# A LONG-TERM TRIAL OF KEYSTROKE PROFILING USING DIGRAPH, TRIGRAPH AND KEYWORD LATENCIES

Paul S. Dowland and Steven M. Furnell
*Network Research Group, School of Computing, Communications and Electronics,*
*University of Plymouth, Drake Circus, Plymouth, PL4 8AA, United Kingdom,*
*info@network-research-group.org*

Abstract:    A number of previous studies have investigated the use of keystroke analysis as a means of authenticating users' identities at the point of initial login. By contrast, relatively little research has focused upon the potential of applying the technique for identity verification during the logged-in session. Previous work by the authors has determined that keystroke analysis is a viable metric for continuous monitoring, provided that sufficient data is captured to create a reliable profile. This paper presents a series of results from a three-month trial in which profiles were created using digraph, trigraph and keyword-based keystroke latencies. The profiles were based upon a total of over 5 million keystroke samples, collected from 35 participants. The results demonstrate that the techniques offer significant promise as a means of non-intrusive identity verification during keyboard-related activities, with an optimum false acceptance rate of 4.9% being observed at a rate of 0% false rejection.

Key words:    Authentication, Misuse Detection

## 1.        INTRODUCTION

Over the last twenty years, the concept of keystroke analysis has been the focus of considerable research as a means of user authentication. The potential for profiling of keypresses was first identified by Gaines et al (1980). Since then, a number of research projects have been conducted to evaluate different methods of data gathering (using a range of operating systems and considering a variety of metrics) and post-processing techniques (ranging from purely statistical to AI/neural network approaches).

To date, however, virtually all published studies have focussed upon looking at the application of static strings, such as username and password pairs using the inter-keystroke digraph latency timing method. From the earliest studies in 1980 (Card et al & Gaines et al), the focus has been on the analysis of digraph latencies. Later studies, such as those by Joyce & Gupta (1990) and Mahar et al (1995) further enhanced the work, identifying additional statistical analysis methods that provided more reliable results.

In Legget et al. (1991), the concept of dynamic keystroke analysis was first proposed, with the introduction of a reference profile that could be used to monitor a live user session. Brown and Rogers (1993) also explored the idea of dynamic analysis, presenting preliminary results.

The authors' previous research (Dowland et al., 2002) described an experiment evaluating keystroke analysis based on inter-keystroke digraph latencies under Windows. This earlier trial concentrated upon the capture and subsequent analysis of digraph latencies using inter-keystroke timings. The trial results demonstrated the viability of this method, but suggested that, to be a reliable authentication measure, user profiles would need to be based upon much larger sample sizes. The previous trial was also based on a limited number of users in order to quickly evaluate the viability of the technique.

This paper presents the results of a long-term trial that was aimed at evaluating a range of techniques using a larger number of participants. This trial captured and evaluated trigraph and keyword latencies, in addition to digraph timings, under the Windows operating system. The paper begins by introducing the technical aspects of the trial conducted over a three month period before considering the statistical approach taken with the data analysis stage. The results are presented and discussed, leading to some overall conclusions, and proposals for future work.

## 2.    CAPTURING KEYSTROKE DATA IN WINDOWS

While keystroke analysis has been investigated (and hence implemented) in previous studies, a GUI environment (e.g. Microsoft Windows) introduces new challenges. In previous published studies, the user has been required to type and interact with a specific application (typing either pre-defined or free-form text). While this approach makes the development of the keystroke monitoring software simple, and maintains the consistency of the

test environment, it is not representative of normal typing behaviour as the user becomes focussed upon the task of typing, rather than focussed upon a task that *involves* typing. If the aim is to produce static keystroke analysis for occasional authentication judgements (e.g. supplementing login authentication) then this approach will work well. However, to implement continuous supervision using dynamic keystroke analysis it is necessary to monitor the users' normal behaviour when interacting with their normal applications and operating system environment. Even providing a simulation of these environments may not be sufficient to obtain valid sample data upon which to base a profile.

In order to address this problem, software was developed that would transparently monitor and log all typing activity. The system was designed to allow keystroke data to be collected under the Microsoft Windows XP environment (although the technique is equally applicable in all Windows operating systems). In order to collect the required data, it was necessary to implement a mechanism for acquiring user typing patterns across all applications running within a users' active session. This is important as the experiment was designed to create a profile for each user based upon their typical typing patterns when using their computer (not constrained to a specific application or task). The implementation of the keylogger utilised several key features of the Windows operating system and the underlying chains of messages on which the operating system is built (these are briefly discussed in the following section). The authors have not investigated the applicability of these techniques under other operating systems but it is likely that the same system could be developed under other systems providing access is given to the keypress events at an appropriate level.

