DragPIN: A Secured PIN Entry Scheme to Avert Attacks

Rajarajan Srinivasan
School of Computing, SASTRA University, India

Abstract: *Personal Identification Numbers (PIN) are widely used for authenticating users for financial transactions. PIN numbers are entered at Automatic Teller Machine (ATMs), card payments at Point of Sale (POS) counters and for e-banking services. When PIN numbers are keyed in by the users, they are vulnerable to shoulder surfing and keylogging attacks. By entering PIN numbers through virtual keyboards, the keylogging attacks can be mitigated, but it elevates the risk of shoulder surfing. A number of shoulder surfing resistive keyboard schemes have been proposed. But many of them offer inadequate security and are poor in usability. They also demand substantial user intelligence, training, user memory and additional devices for entering the PIN numbers. Keeping in mind that securing PIN number should not be done at the cost of user inconvenience, a new scheme based on key sliding is proposed in this paper. Two variations of the scheme are presented. They are based on manual and automatic sliding of keys and indirect user entry of PIN numbers. Our proposed schemes are simple and easy to adopt. They are sufficiently stronger against attacks. Our extensive analysis and user study of the schemes have proved their security and usability.*

Keywords: *PIN, Shoulder surfing, keylogging, virtual keyboard, user authentication, e-banking, man-In-the-middle attacks.*

Received September 24, 2014; accepted August 12, 2015

1. Introduction

Personal Identification Numbers (IPIN) numbers comprise of 3 to 6 digits and are either assigned by the service providers or chosen by the users. PIN numbers are normally associated with various types banking services. To complete a transaction, users would be required to supply their PIN number assigned to their accounts. The user submitted numbers will be verified against the stored numbers. Sometimes a dynamically generated number called One Time Password (OTP) may be used as the PIN number. Though PIN numbers are simple and effective in securing the accounts, they are prone to shoulder surfing attacks. In a shoulder surfing attack, an attacker observes the authentication process of the user and learns the PIN number. Usage of virtual keyboards makes the attacker's job much easier since the keyboard entries are made on the screen. A safety measure to prevent this attack is ensuring that no one is around before typing the PIN number. But at public places like Automatic Teller Machine (ATM) centres, cyber cafes, departmental stores etc. it is difficult to enforce that. Another option is to make use of OTPs for transactions. But they may incur additional costs and delays. OTP attacks are also prevalent [12].

In case of a human shoulder-surfing attack, the attackers rely on their ability to observe and remember the details that they have observed [8, 11, 15, 18]. In a PIN entry process on a virtual keyboard, the user directly clicks the numbers one by one and that gives enough opportunity for the observer to note down the individual digits to reconstruct the whole PIN. So any Security mechanism that avoids the direct entry of digits and increases the difficulty of an attacker to observe the PIN entry in order to trace the actual number will suffice to mitigate shoulder surfing attacks. But when a shoulder surfing attack is aided by a recording device such as a mobile camera or a malware that could record the on-screen activities to generate a video is very difficult to defend [20]. This is because the attacker could view the recorded video any number of times and gradually reproduce the PIN number. There are proposals made to curb these kinds of recorded shoulder-surfing attacks. But such schemes are more complex to implement and are tedious to be followed by normal users.

2. Related Works

Recognizing the possibility of attacks on PIN numbers during the PIN entry process, numerous researchers have focused on developing new schemes that are aimed at mitigating those attacks. A survey of numerous virtual keyboards is done in [4]. The method of [19] requires that the user should perform a mathematical operation on each digit of his PIN with a random number provided by the authenticator. The result is entered by the user. At the server end, the same calculations are repeated to retrieve the digits. They are verified against the actual PIN digits stored. This approach demands some level of competence from users to carry out the mathematical calculations and that may also result in several incorrect entries.

The scheme called ColorPIN was proposed for PIN entries at ATM centres [2]. In this scheme, each PIN digit is associated with a colour. At the PIN entry interface, each digit has got three random alphabets displayed in three colours at its bottom. Every alphabet repeats three times in all the three colours under different numbers on the keypad. The user should recall the PIN digit and the colour associated with that digit. He should then identify the alphabet displayed in that colour and type the same on the keypad. Since the same alphabet appears multiple times, an observer will not be able to detect the PIN digit even if he knows the alphabets entered. But by recording the PIN entry process and the sequence of four alphabets entered, an attacker may succeed in revealing the PIN number in a maximum of 81 attempts. A difficulty with this scheme is that the users have to remember the PIN digits as well as their respective colours. The model of this scheme is presented in Figure 1.

