A Preliminary Investigation of User Authentication Using Continuous Keystroke Analysis

P. S. DOWLAND[1], H. SINGH[2], S. M. FURNELL[3]

[1]pdowland@plymouth.ac.uk
[2]hsingh@jack.see.plym.ac.uk
[3]sfurnell@ plymouth.ac.uk
Network Research Group
Department of Communication and Electronic Engineering
University of Plymouth
Drake Circus
PLYMOUTH
PL4 8AA
United Kingdom
Tel: +44 1752-233521 Fax: +44 1752-233520

Key words: Keystroke Analysis, Authentication, Biometrics, Data Mining.

Abstract: There has been significant research in to the provision of reliable initial-login user authentication, however there is still a need for continuous authentication during a user session. This paper presents a series of results from the preliminary statistical analysis of multi-application keystroke data. This has been contrasted with a Data Mining approach to the production of a unique user profile. This paper aims to determine which approach provides the best basis for further research. It is determined that the technique offers promise as a discriminator between individuals in an operational context, but further investigation with larger data sets is required with a combination of approaches being considered in order to improve the accuracy.

1. INTRODUCTION

There have been a number of previous studies that have considered the security weaknesses in modern IT system and, whilst various recommendations and technical solutions have been proposed, many still rely on enhancing the initial login-stage mechanism (e.g. via biometric identification, smart cards etc.) [COPE 90, SHER 92, MILL 94]. Whilst this improves the initial authentication judgement, there is still a need for user authentication throughout a session. In most systems there is no further check on a users' identity beyond the initial username/password. Once a user gains legitimate access to IT resources, it is feasible for there to be no further challenge, with the only possibility for detection of a masquerader being the post-event detection of a major incident (i.e. an impostor can masquerade as the valid user without detection or challenge).

To counter this risk, it is suggested that some form of user monitoring is desirable to continuously (or periodically) authenticate the user in a transparent manner. Whilst such monitoring is technically feasible, there are significant issues to be considered in selecting appropriate attributes to assess. This is particularly important, as continuous monitoring must be transparent to the end user in order to minimise any perceived inconvenience (with the exception of appropriate challenges in the event of a significant profile deviation).

This paper specifically considers the problems of continuous user authentication using keystroke digraph latencies. This area has not received much attention and as such, most of the background research is based upon static keystroke analysis [JOBU 89, BROW 93, JOYC 90] (i.e. where the users' typing was constrained). Keystroke analysis is, however, considered by end users as the most acceptable form of continuous authentication [FURN 00]. A GUI environment produces a new challenge, as there is no option to control the users' typing. This can cause problems, as it is difficult to determine in which application individual digraph pairs were entered. This paper will introduce a statistical approach for detecting deviation from a user's historical keystroke profile captured under a multi-tasking windowed environment. Following this initial analysis, a Data Mining (DM) approach will be considered in order to determine the potential for improving user classification. It should be noted that the aim of this paper is to determine which approach provides the best basis for further research and is not intended as a thorough analysis of keystroke latencies for user authentication. Finally, some thoughts on future work are introduced which will be developed further.

2. EXPERIMENT OVERVIEW

Although there have been a series of papers describing the mechanisms for keystroke analysis, the authors have been unable to identify any research specifically focussed on continuous keystroke analysis in which the collection of users typing samples was not artificially constrained in some way through a custom interface (e.g. asking the user to type known strings).

The experiment was designed to allow keystroke data to be collected under the Microsoft Windows NT environment. In order to collect the required data, it was necessary to implement a mechanism for acquiring keystroke notifications across all applications running within a users' active session. As the client systems were running Microsoft Windows NT v4.0, it was necessary to implement a system-wide hook function that would receive keyboard events through the Windows message chain. System-wide hooks allow a specified code block (hook-function) to receive the appropriate Windows messages (e.g. WM_KEYUP for the key-up event) irrespective of the target application (i.e. it is possible for a hook function residing in a system DLL to receive keystroke notifications for all currently running applications). This effectively allowed application keystroke data to be duplicated and directed towards the data logger on the client workstation. Technical details of the implementation of the hook function and its associated support files are beyond the scope of this paper and, as such, have been omitted. There are a number of resources available that provide further information for interested readers [DOWL 00, MICR 00]. In order to determine accurate digraph latencies, it was also necessary to implement a high-accuracy timer (as the default timers available do not offer adequate accuracy for the millisecond latencies expected).