Figure 1 illustrates the software architecture used to capture and log keystroke activity under Windows. As keys are pressed, messages are generated by Windows for both key up and down events. These messages are captured through the use of a hook function that redirects messages to a nominated program. The messages are passed from the hook (implemented as a system-wide DLL written in C) to the keylogger (implemented in Visual Basic and deployed as a system tray application). The keylogger functioned completely transparently to the user, requiring no user action to start or stop the logging process. The application was automatically started when the operating system (O/S) booted (run from the Startup program group on the start menu) and shut down automatically when the O/S closed. Gathered data was automatically saved after every 500 digraphs pairs and when the application was closed. To reassure users, an option was included to suspend logging of keystrokes. This was included due to concerns expressed

by some users about monitoring of specific inputs – e.g. the typing of on-line banking login details.
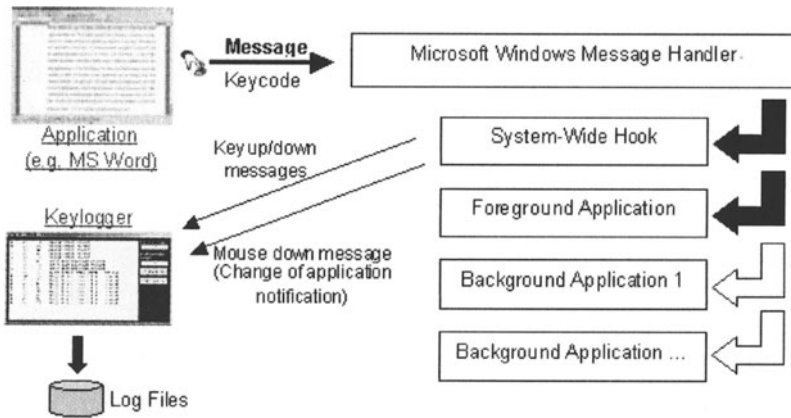


*Figure 1.* Implementation of keylogger

For each digraph pair logged, the application stored five items of information (Table 1) – these being written to an Access database after every 500 digraphs. This process was also repeated for each trigraph and keyword latency (i.e. trigraphs were stored as three consecutive characters and keywords as a string).

*Table 1.* Keylogger attributes logged per digraph

| Item | Data types |
|------|-----------|
| AutoID | Auto-incrementing record number. This is used to maintain the order of the keystrokes typed as the timestamp is only accurate to 1 second. |
| Left character (C1) | ASCII code representing character |
| Right character (C2) | ASCII code representing character |
| Latency | Integer representing inter-keystroke latency in milliseconds |
| Application | String containing the window title from the foreground application. |
| Timestamp | A timestamp is added to every keystroke logged for later use. |

While digraph and trigraph logging were based upon all keystrokes entered, keyword logging was based on a look up list. The top 200 commonly occurring words in the English language were monitored, and as each word was entered, its latency was recorded.

## 3.      EXPERIMENTAL PROCEDURE

For this experiment a total of 35 users were profiled over a period of

three months. Unfortunately several users disabled the keylogger when entering sensitive information and consequently forgot to re-enable it. Despite this, the key-logging trial collected considerable volumes of data with nearly six million samples collected across digraphs, trigraphs and keywords (Table 2).

*Table 2.* User profile results

| User | Mean Digraph Latency (ms) | Typing Skill Classification | Digraphs | Trigraphs | Words |
|------|------|------|------|------|------|
| User 1 | 91 | Best | 34352 | 23352 | 1403 |
| User 2 | 156 | Average (skilled) | 53306 | 36912 | 2599 |
| User 3 | 99 | Best | 156718 | 107107 | 6154 |
| User 4 | 251 | Average (non-skilled) | 27324 | 18688 | 1310 |
| User 5 | 112 | Good | 50822 | 36713 | 1465 |
| User 6 | 154 | Average (skilled) | 50167 | 34484 | 1885 |
| User 7 | 106 | Good | 78579 | 54959 | 4349 |
| User 8 | 130 | Good | 50102 | 35102 | 2932 |
| User 9 | 97 | Best | 37618 | 24755 | 1741 |
| User 10 | 145 | Average (skilled) | 70337 | 48942 | 4643 |
| User 11 | 147 | Average (skilled) | 227660 | 145846 | 10617 |
| User 12 | 102 | Good | 20216 | 14142 | 1032 |
| User 13 | 157 | Average (skilled) | 65312 | 43015 | 1730 |
| User 14 | 141 | Average (skilled) | 33639 | 23090 | 1784 |
| User 15 | 139 | Good | 15951 | 11159 | 1068 |
| User 16 | 150 | Average (skilled) | 42839 | 30299 | 2037 |
| User 17 | 106 | Good | 105543 | 68068 | 3173 |
| User 18 | 177 | Average (skilled) | 89730 | 59292 | 3121 |
| User 19 | 117 | Good | 103876 | 71635 | 4617 |
| User 20 | 121 | Good | 78597 | 53495 | 4479 |
| User 21 | 141 | Average (skilled) | 80626 | 55881 | 2807 |
| User 22 | 110 | Good | 117365 | 79534 | 6557 |
| User 23 | 131 | Good | 118805 | 77013 | 5682 |
| User 24 | 89 | Best | 201260 | 131954 | 8517 |
| User 25 | 203 | Average (skilled) | 38944 | 26655 | 2266 |
| User 26 | 192 | Average (skilled) | 48469 | 33907 | 2555 |
| User 27 | 125 | Good | 33068 | 23115 | 1679 |
| User 28 | 91 | Best | 70217 | 47033 | 2128 |
| User 29 | 104 | Good | 88059 | 55707 | 3815 |
| User 30 | 202 | Average (skilled) | 40741 | 28789 | 1007 |
| User 31 | 86 | Best | 310823 | 211419 | 19726 |
| User 32 | 93 | Best | 353867 | 237274 | 18056 |
| User 33 | 144 | Average (skilled) | 276669 | 183455 | 6057 |
| User 34 | 143 | Average (skilled) | 124409 | 87079 | 953 |
| User 35 | 130 | Good | 140044 | 85413 | 6240 |
| | | **Totals** | **3,436,054** | **2,305,283** | **150,184** |