Figure 1. Colour PIN-PIN entry based on secret colors and alphabets [2].

A scheme based on challenge response paradigm and a table look up was proposed in [9]. Here the user has to first listen to a random challenge digit over a secured ear phone connection. Then by looking over the look-up table consisting of the next PIN digit and the challenge digit, he need to click on a response button that depicts the relative position of the challenge digit with respect to the PIN digit. The procedure has to be repeated for each digit. This scheme requires a secured channel for communicating the challenge to the user. The PIN entry scheme of [14] is based on the concept of cognitive trapdoor games. In this scheme, the user should successfully play and win a game to complete his PIN entry. The keypad displays two group of digits randomly organized into white and black. There are two buttons at the bottom representing the two colours. In each round, the user has to recognize his PIN digit either in the white group or in the black group and press the button for that colour. For each round, the colour combinations of the numbers will be rearranged. There is a time limit of 0.5 seconds set for the display of the keypad. Though this scheme offers adequate security for both cognitive SSAs and recording attacks, it encourages the attacker to instigate a random guessing based attack since the PIN digits are not precisely entered and only a group has to be chosen.

Figure 2. PIN numbers displayed in two distinct colored sets [14].

A different PIN scheme based on sequence of locations instead of digits is proposed in [16]. The ATM keypad presents a table of A x A size. Each cell in the table contains a number. Every number is repeated A times and placed at random positions. In each round of the PIN entry, the user has to press the number displayed within the cell that corresponds to the position pertaining to that round. This scheme offers limited resilience against attacks based on recording devices. A simple scheme more suitable for defying cognitive attacks by observers is presented in [6]. The scheme named SwitchPIN incorporates a switch button at the bottom of the numeric keypad. When the users press and hold the switch button, the numbers on the keypad are randomized. Users should notice and remember the digits in the actual positions of their four PIN digits presently displayed on the challenge pad before releasing the switch key. Once the switch key is released, the keypad is restored to the regular mode. Now users should correctly enter the four digits they identified in the challenge pad previously. It offers no resistance against any form of recording attacks. Another PIN entry scheme that resists human shoulder surfing is proposed in [3]. It comprise of a user interface that has two arrays with digits and symbols along with a left and right arrow buttons. There are two stages and three rounds to complete the entry of a four digit PIN. In the first stage, the ten digits from 0 to 9 are shown along with ten symbols. Users should note down the symbol displayed under their first PIN digit of their PIN number. This is called pass-object and it will be used as a token in the remaining stages. In the each subsequent stage, users are required to scroll the symbols such that their pass-object is aligned under the current PIN digit. While being resistant against cognitive shoulder surfing attacks, this scheme is weaker in the face of a recording induced attack. If an attacker succeeds in obtaining the screen shots of the four rounds of the PIN entry process, there will be ten possible sequences of PIN entries as there are ten symbols available.

There are schemes that involve the usage of additional devices for PIN entries. One such scheme is called mobile PIN [1]. Certain schemes [1, 9] also involve a smart phone with camera facility. A

comparison of the various proposed schemes and our proposed schemes are presented in Table 1.

Review of the existing schemes has revealed that many of them have the following limitations:

- Certain schemes are only capable of preventing Human Shoulder Surfing Attacks (HSSAs) and are not useful for protecting against Recording based Shoulder Surfing Attacks (RSSAs) [18].
- Some schemes employ additional tools or devices such as mobile phones, audio headphones, sensors, cameras etc. besides the user terminals and the authenticating servers [5, 13]. This makes their implementation to be expensive.
- Most schemes require users to go through multiple rounds, typically 'n' rounds for 'n' digits of a PIN number to accomplish the PIN entry. It makes PIN entry time consuming and tedious.
- There are schemes that involve some arithmetic calculations to be carried out by users and to use their results for the PIN entry. This needs some level of competence from the users to do the mathematical operations mentally.

Table 1. Comparison of pin entry schemes based on their security and usability.