To eliminate extreme short/long digraph latencies that may adversely affect the distribution of digraph times, any digraph pair whose latency fell outside a nominal range was excluded from the archived data. For the purposed of this experiment the range was restricted to times above 40ms and below 750ms. These thresholds are based on the original experiments carried out by the authors [FURN 95] and are designed to eliminate samples where two keys may have been accidentally struck together (thus, producing an infeasibly small latency) or, where the user may have made a pause in their typing and thus introduced an unnaturally large inter-keystroke latency. The output of this pre-processing was a data file containing the following structure:

first_char *second_char* *digraph_latency*

For this experiment a total of ten users were profiled. As the intention was to evaluate the analysis mechanisms without implementing a large-scale trial, tests were carried out using a small set of test subjects. The main limiting factor was the need to collect data over a prolonged period (weeks rather than hours). Despite the small scale of the trial, it still proved difficult to collect sufficient data in order to provide a valid comparison between users. Due to this limited set of data, analysis has focussed on the 4 main users who provided the largest profiled data sets in order to best illustrate the trends observed.

3. STATISTICAL ANALYSIS

Following the pre-processing described in the previous section, the experimental data for each user was then processed off-line to calculate the mean and standard deviation values for each unique digraph pair. In the event that any digraph pair had a standard deviation greater than its mean value, the digraph samples were sorted and the top/bottom 10% were then removed with subsequent re-calculation of the mean and standard deviation values – this was only attempted where at least ten samples were available for the digraph pair. The reason for this additional step was to remove digraph samples where the latencies would have an adverse affect on the standard deviation (i.e. the distribution of samples was tightened).

Once a set of digraph pairs was produced (with corresponding mean/standard deviation digraph latency values), the user's profile was further constrained by filtering out digraph pairs where the sample count fell below a nominal threshold value. Our experiments fixed this value at fifty samples; however, the software used for analysis allows a variable threshold that will be investigated further in the future work described in a later section. A summary of the profiles generated by this method is shown in *Table 1*.

Table 1: Summary of user profile statistics

User	Unique Digraph Pairs	Filtered Digraph Pairs	Average Typing Speed
User A	466	122	151ms
User B	405	51	145ms
User C	412	89	206ms
User D	461	127	162ms

Once a user profile was generated, the profile was evaluated by comparison with the users' raw keystroke data. This allowed the test profile to be evaluated using the users' own data (to test the False Rejection Rate – FRR) and against other users' keystroke data (to test the False Acceptance Rate – FAR).

As there is likely to be significant variation in a users' own session data, a compensatory factor was applied to the standard deviation that could be varied in a "live" environment according to the security needs of the organisation. This factor allowed the number of standard deviations from the mean to be adjusted. For the purposes of this experiment, four weightings were considered, namely 0.5, 1, 1.5 and 2. This produced an acceptable digraph range:

digraph range = mean ± (standard deviation * weighting factor)

When viewing the preliminary results (*Figure 1*), if we consider the four users A, B, C and D and follow the vertical columns of data, we can see a clear peak for each users data when compared with their own profile. This is most noticeable for user C where a significant peak is observed (50% of all digraphs accepted) compared with 35% when user B's digraph data was tested against the same profile.