Before considering the data from each user, the typing skill for each participant was evaluated based on the categorisations proposed by Card et

al. (1980) where typists are broadly categorised into one of six categories. The results are presented in Table 2 together with the quantity of samples for each user (shown separately for digraph, trigraph and keywords). The results are weighted towards typists with above average skills due to the nature of the test subjects (i.e. all subjects were regular computer users who spent prolonged periods typing). This was considered acceptable as the likely use for a fully implemented system would be in environments with semi-skilled users (i.e. relatively few unskilled/poor typists).

## 4.      STATISTICAL ANALYSIS

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 log files. For the purpose of this experiment the range was restricted to times above 10ms and below 750ms. In an earlier trial the range was restricted to 40ms – 750ms, with these thresholds based on previous work conducted by Furnell (1995), and were 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. Unfortunately, the low pass filter was responsible for substantial quantities of data being removed from the user profiles and, as such, was reduced to 10ms for the purposes of this trial. If a digraph was removed due to the filtering, this also reset the trigraph and keyword logging so no further thresholds were needed for these two measures.

Following the initial filtering, the experimental data for each user was processed off-line to calculate the mean and standard deviation values for each unique digraph, trigraph or keyword. In the event that any profiled sample had a standard deviation greater than its mean value, the samples were sorted and the top/bottom 10% was then removed, followed by subsequent re-calculation of the mean and standard deviation values. The reason for this additional step was to remove samples where the latencies would have an adverse affect on the standard deviation (i.e. the distribution of samples was tightened).

Once all the user profiles were calculated, another application (the data comparator) was used to generate tables of results for each of the methods. The data comparator (Figure 2) was based on the original analyser developed in the previous trial (Dowland et al., 2002). A small number of

additional features were introduced to the comparator to cater for the inclusion of trigraphs and keyword profiles.
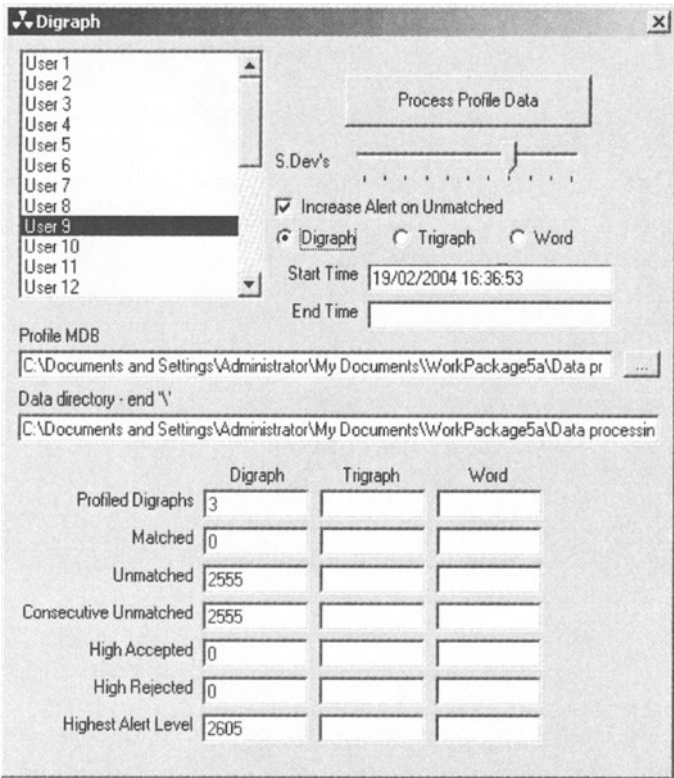