Proposed Scheme	Resistance to Shoulder Surfing	Usability	User Role	No. of Rounds /Stages	Implementation Overhead
SecurePINEntry[19]	HSSA: High RSSA: High	Moderate	Users need to perform mathematical operation	1	Secured channel needed for informing random number to user.
ColorPIN [2]	HSSA: High RSSA: Mod.	Moderate	Users should remember their PIN digits with their respective colours in order to identify the alphabets in the same colours	1	Keypad should include ten random alphabets that are displayed in three different colours under each digit
CognitiveTrap DoorGames[14]	HSSA: High RSSA: Poor	Poor	Users should observe and identify the colour of the group into which their PIN digit belongs to	4	For each round, the keypad numbers are randomized and divided under two sets of distinct colours
SwitchPIN[6]	HSSA: High RSSA: Low	Moderate	Users should note down positions of PIN digits on the challenge pad	2	Key pad values randomized when a button pressed and reverts when button released
PassObjects[3]	HSSA: High RSSA: Low	Moderate	Users should remember the PassObject displayed in first round to use for PIN entry	4	Keypad comprising of the ten digits and ten symbols that are randomized for each stage
Visual Authentication Protocols[10]	HSSA: High RSSA: High	Poor	User should take a snap shot of the QCODE by their phone camera and use the keyboard displayed on the phone.	2	Every user should posses a camera that is equipped a camera and is installed with the necessary software and a private key for decryption.
STL[9]	HSSA: High RSSA: High	Poor	User should listen to the challenge, locate that in the table and press response button correctly	4	User should be informed of the challenge value by voice through a secured ear-phone
Gaze-Based Password Entry[5]	HSSA: High RSSA: High	Poor	User should correctly position their eye pupil on the necessary key	1	The user terminal should be equipped with eye tracking devices to capture the user's eye movements.
RoratyPIN [17]	HSSA: Mod. RSSA: Poor	Moderate	User should rotate the circles and place all the PIN digits along one sector	1 or 2	In a circular keyboard, digits from 0 to 9 have to be placed randomly in four co-centred wheels and each wheel could be rotated independently.
DragPIN: Manual	HSSA: High RSSA: Mod.	High	Users need to align same coloured squares on their four PIN positions using the left/right scroll buttons	1	Five coloured squares are to be chosen and they should randomly positioned on the screen with each colour repeated twice
DragPIN: Auto	HSSA: High RSSA: High	High	Users need to choose the same alphabet on their PIN positions by pressing the space bar	1	Five alphabets in different colours are to be animated to move from left to right
HSSA: Human Shoulder Surfing Attack RSSA: Recorded Shoulder Surfing Attack					

3. Security and Usability

For any user authentication scheme to be adopted for regular usage, it needs to satisfy the requirements of usability and security. But the two requirements generally do not go hand in hand. Attempting to strengthen security would invariably worsen the usability of the scheme. So the need is to find a fine balance of between the two. According to our analysis of existing schemes, we have classified their usability and security as Poor, Moderate and High. The basis of our classification of the schemes is provided below:

a. *Usability*:

- *User competence required*: Some schemes require the users to posses some amount of capability to accomplish the PIN entry.
- *Memory requirements*: Certain schemes obligate users to remember and recall additional information other their PIN numbers to complete their PIN entry.
- *No. of rounds / stages involved*: The PIN entry process itself may have to be completed by going
- Through multiple rounds. This would prolong the time of PIN entry and could annoy the users.

- *Usage of secondary channels*: There are schemes that involve additional channels as part of their implementation. For example a scheme demands the usage of head phone to listen to a voice data, another scheme involves the usage of mobile phone camera.

According to the above aspects we have classified the reviewed schemes. A scheme is assigned 'poor usability' if it demands all of the above. A 'moderate usability' is one that involves any two and a scheme is identified to have 'high usability' if it have none of those overheads. In addition to rigours analysis of the proposed methodologies of those schemes, we had also considered the usability study results of the schemes.

b. *Security*: The security of the schemes is assessed on two aspects. Its expected defence against HSSA and its defence against RSSA. By careful analysis of each scheme, we have assessed their level of protection against each of those attacks. This information is also normally declared by the authors in the security analysis sections. A scheme that could withstand any form of attack of HSSA or RSSA is identified to have 'High security' in the respective security category. Schemes that only promise reasonable security for HSSA or RSSA are treated as possessing 'Moderate security'. Some schemes only offer protection against HSSA but not against RSSA. Such schemes are given 'Poor' for RSSA.

The result of the comparison of the existing schemes that is presented in Table 1 presents the relative strength and weaknesses of the reviewed schemes. The table also points out that our proposed schemes have score higher over many of the existing schemes.

