length of the needle and length of the cone .This would represent two features. So, for every sample we would measure two quantities and this would form a two-dimensional feature vector to help identify the type of tree. These vectors are then represented by points in feature space.
- **Data Labeling**
  For supervised approaches, the data must be labeled or truthed. This requires the neural network designer or model, to assign target values to each sample collected. If the label is not already a number, then it must be converted to a numerical form in order for the neural network to be trained via computer.
- **Classifier Coding**
  For classifiers, the outputs are generally coded with a 1 for existence in that class and 0 or -1 for absence from that class. With sigmoidal output neurons, sometimes the target output values are pushed back from the extreme edges of the sigmoid so that 0.9 and 0.1 for a logistic function or 0.9 and -0.9 for a hyperbolic tangent function are used instead.
  The output need to be thresholded so that a value above the threshold indicates that a given input is classified in that class and a value below threshold indicates that input is not a member of that class. This thresholding is accomplished by using a step function. Sometimes, it is useful to have an upper and lower threshold for a given classifier design, permitting the classifier to have a “not sure” or indeterminate region. If an output falls above the upper threshold, it is marked as part of class. If it falls below threshold, it is marked as not part of the class. If it falls between the two thresholds, then the class should be considered indeterminate. These results in two binary outputs: one indicating class membership and one indicating no class membership.

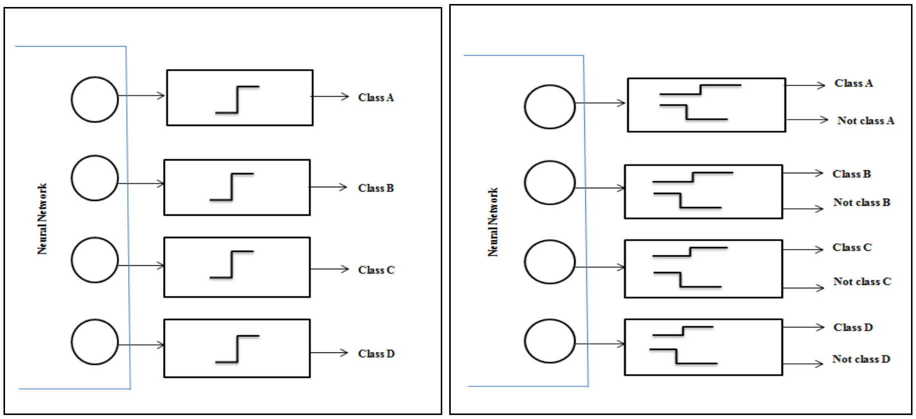

Figure 5: Process of Thresholding Neural Network Outputs to Determine Class Membership

- Pre-processing module
- A degree of inconsistency exists in users typing rhythms. While some individuals may be highly consistent, others are not. This causes problems during classification due to large spurious outliers. In order to reduce these problems, the data is pre-processed before the template is created [14]. To achieve this, the z-score values for each sample in the training set with respect to its class were calculated.

$$Z_i = \frac{x_i - \mu(x)}{\sigma(x)}$$

Where μ(x) and σ(x) are the mean and standard deviations of the feature and x_i is the ith sample of the feature. Values that fall outside the neighbourhood of z are eliminated and replaced with the mean of the rest of the feature samples.
In case of positive amendments, an update of the stored template with the verification template (user login template), will be appropriate. There will be an adaptation mechanism for creating an updated template, including a new sample and discarding the oldest one.

5. Results and Discussion

In the experiments conducted there were three situations of authentication.

- Legitimate user authentication: the users tried to be authenticated in their own account.
- Impostor user authentication: the users tried to be authenticated in other user's accounts, knowing the string typed by their owners.
- Observer impostor user authentication: the users observed how the other users type their strings, then they tried to be authenticated in their accounts.
- User's samples were collected in different periods of time, the samples like the start time, end time and the difference in time (the moment at which the first key was released and second was pressed) for each keystroke as well as the total time required to enter a password.
- The performance of biometrics systems are generally measured by two kinds of error rates [24].
- False Acceptance Rate (FAR): the probability that the system will fail to reject an impostor user.
- False Rejection Rate (FRR): the probability that the system will fail to verify the legitimate user claimed identity.
- Other performance measures based on these rates are [21] as follows.
- Zero FAR: FRR when the FAR is equal to zero.
- Zero FRR: FAR when the FRR is equal to zero.
- Equal Error Rate (EER): the value when the FAR and FRR are equally likely.
- The operating threshold employed by a system depends on the nature of the application and it is very difficult to find a system that operates in one of these three points [21]. In practical applications, the system is configured to operate around or between these points. While registering a user the keystroke time for each key and for entire password is saved to the database, and is used while authenticating the user with a threshold limit as specified.