*Figure 2.* Data comparator (running)

In the previous trial, when a digraph was processed that did not exist in the reference profile, the alert level remained static (simply increasing the count of unmatched digraphs). This trial considered the role of unmatched samples as they are a potential indicator of impostor activity. I.e. if a user types a specific sample infrequently (to the extent that there is insufficient data on which to base a profile), it is reasonable to assume that these occurrences are un-representative of that user's normal typing behaviour. By default, in this trial, an unmatched sample increased the alert level by one, whilst a matched accepted/rejected sample varied the alert level by two accordingly. This behaviour can be adjusted by selecting the checkbox in the comparator – once unchecked; the alert level was not affected by unmatched samples.

Before starting the full profile comparisons a trial comparison was conducted based on a random selection of five users in order to determine the optimum settings for the deviation threshold. In the previous study the deviation settings were chosen from a range of 0.5, 1.0, 1.5 and 2.0 standard deviations with the best results obtained at 0.5. In order to determine an optimum setting, profile comparisons were made between 0.5 and 1.0 standard deviations (values below 0.5 had already been assessed in earlier trials). For the randomly selected users the best results were obtained at 0.7 with an increase in alert level above and below this threshold. As such, the later comparisons were performed with standard deviations settings of 0.6, 0.7 and 0.8. The permitted deviation was determined by the slider control that selects the number of standard deviations from the mean.

digraph mean ± (digraph standard deviation * permitted deviation)

Once the profile comparison was started each users' reference profile was loaded and then compared against the raw keylogger data files for all 35 users. This resulted in a table of 35 sets of statistics for each user. This process was repeated for trigraphs and keywords with three different profile deviation settings (0.6, 0.7 and 0.8 standard deviations from the mean). NB a setting of 0.5 standard deviations was introduced to the trigraph comparisons due to poor performance at 0.6 and 0.7 and unmatched alert increases were optionally applied to digraphs and trigraphs (hence doubling the number of comparisons for these metrics). With an average of nearly 100,000 samples per data file, each data comparison took approximately two hours with a total of 17 comparisons conducted – six for digraph, eight for trigraph and three for keywords (see Table 3).

*Table 3.* Profile comparison settings

| Metric | Standard Deviations (S.D.) |
|--------|----------------------------|
| Digraphs | 0.6, 0.7, 0.8 S.D. |
| Trigraphs | 0.5, 0.6, 0.7, 0.8 S.D. |
|  | *0.5 added due to poo performance at 0.6 and 0.7* |
| Keyword | 0.6, 0.7, 0.8 S.D. |

Once the profile comparison was completed, the results were exported and a number of functions were used to derive 2-dimensional tables of data from the raw results from the comparator from which the FAR/FRR figures could be derived.

Following the basic analysis described in this section, a further modification was made to the comparator to determine how many keystrokes

were needed before either the valid user was challenged or an impostor detected. The threshold for this challenge was based upon the best performance thresholds from the earlier trials and was initially set at an alert level of 70. The results from this trial using the digraph keylogger files at a threshold of 0.7 standard deviations is presented in Table 4. The results from this trial were somewhat variable, while some users had good results (e.g. user 7, 10 and 26), most user profiles had only moderately successful results. If we consider user 2, while 29/34 (85%) impostors were challenged in less than 100 digraphs, user 16 (when acting as an impostor against user 2's reference profile) was able to type over 40,000 digraphs before being challenged.

The results in Table 4 can also be considered in terms of the average number of keystrokes required before a challenge is issued. The results show that an average of 6,390 digraphs were accepted before an impostor was challenged compared with an average of 68,755 digraphs before the valid user was challenged. While these results seem to provide the appropriate differentiation between impostor and valid user, giving an impostor the opportunity to type over 6,000 digraphs presents a major security risk.

*Table 4.* Number of keypresses before a challenge

|  | User 1 | User 2 | User 3 | User 4 | User 5 | User 6 | User 7 | User 8 | User 9 | User 10 | User 11 | User 12 | User 13 | User 14 | User 15 | User 16 | User 17 | User 18 | User 19 | User 20 | User 21 | User 22 | User 23 | User 24 | User 25 | User 26 | User 27 | User 28 | User 29 | User 30 | User 31 | User 32 | User 33 | User 34 | User 35 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0.7 SD | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

For this trial the False Rejection Rate (FRR) was fixed at 0% (i.e. the valid user would not be rejected by the system). The False Acceptance Rates (FAR's) were then calculated for each user at the deviation thresholds specified in Table 3, and are shown in Table 5.