4. Scope and Contribution

In this paper, we are proposing an innovative PIN entry scheme in two variants. Our proposed schemes differ substantially from other schemes due to their strong defence against human and recorded shoulder surfing attacks and their excellent usability. Through stringent analysis and rigorous user studies, we have established that our proposed schemes are able to withstand many of the attacks against which other schemes that are found in literature are either ineffective or only moderately effective. Even the schemes that are found to be stronger against attacks are weaker in usability due to their complex procedures or due to their usage of additional devices. Our original contributions are the following:

- Detailed review and comparison of the existing schemes.
- Two schemes to prevent attacks on PIN entry based on the sliding of alphabets.
- Implementation of the proposed schemes. Through user testing we were able to demonstrate that the proposed schemes are apt for real world installations and usage.

5. Paper Organization

The remaining chapters of the paper are organized as follows. The proposed scheme is narrated in section 6. In section 7, the security analysis of the proposed scheme is presented. User study results are discussed in section 8. Section 9 provides comprehensive discussion of the proposed schemes and section 10 concludes the paper along with the future scopes and research directions.

6. Proposed PIN Entry Scheme

In this section, we present the model of our proposed PIN entry system. There are two variations of the scheme constructed based on the basic methodology. We first narrate the basic building blocks of the two schemes.

6.1. Basic Method

We consider that a PIN is a number consisting of N digits. Since the common size of PIN number is 4, we assume 4 as the value of N in our implementation. (PIN= d1d2d3d4). The N digits are a combination of any four digits from 0 to 9. Initially, the user is shown a 4 x 10 sized grid. The 4 rows represent the 4 digits of the PIN number and the 10 columns represent the ten digits from 0 to 9. A group of five English alphabets are randomly selected. Each row of the grid is loaded with all the five alphabets appearing twice in a single row at random column positions. But no two recurrences of the same alphabet would occur in consecutive positions on the same row. To successfully enter the PIN number, users need to align the same alphabet in all the four columns equivalent to their PIN digits in each of the four rows such that $G_{1,d1} = G_{2,d2,} = G_{3,d3} = G_{4,d4}$= A where A is one of the five alphabets. For example, if PIN number is 5269 and user's choice of alphabet is E, then he has to align the alphabet E on the 5^{th}, 2^{nd}, 6^{th} and 9^{th} columns of the 1^{st}, 2^{nd}, 3^{rd} and 4^{th} rows of the grid respectively. The screen shot for the alignment of this example is presented in Figure 2.

6.2. The PIN Entry Interface

The proposed scheme comprise of a user interface through which the user has to input his PIN number. It is designed to prevent shoulder surfing attacks. The elements of the user interface are briefed below:

- *User Interface*: The user interface comprise of a grid of ten columns and four rows. The ten columns represent the ten digits (0 to 9) to be entered by the users as per their actual PIN number. The four rows represent the four PIN digits

- *Coloured Alphabets*: There are a total of five alphabets presented on four squares painted with different colours and are loaded on the ten columns of the grid. Each alphabet is repeated twice on the same row. Figure 3 has the model of the four coloured alphabets that we used in our prototype.
- *Left/Right Arrows*: In the first variation of the scheme, the user have to use the left and right arrows of the keyboard to cause the sliding of the alphabets on the screen. Clicking the left arrow would cause all the alphabets in the current row to slide in the left direction by one column for a single click. Similarly pressing the right arrow would cause the movement in right direction.
- *Submit button*: This is placed at the bottom of the screen. Clicking this would cause the grid of 40 elements pertaining to the current arrangement made by the user to get forwarded to the authenticator for verification.

Our literature survey has revealed that the user interface has to be made simple and attractive. We opted for alphabets and colours to construct the interface for our scheme since users are already familiar with alphabets and colours. So they are more likely to feel encouraged to use our model without any discomfort. The design of PIN entry screen is same for both the variations of proposed scheme. In the next sections, we explain the principles and procedures of the two schemes that we have proposed.

Figure 3. Sample colour-alphabets used in the proposed scheme.

6.3. Manual-Sliding PIN Entry

This scheme makes use of the interface described in the previous section. When the user opens the screen for the PIN entry, the screen containing the text box for entering the User Id and the PIN entry interface are displayed. The PIN entry interface contains five alphabets embedded in five coloured squares randomly positioned on a grid of four rows and ten columns. Users have to make use of this interface for feeding their PIN entry into the system. The process for entering the PIN number involves the following steps:

1. User should mentally choose an alphabet for the PIN entry process. Since each alphabet also encompasses a unique colour, the user selection could be either one of the colours or an alphabet as per his preference.
2. By recalling the PIN digits of the PIN number, users must use either the left or right arrow keys of their keyboard to slide the alphabets in the left or the right direction so that the chosen alphabet is positioned on the column number equivalent to the PIN digit.
3. Once the first row's alignment for the first PIN digit is done, user should press the enter key to commence entry of the second PIN digit in the second row.
4. Now the pressing of the left / right arrows cause the sliding of alphabets in the second row. User should use them to get the same alphabet that was placed on the first PIN digit, to be positioned on the second PIN digit's column too.
5. The previous step has to be repeated two more times for entering the third and fourth digits.
6. When the enter key is entered for the fourth time, the system recognizes that the user has completed the entry of all four digits and ready to submit. Now the submitted button has to be clicked to forward the entries to the authenticator for verification.