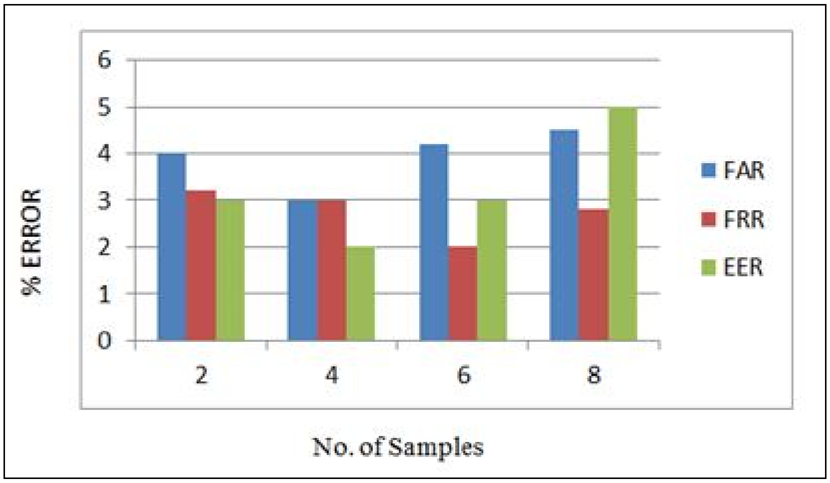

Figure 6: Classification On The Basis Of FAR, FRR and EER

Method	FAR	FRR	EER	Accuracy
Harun et al.[1]	-	-	22.9%	-
N. Capuano[12]	-	-	-	97%
M.S. Obaidat and D.T Macchairolo[18]	-	-	-	97.5%
Haider et al.[19]	6%	2%	-	-
Robinson et al.[22]	9%	10%	-	-
Monrose et al [20]	20%	20%	-	-
Lin[24]	1.11%	0%	-	-
Our Research	4.8%	3.1%	5%	-

Table 1: Comparison of Our Method with Other Methods

6. Conclusion

This paper presents a methodology through typing biometrics features that improves the usual login-password authentication. Some experiments were conducted and the best performance was achieved using a statistical classifier based on distance and the combination of four features (key code, DD, UD, and DU times), obtaining a 1.35% FRR and a 1.80% FAR. The FAR and FRR values were calculated for each and every key. The MLP was considered for enrollment purpose and for maintaining and updating the database of several user entries. Comparing with other approaches Like RBFN, it was observed that MLP is better for obtaining accurate results. The use of four features to authenticate users is novel, since prior studies used just one or two features. Distance classifier gives the equalized EER values of 4% to 10%.This paper shows the influence of some practical aspects, which were tested and observed and shows that they have a relevant influence in the performance results. These aspects are: the familiarity of the target string, the two-trial authentication, the adaptation mechanism, the timing accuracy, and the number of samples in enrollment.