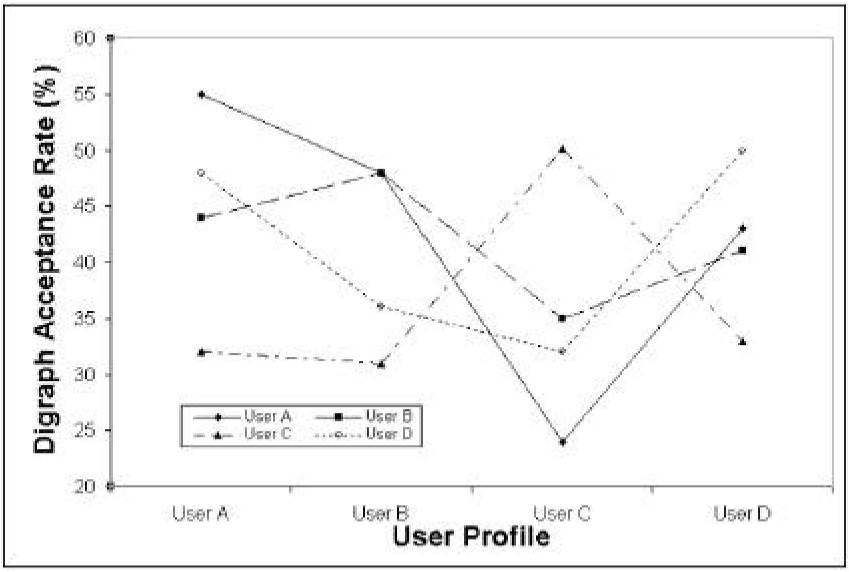

Figure 1: User profile comparisons

Although there was a clear correlation between user C's profile and data, if we consider user A, there was a high FAR for data from users D and B (impostors) when compared with user A's profile. We can also see that in user B's profile the impostor "user A" achieved the same acceptance rate (48%). It is clear from these results that an additional measure of acceptance/rejection is required. To further test the FAR/FRR of the test system, the analysis software monitored the number of consecutively rejected digraph pairs – representing the highest alert level of the system (*Figure 2*).

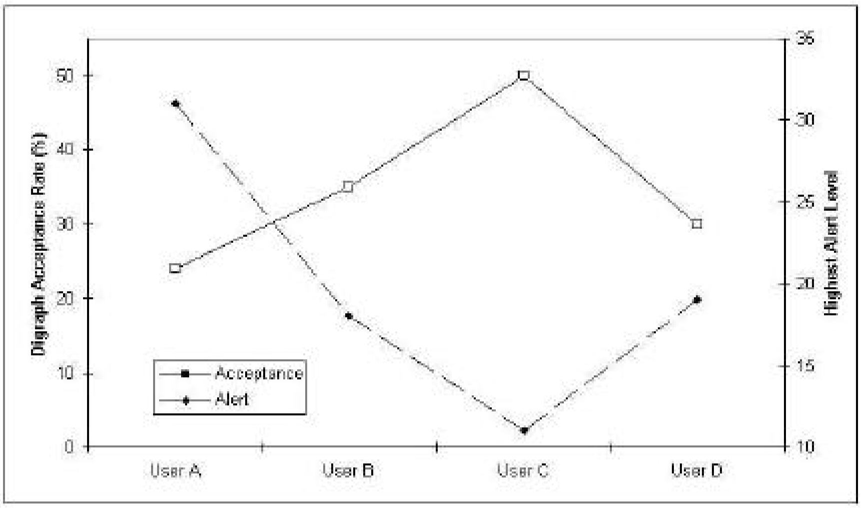

Figure 2: Single user profile comparison

When considering (*Figure 2*) we can identify two distinct trends. Firstly, the top line plots the digraph acceptance rate for all user data sets against user C's profile. Here we can see a clear peak correlating to user C's own data and corresponding reductions in the acceptance rates for the other users' data. Secondly, the lower line indicates the highest alert level detected by the analysis software. This is simply a record of the highest count of consecutively rejected digraph times (excluding non-profiled digraph pairs). Again, we can see a correlation between user C's own data when compared with their profile and corresponding increases in the alert level as impostor data sets are compared with the target profile.