For example, if the PIN number is 5269 and user opts for the alphabet E, then he should slide the grid rows to get the square holding E to come on the 5th column of the first row, 2nd column of the second row, 6th column of the third row and the 9th column of the fourth row. This is shown is Figure 4.

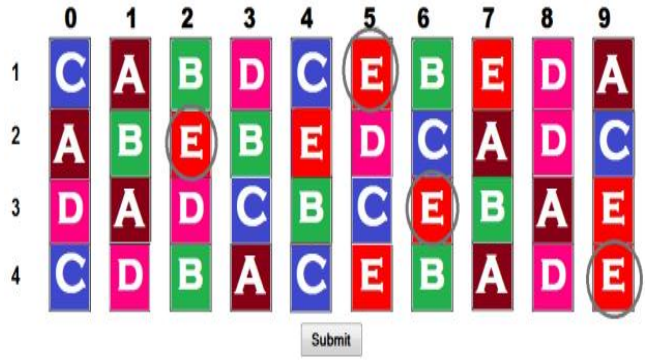

Figure 4. Screen shot of PIN entry for PIN no. 5269 with alphabet selection of 'E'. The arrangement of alphabet squares for the PIN positions is circled.

6.4. Auto-Sliding PIN Entry

In this section, the detail of the second variation of the proposed scheme is presented. Though the manual variant of the scheme is reasonably stronger against human shoulder surfing attacks, its strength against recording attacks is relatively lower. The second model of the scheme is based on the idea of animated sliding. In this model, the sliding of alphabets happens automatically based on a timer. The following steps summarize the PIN entry process of this variant:

1. Users should press the start button to commence the process.
2. They will now see the alphabets in the first row to start slide from right to left at a pre-determined speed. Only the first row slides now.
3. Users should focus on the column position equivalent to their first PIN digit of the PIN number.

4. When they see their chosen alphabet to appear on top of the column, they need to press the Space bar once. This conveys the system that the first row in its current instance has to be taken for the first PIN digit.
5. But the first row continues to slide.
6. To stop sliding of first row and to commence the sliding of the second row, the Enter key needs to pressed.
7. Now the sliding of the second row commences.
8. Following the same steps, users should enter the remaining PIN digits.
9. Once all the digits are entered, the submit button has to be pressed to submit the captured values for verification at the authenticator end.

For entering the PIN number 5269, user commence the sliding of alphabets by pressing the start button. Then the alphabets in the first row start to slide. User presses the space bar when the alphabet 'E' reaches fifth column. Then he presses the enter-key. User enters his second, third and fourth digits in the same way. Finally, the submit button is clicked by the user. Figure 5 contains the screen shot of the PIN entry for 5269. It is evident from the picture that unlike the manual variant, the column positions for 5269 do not hold the alphabet which the user actually entered for the PIN entry.

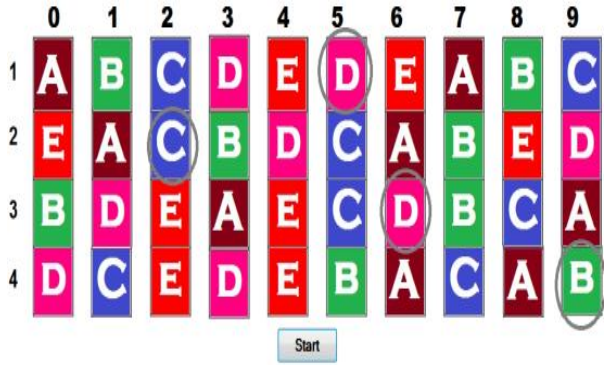

Figure 5. Screen shot of PIN entry interface of auto-sliding variant after the four digits are entered for the PIN number 5269.