*Table 5.* Results from single-metric measures

| Standard Deviation (Unmatched Alert) | Digraph FAR 0.6 | | Digraph FAR 0.7 | | Digraph FAR 0.8 | | Trigraph FAR 0.5 | | Trigraph FAR 0.6 | | Trigraph FAR 0.7 | | Trigraph FAR 0.8 | | Keyword FAR 0.5 | Keyword FAR 0.6 | Keyword FAR 0.7 | Keyword FAR 0.8 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | N | Y | N | Y | N | Y | N | Y | N | Y | N | Y | N | Y | N | N | N | N |
| User 1 | 2.9 | 2.9 | 2.9 | 8.6 | 2.9 | 8.6 | 8.6 | 17.1 | 8.6 | 8.6 | 8.6 | 25.7 | 8.6 | 25.7 | 97.1 | 97.1 | 91.4 | 91.4 |
| User 2 | 0.0 | 2.9 | 2.9 | 5.7 | 2.9 | 11.4 | 34.3 | 0.0 | 34.3 | 34.3 | 34.3 | 5.7 | 31.4 | 5.7 | 2.9 | 2.9 | 2.9 | 5.7 |
| User 3 | 0.0 | 0.0 | 0.0 | 5.7 | 2.9 | 11.4 | 74.3 | 5.7 | 62.9 | 62.9 | 54.3 | 5.7 | 40.0 | 25.7 | 0.0 | 8.6 | 20.0 | 28.6 |
| User 4 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 2.9 | 5.7 | 0.0 | 5.7 | 5.7 | 5.7 | 0.0 | 5.7 | 0.0 | 5.7 | 0.0 | 0.0 | 0.0 |
| User 5 | 0.0 | 2.9 | 0.0 | 8.6 | 0.0 | 11.4 | 28.6 | 2.9 | 28.6 | 28.6 | 28.6 | 14.3 | 28.6 | 22.9 | 8.6 | 11.4 | 14.3 | 20.0 |
| User 6 | 0.0 | 0.0 | 0.0 | 8.6 | 0.0 | 22.9 | 28.6 | 2.9 | 28.6 | 28.6 | 28.6 | 5.7 | 25.7 | 11.4 | 34.3 | 11.4 | 2.9 | 8.6 |
| User 7 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 2.9 | 34.3 | 0.0 | 28.6 | 28.6 | 25.7 | 0.0 | 20.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| User 8 | 11.4 | 5.7 | 2.9 | 5.7 | 2.9 | 40.0 | 28.6 | 17.1 | 28.6 | 28.6 | 25.7 | 28.6 | 25.7 | 37.1 | 0.0 | 0.0 | 2.9 | 8.6 |
| User 9 | 2.9 | 2.9 | 2.9 | 0.0 | 0.0 | 14.3 | 17.1 | 5.7 | 17.1 | 17.1 | 17.1 | 11.4 | 14.3 | 20.0 | 22.9 | 25.7 | 34.3 | 34.3 |
| User 10 | 0.0 | 0.0 | 0.0 | 8.6 | 2.9 | 14.3 | 37.1 | 0.0 | 31.4 | 31.4 | 28.6 | 2.9 | 22.9 | 8.6 | 2.9 | 0.0 | 0.0 | 0.0 |
| User 11 | 0.0 | 0.0 | 0.0 | 14.3 | 17.1 | 22.9 | 82.9 | 2.9 | 65.7 | 65.7 | 51.4 | 0.0 | 11.4 | 11.4 | 37.1 | 5.7 | 2.9 | 28.6 |
| User 12 | 0.0 | 5.7 | 0.0 | 5.7 | 14.3 | 14.3 | 2.9 | 14.3 | 2.9 | 2.9 | 2.9 | 0.0 | 2.9 | 28.6 | 22.9 | 25.7 | 31.4 | 40.0 |
| User 13 | 8.6 | 11.4 | 5.7 | 11.4 | 0.0 | 11.4 | 40.0 | 2.9 | 40.0 | 40.0 | 37.1 | 11.4 | 37.1 | 11.4 | 8.6 | 8.6 | 11.4 | 14.3 |
| User 14 | 2.9 | 2.9 | 2.9 | 8.6 | 0.0 | 34.3 | 8.6 | 8.6 | 8.6 | 8.6 | 8.6 | 22.9 | 5.7 | 28.6 | 0.0 | 2.9 | 5.7 | 5.7 |
| User 15 | 0.0 | 0.0 | 0.0 | 2.9 | 8.6 | 8.6 | 0.0 | 14.3 | 0.0 | 0.0 | 0.0 | 17.1 | 0.0 | 20.0 | 22.9 | 17.1 | 25.7 | 25.7 |
| User 16 | 0.0 | 5.7 | 0.0 | 0.0 | 5.7 | 5.7 | 22.9 | 0.0 | 20.0 | 20.0 | 14.3 | 2.9 | 14.3 | 5.7 | 0.0 | 0.0 | 2.9 | 14.3 |