4. DATA MINING ANALYSIS

The methodology described in the previous sections, using traditional statistical approaches, requires a significant level of manual intervention in the data analysis stages. Further, it is time consuming when considering the amount of data generated from a single session or multiple sessions and the number of users on a system. From this we can determine there is a need to automate some of the data analysis pre-processing stages. These stages offer the opportunity to investigate Data Mining (DM) methodology and algorithms, a previously untried approach in this field, in order to eliminate the manual approaches adopted and also to compare the FAR/FRR percentage accuracy with the statistical approach. Data Mining can be described as a collection of techniques and methodologies used to explore vast amounts of data in order to find potentially useful, ultimately understandable patterns [FAYY 96] and to discover relationships. The methodology used to analyse the raw keystroke data is derived from the four main activities of DM; selection, pre-processing, data mining and interpretation [FAYY 96]. DM is an iterative and interactive process, involving numerous steps with many decisions being made by the user. Different algorithms are optimised based on the predefined DM task. This involves deciding whether the goals of the DM process are classification, association, or sequential [MICH 94].

For the purpose of this work, the data sets were split into a ratio of 9:1 hence into two parts; a training set and a testing set, which is a commonly used technique known as train and test. The Intelligent Data Analysis (IDA) Data Mining Tool [SING 99] is used to analyse the sample data sets which incorporates algorithms from the fields of Statistical, Machine Learning and Neural Networks. Six algorithms, k-NN, COG, C4.5, CN2, OC1 and RBF were chosen for this investigative work. The algorithm or classifier is subjected initially with the training set and then the classification accuracy is tested using the unseen data set or testing set. The results give an indication of the error rate (or FAR) and the overall classification accuracy of the trained algorithms.

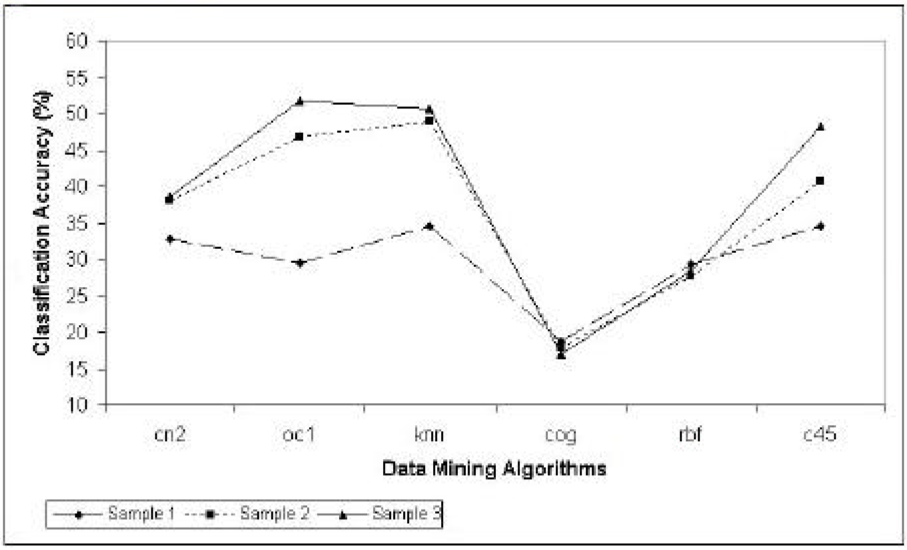

Figure 3: Varying sample sizes with fixed number of classes and attributes

The percentage classification accuracy obtained is encouraging as depicted in (*Figure 3*), which shows that when the sample size is increased, the classification accuracy obtained increases proportionally, except for the

COG a statistical based algorithm and RBF a Neural Network based algorithm. This is important when considering the size of data being analysed and hence eliminates the ad-hoc approaches adopted using traditional statistical methods.

The initial results suggest that Machine Learning (OC1 and C4.5) and Statistical (k-NN) based algorithms are suitable for these types of data sets. Despite the results, more work needs to be carried out in order to correlate the results to a specific or group of algorithm(s), in order to obtain a higher percentage of classification accuracy.