6.5. Method of PIN Verification

A unique characteristic of our proposed schemes is that users do not directly enter their PIN numbers. If the actual PIN number is directly provided at the user terminal, then it is a simple task to forward it to the server and verify it against the stored value. But in our scheme, the user aligns four coloured alphabet squares relative to his PIN number. The actual PIN number is only known to the user and the server. So, for PIN verification, the PIN entry grid comprising of the 40 alphabets are converted into an array of numbers. Each alphabet is given a unique number. The same alphabet in different rows is represented using the same number. For example, the alphabet E is represented by number 3 in all the eight columns in the array. Table 2 contains the sample array of numbers representing the PIN entry made for 5269. The authenticator at the server will then retrieve the actual PIN number from the database and invoke the PIN verification module to verify the correctness of the user entered PIN number. The algorithm behind the PIN verification module is as follows:

Algorithm 1: Algorithm VerifyPIN.

```
VerifyPIN( PIN, PINArray )
d1 → PIN / 1000
PIN =PIN % 1000
d2 → PIN / 100
PIN →PIN % 100
d3 → PIN / 10
PIN → PIN % 10
d4 → PINNo
if PINArray[0, d1] = PINArray[1 ,d2] = PINArray[2, d3] = PINArray[3, d4] then
        return true
else
        return false
end if
end VerifyPIN
```

The algorithm extracts the four PIN digits from the actual PIN number. Using the PIN digits as index into the PIN array that is received from the client, it verifies whether the numbers in the respective positions are equal. For example, to verify the PIN number 5269, the following condition will be evaluated by the algorithm:

If PINArray[0, 5] = PINArray[1, 2] = PINArray[2, 6] = PINArray[3, 9] then
        return true
else
        return false

For the sample data presented in Table 2, the condition will result in true since all positions at [0, 5], [1, 2], [2, 6] and [3, 9] have the same number which is 9.

Table 2. Sample array for User's PIN Entry for 5269.

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
0	5	3	5	7	1	9	1	9	7	3
1	7	5	9	5	1	3	3	9	1	7
2	7	5	7	1	3	3	9	1	9	5
3	1	7	9	5	1	7	5	3	3	9

7. Security Analysis

PIN numbers are frequently compromised due to keylogging and shoulder surfing attacks. While keylogging attacks are easily tackled using virtual keyboards, there is not a very effective solution exist to counter the shoulder surfing attacks. Especially when recording devices are involved in the attack, it is much more difficult to defy them. Many of the proposed schemes are not high usability. So we have set as the goal of our proposed approach to meet the requirements of security and user friendliness. In this section, we provide a thorough analysis of our system for its security against several common attacks.

7.1. Man in the Middle Attacks

In this attack, the attacker eavesdrop the interaction between the sender and the receiver to capture the secret credentials. He could latter use those credentials to impersonate the user. When the user id and the PIN number are transmitted over the network, they are vulnerable for a Man in the Middle Attacks (MITM) attack. But in our approach, since the actual PIN number is never transmitted between the user and the server, an attacker could at most capture only a collection of numbers that do not directly reveal the PIN number. But if utmost security is preferred, we suggest one more tactics. In our implementation, we used the same numbers to represent all the eight occurrences of an alphabet. For example, number 9 represents alphabet 'E'. This is necessary for the verification of user entered PIN. In order to avoid giving any chance for a MITM attack, each one of the 40 square boxes may be given unique numbers from 1 to 40. So during a MITM attack, an attacker will just find forty numbers randomly positioned in an array. But the authenticator stores the groups of 8 numbers that represent the same alphabet and this could be used to map the eight numbers to a specific alphabet. For example, the group of numbers comprising of {4, 7, 12, 17, 23, 26, 31, 33} may be treated as equivalent to alphabet E.

7.2. Keylogging Attacks

A keylogger is either a software or hardware based attack. It results in the keyboard entries of users getting extracted from the system and forwarded to the miscreant. Keylogging attacks are only possible when passwords or PIN numbers are entered through the mechanical keyboards. To obviate the possibility of these attacks, the virtual on-screen keyboards are widely used, particularly for the banking applications. In our proposed schemes, the PIN number is not entered through the keyboard. In both the variants, users just press the space bar and enter key on the keyboard. But an attacker will get no clue about the PIN numbers by retrieving these entries through keylogging.

7.3. Shoulder Surfing Attacks

7.3.1. Human Shoulder Surfing

In a human shoulder surfing attack, the attacker is strategically located in a close proximity to the user such that he could clearly view the user entries for the password or the PIN. In a simple PIN entry scheme made on an on-screen keyboard, the susceptibility of the PIN number to be detected by the observer is very high. But any scheme that increases the memory requirement of the observer diminishes the chances of a HSSA.