7. References

1. Harun, N.; Woo, W.L. ; Dlay, S.S. ;" Performance of keystroke biometrics Authentication system using Artificial Neural Network (ANN) and Distance Classifier Method " International Conference on Computer and Communication Engineering (ICCCE), 2010
2. Schclar, A.; Rokach, L.; Abramson, A.; Elovici Y,"User Authentication Based on Representative Users" IEEE Transactions on systems, man and cybernetics part C, Vol 42, NO. 6, pp.1669-1678,(NOVEMBER 2012)
3. BioPassword: History and Science of keystroke dynamics. http://www.biopassword.com/resources/
4. Bhatt, S.; Oxford Coll. of Sci., Bangalore, India ; Santhanam, T." Keystroke Dynamics for biometric authentication-A survey", International Conference on Pattern Recognition, Informatics and Medical Engineering (PRIME), 21-22 Feb. 2013
5. D.Shanmugapriya and G.Padmavathi, "An Efficient Feature Selection Technique for User Authentication using Keystroke Dynamics," IJCSNS, vol. 11, no.10, 2011.
6. Kenneth Revett , A machine learning approach to keystroke dynamics based user authentication , Int. J. Electronic Security and Digital Forensics, Vol. 1, No. 1, 2007
7. Pin Shen Teh, Andrew Beng Jin Teoh, Thian Song Ong, Han Foon Neo,"Statistical Fusion Approach on Keystroke Dynamics". Third International IEEE Conference on Signal-Image Technologies and Internet-Based System (SITIS '07), pp. 918-923, (2007).
8. L. K. Maisuria , C. S. Ong and W. K. Lai, " A comparison of artificialneural network and cluster analysis for typing biometricsauthentication", International Joint Conference on Neural Network, IJCNN'99, vol.5, pp 3295-3299, (1999).
9. Hasimah Ali, Wahyudi and Momoh J.E Salami, "Intelligent Keystroke Pressure-Based Typing Biometrics Authentication System by Combining ANN and ANFIS-Based Classifiers", International Colloquium on Signal Processing & Its Applications (CSPA), pp198,(2009).
10. S. Modi and S. J. Elliott, "Keystroke dynamics verification using a spontaneously generated password password," in Proc. 40th Annu. Int. Carnahan Conf. Security Technol., Lexington, KY, Oct. 2006, pp. 116–121.
11. A. Peacock, X. Ke, and M. Wilkerson, "Typing patterns: A key to user identification," IEEE Security Privacy, vol. 2, no. 5, pp. 40–47, Sep./Oct. 2004.
12. N. Capuano, M.Marsella, S.Miranda and S. Salerno, "User Authentication with Neural networks", Univerity of Salerno Italy. http://www.capuano.biz/Papers/EANN_99.pdf
13. R. O. Duda, P. E. Hart, and D. G. Stork. Pattern Classification,2nd edition, John Wiley & Sons, Inc.,2001
14. S R Shorrock, D J Atkinson, S S Dlay, Biometric verification of computer users with probabilistic and cascade forward neural networks.
15. M. Brown and S.J Rogers, "User identification via keystrokecharacteristics of type names using neural networks." International journal of Man Machine studies, vol 39, pp 999-1014,(1993).
16. Ross, A., Shah, J., & Jain, A. K. March 2005. Towards reconstructing fingerprints from minutiae points. Proceedings of SPIE Conference on Biometric Technology for Human Identification II, 5779, 68–80.

17. Hocquet, S., Ramel, J.-Y., & Cardot, H. 2005. Fusion of methods for keystroke dynamic authentication. Autoid, 0, 224–229.
18. M.S. Obaidat and D.T Macchairolo, "A multilayer neural network for computer access security", IEEE transactions on Systems, Machine and Cybernetics, vol 24(5), (1994).
19. S. Haidar, A. Abbas, and A. K. Zaidi, "A multi-technique approach for user identification through keystroke dynamics," in Proc. IEEE Int. Conf.Systems, Man, and Cybernetics, vol. 2, 2000, pp. 1336–1341.
20. F. Monrose andA. D. Rubin, "Keystroke dynamics as a biometric for authentication," Future Gen. Comput. Syst., vol. 16, no. 4, pp. 351–359, 2000.
21. S.Bleha and M. obaidat. Computer user's verification using the perceptron algorithm. IEEE Transactions on Systems, Man and Cybernetics, 23(3):900 -902, May. 1993
22. J. A. Robinson, V. M. Liang, J. A. Michael, andC. L. MacKenzie, "Computer user verification login string keystroke dynamics," IEEE Trans. Syst., Man, Cybern., vol. 28, no. 2, pp. 236–241, Mar.–Apr. 1998.
23. D. Polemi. (1997) Biometric Techniques: Review andEv aluation of Biometric Techniques for Identification and Authentication, Including an Appraisal of the Areas Where They are Most Applicable. Institute of Communication andComputer Systems, National Technical University of Athens, Athens, Greece. [Online]. Available: ftp://ftp.cordis.lu/pub/infosec/docs/biomet.doc, EU Commission Final Rep.
24. D. T. Lin, "Computer-access authentication with neural network based keystroke identity verification," in Proc. Int. Conf. Neural Networks, vol.1, 1997, pp. 174–178.
25. Lívia C. F. Araújo, Luiz H. R. Sucupira Jr., Miguel G. Lizárraga, Lee L.Ling, andJoão B. T. Yabu-Uti "User Authentication Through Typing Biometrics Features," IEEE Transactions on signal processing, Vol. 53, no. 2, February 2005
26. A. K. Jain, R. Bolle, andS. Pankanti, Biometrics: Personal Identification in Networked Society. Norwell, MA: Kluwer, 1999