| User 17 | 0.0 | 0.0 | 2.9 | 11.4 | 14.3 | 25.7 | 48.6 | 2.9 | 48.6 | 48.6 | 45.7 | 8.6 | 40.0 | 11.4 | 5.7 | 5.7 | 5.7 | 11.4 |
| User 18 | 8.6 | 11.4 | 17.1 | 8.6 | 17.1 | 25.7 | 48.6 | 17.1 | 42.9 | 42.9 | 40.0 | 0.0 | 37.1 | 5.7 | 28.6 | 8.6 | 11.4 | 17.1 |
| User 19 | 0.0 | 2.9 | 2.9 | 5.7 | 2.9 | 8.6 | 65.7 | 0.0 | 62.9 | 62.9 | 62.9 | 8.6 | 62.9 | 11.4 | 5.7 | 14.3 | 20.0 | 5.7 |
| User 20 | 0.0 | 0.0 | 0.0 | 5.7 | 5.7 | 22.9 | 42.9 | 0.0 | 40.0 | 40.0 | 31.4 | 5.7 | 28.6 | 8.6 | 0.0 | 2.9 | 2.9 | 8.6 |
| User 21 | 0.0 | 0.0 | 0.0 | 8.6 | 2.9 | 2.9 | 40.0 | 0.0 | 28.6 | 28.6 | 17.1 | 2.9 | 5.7 | 2.9 | 0.0 | 2.9 | 5.7 | 11.4 |
| User 22 | 2.9 | 8.6 | 5.7 | 8.6 | 8.6 | 25.7 | 65.7 | 0.0 | 57.1 | 57.1 | 45.7 | 5.7 | 31.4 | 17.1 | 0.0 | 5.7 | 5.7 | 45.7 |
| User 23 | 68.6 | 54.3 | 51.4 | 62.9 | 60.0 | 80.0 | 71.4 | 45.7 | 71.4 | 71.4 | 68.6 | 34.3 | 68.6 | 45.7 | 37.1 | 37.1 | 45.7 | 14.3 |
| User 24 | 2.9 | 0.0 | 2.9 | 5.7 | 5.7 | 8.6 | 62.9 | 2.9 | 54.3 | 54.3 | 25.7 | 14.3 | 14.3 | 28.6 | 0.0 | 2.9 | 8.6 | 0.0 |
| User 25 | 0.0 | 2.9 | 0.0 | 2.9 | 0.0 | 14.3 | 14.3 | 0.0 | 8.6 | 8.6 | 8.6 | 2.9 | 8.6 | 0.0 | 2.9 | 0.0 | 0.0 | 5.7 |
| User 26 | 0.0 | 2.9 | 0.0 | 8.6 | 14.3 | 5.7 | 22.9 | 0.0 | 22.9 | 22.9 | 20.0 | 0.0 | 17.1 | 0.0 | 5.7 | 0.0 | 0.0 | 5.7 |
| User 27 | 2.9 | 2.9 | 2.9 | 17.1 | 2.9 | 48.6 | 11.4 | 11.4 | 11.4 | 11.4 | 17.1 | 0.0 | 11.4 | 45.7 | 8.6 | 2.9 | 5.7 | 37.1 |
| User 28 | 2.9 | 2.9 | 2.9 | 5.7 | 2.9 | 17.1 | 42.9 | 8.6 | 42.9 | 42.9 | 11.4 | 28.6 | 42.9 | 14.3 | 8.6 | 17.1 | 28.6 | 31.4 |
| User 29 | 5.7 | 5.7 | 2.9 | 22.9 | 20.0 | 48.6 | 48.6 | 11.4 | 42.9 | 42.9 | 42.9 | 20.0 | 42.9 | 22.9 | 11.4 | 8.6 | 14.3 | 8.6 |
| User 30 | 14.3 | 8.6 | 14.3 | 14.3 | 11.4 | 20.0 | 22.9 | 20.0 | 22.9 | 22.9 | 42.9 | 22.9 | 20.0 | 37.1 | 45.7 | 28.6 | 28.6 | 5.7 |
| User 31 | 0.0 | 0.0 | 2.9 | 8.6 | 5.7 | 8.6 | 0.0 | 0.0 | 0.0 | 0.0 | 22.9 | 2.9 | 0.0 | 5.7 | 0.0 | 0.0 | 2.9 | 14.3 |
| User 32 | 0.0 | 0.0 | 0.0 | 14.3 | 11.4 | 20.0 | 54.3 | 0.0 | 82.9 | 82.9 | 0.0 | 0.0 | 0.0 | 14.3 | 2.9 | 5.7 | 14.3 | 28.6 |
| User 33 | 17.1 | 14.3 | 17.1 | 14.3 | 22.9 | 28.6 | 88.6 | 77.1 | 48.6 | 48.6 | 65.7 | 54.3 | 57.1 | 71.4 | 80.0 | 68.6 | 20.0 | 97.1 |
| User 34 | 5.7 | 2.9 | 11.4 | 34.3 | 14.3 | 14.3 | 60.0 | 0.0 | 65.7 | 65.7 | 42.9 | 0.0 | 37.1 | 2.9 | 97.1 | 97.1 | 97.1 | 14.3 |
| User 35 | 22.9 | 14.3 | 5.7 | 8.6 | 25.7 | 54.3 | 71.4 | 25.7 | 65.7 | 65.7 | 65.7 | 25.7 | 62.9 | 45.7 | 34.3 | 5.7 | 11.4 | 14.3 |
| Average | 5.2 | 5.0 | 4.9 | 9.8 | 8.1 | 20.5 | 38.2 | 9.1 | 33.3 | 33.3 | 29.5 | 13.0 | 25.2 | 18.9 | 18.3 | 15.2 | 16.5 | 20.2 |