- Manual-Drag Scheme: Our scheme comprise of five different alphabets repeated twice in each row at random positions and a total of forty columns used. When the sliding occurs, the entire row moves either in the left or the right direction. If a HSSA has to succeed on our scheme, then the attacker should be capable of memorizing all the forty entries on the screen which is an unlikely possibility. Even if we assume that the observer attempts to note down the alphabet positions in a paper, our user study have shown that the users could complete their PIN entry and submit it in an average of 2 minutes time. That is insufficient time for the observer to copy everything onto a paper. So the proposed scheme is sufficiently harder to be attacked through HSSA.
- Auto-Sliding Scheme: The possibility of HSSA on this variant of our scheme is completely ruled out. Since the user does not align the alphabets on the screen as per his PIN entry and he just accomplishes that by pressing the space bar at an appropriate moment during the sliding, the attacker gets no idea on what could be the positions of the alphabets on each row. The only way a HSSA may chance in this scheme is if the attacker is able to observe the space bar pressing and the alphabet positions as on the screen at that moment. But it is impossible to achieve that since alphabets continue to slide at a considerable speed. So an observer who could notice the space bar entry would surely miss to capture the alphabet positions exactly at that moment.

7.3.2. HSSA Resistance Evaluation

To demonstrate the resistance of the scheme against HSSA, an evaluation with the help of participants who would act as potential attackers was conducted. We set up a hall equipped with Liquid Crystal Display (LCD) projector for our testing. We inducted 60 participants who are engineering graduate students of our university. After a detailed introduction to our proposed scheme, the participants were requested to keenly observe the PIN entry process and then to predict the PIN number which they consider as being entered by the user. The participants had enthusiastically taken part in this exercise. Out of the 60 participants, none mentioned correct PIN number. They opined that the time they got to observe the screen during the PIN entry was inadequate to note down all the screen entries. The result of the HSSA evaluation vindicates our claim about the stronger HSSA resistance of our scheme.

7.3.3. Recorded Shoulder Surfing Attack

This is a much more difficult attack to tackle than a simple cognitive attack by a human being. The attacker has the convenience of watching the recorded

video any number times and to play it in slow motion or pause and play. Developing schemes for providing full resistance to RSSAs would degrade their usability since they either require the usage of additional module or increase of the number of PIN digits [7]. So most of the proposed schemes either offer no protection against RSSA or they have employed some form of additional module, such as a user calculation [19] or a secondary channel [9] or some additional information to be remembered [13] to attain their resistance against RSSA.

- Manual-Drag Scheme: This scheme's implementation consist of a grid of 4 X 10 size, 5 alphabets with each alphabet occurring twice in each row at random positions. The PIN number is entered by arranging the same alphabet on all the four PIN digit positions. If an attacker acquires a recorded copy of the PIN entry then it would be similar to the screen shot of Figure 4. He would know that the PIN number is one of the combinations of the same alphabet in all the four rows. But he does not get a straight forward answer because there are five alphabets and each alphabet appears twice in each row. Given that there are 2 choices for an alphabet in one row, the number of possibilities for one alphabet will be:

$$2 \times 2 \times 2 \times 2 = 16$$

Since there are 5 alphabets used, the total number of possible combinations for five alphabets will be:

$$16 \times 5 = 80$$

So an attacker has to try for a maximum of 80 possibilities in order to detect the actual PIN number. 80 is a sufficiently large size for PIN numbers since the authentication systems would allow only few attempts for entering incorrect PIN numbers. They would lock the system after the permitted number of trails and the user may me required to reset the PIN number. Even if the attacker knows the alphabet chosen by the user, he still has 16 possible PIN numbers to try.

- Auto-Drag Scheme: According to [7], 100 % foolproof against RSSA cannot be achieved without severely compromising on the usability of the scheme. Our review results presented in Table 1 also corroborate this. To attain the maximum protection against RSSA, other proposed schemes have either used secondary channels or some form of computations by the users. Our auto-sliding scheme is probably the first scheme that neither requires any secondary channel nor any complex mathematical computation by the user. Our scheme only demands that the user concentrates on the animated sliding movement of alphabets on the screen and hit the space bar at a particular time. It is much easier than certain games in which the player has to hit a button at a precise second to shoot at a target by a gun. From the screen shot of the PIN entry given in Figure 5, it could be known that the attacker gains no useful information about the PIN from the recorded content. If the user is prudent and aware of the scheme's mechanism, he could mislead the attacker by deliberately choosing the timing to stop the sliding such that his actual PIN positions carry completely wrong alphabets.