5. CONCLUSIONS

It is clear from the results presented in this paper that there is some potential for continuous user authentication based on keystroke analysis. However, it is also clear that a simple statistical approach does not provide sufficient distinction between users. The DM approach is limited due to the nature of the data gathered and will also require further research. It is proposed that further work will investigate the usefulness of trigraph keystroke combinations (timings for three consecutive keystrokes) and the possible use of word-graph timings (timings for frequently occurring words). Further analysis will be carried out on much larger data sets in order to give a higher statistical reliability and will also incorporate high-level characteristics (average typing speed and typing error rates) which will provide additional information to the system-characteristic based DM approach being developed in parallel with this research [SING 01]. Other approaches that will be investigated include, consideration of various standard deviation weightings, varying the minimum number of samples for profiled digraphs and varying the inclusion threshold for each sampled digraph. A further possibility for research may be an investigation into a correlation between digraph latencies and the applications in which they were generated (i.e. application specific keystroke profiles).

This paper has presented a series of results from the preliminary statistical analysis of multi-application keystroke data. This has been contrasted with a DM approach to the production of a unique user profile. Whilst the results from this stage of the research are not as encouraging as we had hoped for, they have shown a potential for the use of continuous user authentication. The next phase will concentrate on a combination of techniques to improve the digraph acceptance rate seen in these results.

6. REFERENCES

[COPE 90] Cope J.B.; "Biometric systems of access control"; Electrotechnology; pp71-74; April/May 1990.

[BROW 93] Brown M. & Rogers S.J.; "User identification via keystroke characteristics of typed names using neural networks"; International Journal of Man-Machine Studies; pp999-1014; 1993.

[DOWL 00] Dowland P.S. & Furnell S.M.; "Enhancing Operating System Authentication Techniques"; Proceedings of the International Network Conference 2000 (INC2000); pp253-261; July 2000.

[FAYY 96] Fayyad U.M.; "Data Mining and Knowledge Discovery: making sense out of data"; IEEE Expert; vol. 11; no. 6; pp20-25; 1996.

[FURN 95] Furnell S.M.; "Data security in European healthcare information systems"; PhD Thesis; University of Plymouth, UK; 1995.

[FURN 00] Furnell S.M., Dowland P.S., Illingworth H.M. & Reynolds P.L.; "Authentication and Supervision: A survey of user attitudes"; Computers & Security; vol. 19; no. 6; pp519-539; 2000.

[JOBU 89] Jobusch D.L. & Oldehoeft A.E.; "A survey of password mechanisms: Weaknesses and potential improvements. Part 1"; Computers & Security; p587-603; 1989.

[JOYC 90] Joyce R. & Gupta G.; "Identity Authentication Based on Keystroke Latencies"; Communications of the ACM; vol. 33; no. 2; pp168-176.

[MICH 94] Michie D., Spiegelhalter D.J. & Taylor C.C.; "Machine Learning, Neural and Statistical Classification"; Ellis Horwood; ISBN 0-13-106360-X; pp136-141; 1994.

[MICR 00] Microsoft Corporation; "Monitoring System Events"; 2000; http://msdn.microsoft.com/library/psdk/winbase/hooks_9rg3.htm

[MILL 94] Miller B.; "Vital Signs of Identity"; IEEE Spectrum; February; 1994.

[SHER 92] Sherman R.L.; "Biometrics Futures"; Computers and Security; vol. 11; no. 2; pp128-133; 1992.

[SING 99] Singh H., Burn-Thornton K.E. & Bull P.D.; “Classification of Network State Using Data Mining”; Proceedings of the 4th IEEE MICC & ISCE ’99; Malacca, Malaysia; vol. 1; pp183-187; 1999.

[SING 01] Singh H., Furnell S.M., Lines B.L. & Dowland P.S.; “Investigating and Evaluating Behavioural Profiling and Intrusion Detection Using Data Mining”; Proceedings of the International Workshop on Mathematical Methods, Models and Architectures for Computer Networks Security (MMM) 2001; St Petersburg, Russia; 21-23 May 2001.