When the results were calculated, the False Acceptance Rates per user were averaged across all users to provide an average FAR for each metric. The averaged results for the statistical approach are shown in Table 6. It

should be noted that the keyword latencies did not use the unmatched alert increase due to the use of a word list/dictionary – i.e. many words would not be matched in the users' profile.

*Table 6.* Final statistical results (best results highlighted)

| Metric | S.D. | Unmatched Alert | FAR |
|---|---|---|---|
| Digraphs | 0.6 | No | 5.2% |
| | 0.6 | Yes | 5.0% |
| | 0.7 | No | 4.9% |
| | 0.7 | Yes | 9.8% |
| | 0.8 | No | 8.1% |
| | 0.8 | Yes | 20.5% |
| Trigraphs | 0.5 | No | 38.2% |
| | 0.5 | Yes | 9.1% |
| | 0.6 | No | 33.3% |
| | 0.6 | Yes | 33.3% |
| | 0.7 | No | 29.5% |
| | 0.7 | Yes | 13.0% |
| | 0.8 | No | 25.2% |
| | 0.8 | Yes | 18.9% |
| Words | 0.5 | No | 18.3% |
| | 0.6 | No | 15.2% |
| | 0.7 | No | 16.5% |
| | 0.8 | No | 20.2% |

# 5.      DISCUSSION

While the results shown in Table 6 show some encouraging FAR levels there is still significant variation with the best results obtained at 0.7 standard deviations for digraphs, 0.5 standard deviations for trigraphs (with increased alert levels for unmatched digraphs) and 0.6 for keywords. However, when the full results are considered (as shown in Table 5), even at the optimum settings, certain users show high FAR levels (e.g. user 23's profile returned FAR levels of 51.4%, 45.7% and 37.1% respectively for digraph, trigraph and keywords at the average optimum settings). It can also be clearly observed that the results for trigraphs and keywords are significantly worse when compared with those for digraphs – this is most likely to be related to the number of underlying samples used for these techniques (i.e. the number of sampled digraphs were significantly higher than that for trigraphs and keywords, with a corresponding increase of samples per digraph). It is probable that over a longer period of time, the profiles could be refined for trigraphs and keywords to produce a more distinct user profile with a corresponding reduction in the FAR.

These results also demonstrate that the techniques can be very effective for some users while very ineffective for others. For example, when considering digraph FAR's at 0.6 standard deviations (where 0% FAR was actually experienced for 19 out of the 35 users – 54.3%) the average FAR (5.2%) has been heavily influenced by a single user (user 23) whose 68.6% FAR dramatically increases the average. In a full implementation, the authors propose that the use of keystroke analysis should only form a part of a comprehensive user monitoring system. As such, a users' typing would only be monitored if the method was shown to be a discriminating authentication technique for that user. The removal of user 23 from the results in Table 5 significantly affects the average FAR's presented in Table 6, reducing the best digraph results from 4.9% to 3.5% , trigraph results from 9.1% to 8.0% and keywords from 15.2% to 14.5%.

Further optimisation can be achieved by removing the worst 5 participants (15%) from the trial results. This provides a significant improvement in the results of the technique with average FAR's as low as 1.7% for digraphs, 4.4% for trigraphs and 12.8% for keywords (Table 7). While the keyword FAR in particular remains unacceptably high, a reference back to Table 5 reveals that there were still almost a third of users for whom 0% FAR was observed at the 0.5 standard deviation threshold. This suggests a clear potential for using the technique in a subset of cases – which could also increase if additional keyword typing samples were obtained to support the profiling.

*Table 7.* Optimised results

| Metric | S.D. | Unmatched Alert | FAR |
|---|---|---|---|
| Digraphs | 0.6 | No | 1.7% |
| | 0.6 | Yes | 2.4% |
| | 0.7 | No | 2.2% |
| | 0.7 | Yes | 7.0% |
| | 0.8 | No | 4.9% |
| | 0.8 | Yes | 17.0% |
| Trigraphs | 0.5 | No | 34.5% |
| | 0.5 | Yes | 4.4% |
| | 0.6 | No | 29.3% |
| | 0.6 | Yes | 29.3% |
| | 0.7 | No | 25.6% |
| | 0.7 | Yes | 10.6% |
| | 0.8 | No | 21.2% |
| | 0.8 | Yes | 15.2% |
| Words | 0.5 | No | 13.8% |
| | 0.6 | No | 12.8% |
| | 0.7 | No | 15.3% |
| | 0.8 | No | 19.7% |

The removal of a number of specific user accounts from the keystroke monitoring process is not an ideal solution to the problem of poor user authentication. Keystroke analysis is unlikely to be used as a sole-method of user authentication, instead, it is envisaged that the methods described in this paper would form a part of a larger authentication system and would be only one of a range of authentication metrics that each user could be monitored with. With a larger number of users (and hence a wider range of user typing abilities and corresponding authentication rates) there is likely to be a proportional increase in the number of users for whom keystroke analysis does not produce appropriate FAR/FRR rates – in these cases other, more appropriate, techniques would have to be used. Identifying the cause of poor user performance when using keystroke analysis is vital; on-going work within the authors' research group will conduct further analysis on the gathered data sets to try to determine the cause of the variation between users and identify common factors (e.g. users' typing abilities, differences between application usage etc.).

# 6.      CONCLUSIONS

It is clear from the results presented in this paper that there is considerable potential for continuous user authentication based on keystroke analysis. The long-term sampling of digraph keystrokes has served to reinforce the validity of the technique, while the introduction of trigraph and keyword monitoring has provided additional metrics that can be used as alternative (or complimentary) techniques. In particular, the use of keyword monitoring has considerable potential when used to monitor for specific, high-risk typed words (e.g. delete, format etc.).

It is also clear that the simple statistical approach does not provide sufficient distinction for all users and a live implementation would have to consider which metric (if any) is most appropriate for each user. It is envisaged that keystroke analysis would become only one of a number of monitoring characteristics used by a more comprehensive system with other authentication and supervision techniques.

Future work will also consider how the individual keystroke metrics can be combined together. For example, by combining the confidence measures of multiple metrics (e.g. monitoring digraphs and trigraphs), coupled with monitoring specific keywords (e.g. the typing patterns for high-risk words – format, delete etc.), it may be possible to provide a higher level of

confidence in the authentication of the user. The potential for this method will be considered in a later paper.

## 7.　ACKNOWLEDGMENTS

## 8.　REFERENCES

Brown M. and Rogers S.J., 1993, "User identification via keystroke characteristics of typed names using neural networks", *International Journal of Man-Machine Studies*, vol. 39, pp999-1014.

Card S.K., Moran T.P. & Newell A., 1980. "Computer text-editing: An information-processing analysis of a routine cognitive skill", *Cognitive Psychology*, vol. 12, pp32-74.

Dowland P.S., Furnell S.M. & Papadaki M., 2002, "Keystroke Analysis as a Method of Advanced User Authentication and Response", *Proceedings of IFIP/SEC 2002 - 17th International Conference on Information Security*, Cairo, Egypt, 7-9 May, pp215-226.

Furnell, S.M., 1995, "Data Security in European Healthcare Information Systems", PhD Thesis, University of Plymouth, UK.

Gaines R., Lisowski W., Press S. and Shapiro N., 1980, "Authentication by Keystroke Timing: some preliminary results", *Rand Report R-256-NSF*, Rand Corporation.

Joyce R. and Gupta G. 1990. "Identity authentication based on keystroke latencies", *Communications of the ACM*, vol. 33, no. 2, pp168-176.

Legett J., Williams G., Usnick M. and Longnecker M., 1991, "Dynamic identity verification via keystroke characteristics", *International Journal of Man-machine Studies*, vol. 35, pp859-870.

Mahar D., Napier R., Wagner M., Laverty W., Henderson R.D. and Hiron M., 1995, "Optimizing digraph-latency based biometric typist verification systems: inter and intra typist differences in digraph latency distributions", *International Journal of Human-Computer Studies*, vol. 43, pp579-592.