8. User Study and Usability Analysis

An important goal that was set for the proposed scheme was to be highly usable. A scheme that is highly secured but is poorly usable will not be adopted for real world usage. People's convenience, ease, time and familiarity all play role in the usability of the scheme. We designed our scheme keeping in mind the usability requirements. The schemes have got the following characteristics in support of its usability:

- Colourful and attractive user interface using english alphabets.
- The entire PIN number is entered in a single round without taking much of user's time.
- User is not required to remember anything other than the actual PIN number to use the system. The present PIN entry has no connection with the previous and the future entries.
- The requirement of user intelligence for using the system is minimal.

In order to formally establish the usability of our proposed schemes, we decided to conduct a user study with participants who represent the real users.

8.1. Implementation of the Proposed Schemes

To conduct the user study, we have implemented a actual software of both the variants of our schemes using .net under visual studio 2010. The software begins with a PIN input of four digits which is used as the equivalent of the actual PIN number stored at the server. Then the PIN entry interface gets loaded. For the selection of the five alphabets with different background colours, we created image files of all the 26 alphabets that are coloured. The software would randomly pick five alphabets from the folder. Then the randomizing algorithm positions the five alphabets randomly on the forty columns of the grid. We added a condition into the algorithm to prevent the occurrence of same alphabets on consecutive columns of the same row. User aligns the alphabet as per the actual PIN number. When user press the submit button, the verifyPIN module gets invoked and it compares the actual PIN number with the PIN number extracted from the array. In case of a match, it displays the message that PIN was successfully verified, otherwise it displays that the PIN entry failed

8.2. Participants

Because the proposed scheme could be used by people

of all age groups and genders, we decided to accommodate participants representing the different age groups and genders. To represent the young users, we inducted 20 students of our university who were in the age group of 19 to 22. Out of the 20, 12 were male and 8 were female. For the representation of middle aged users, we requested for and obtained the consent from 10 staff members to participate in our study. They were in the age group of 33 to 48. All the ten were male. We scheduled for the user study of the students and the staff members on two separate days.

8.3. Evaluation Procedure

We introduced our proposed model to the participants and explained them the method of entering PIN entry using our schemes. We requested them to choose a four digit PIN number and to remember it till they complete their process. The participants took turns to complete their PIN entries using our computers. Every participant had to enter the same PIN number twice on each one of the implementations. We requested the participants to perform their PIN entry at their usual pace without hurry since we were keen to assess the entries as per their normal usage. The following section summarizes the results of our study.

8.4. Results

The objective of the user analysis is to demonstrate the usability of our proposed scheme. The best way to achieve this is by getting it evaluated by actual users. The different time taken by the participants is presented in Table 3. The following sections discuss the details of the tests for the two schemes separately.

- Manual-Drag Scheme: For the 20 participants who completed their PIN entry, the mean, min, max, and median in seconds were 26.5, 23.6, 30 and 26.2. As was expected, the time taken by the staff participants was higher than the younger participants. For the group 10 participants, the mean, min, max and median were 31, 27.3, 32.5 and 29.6. 27 of the participants made correct entries in the first attempt. Only 3 participants required a second entry (Figure 6).

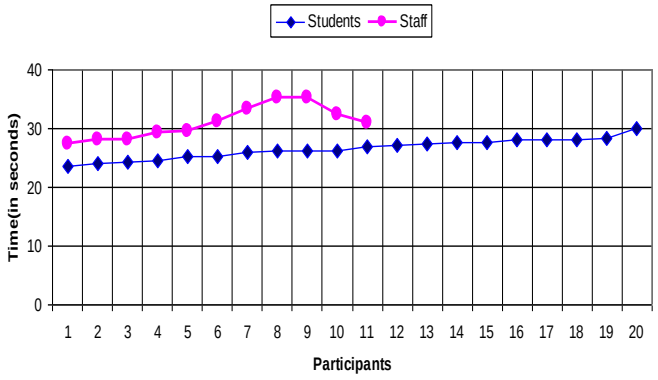

Figure 6. Time taken by participants under manual-sliding scheme.

- Auto-Drag Scheme: The mean, min, max and median time among the student participants were 29.45, 27.5 and 29.2 respectively. But for the staff participants they were 33.3, 29.8, 37.2 and 32.3. The difference between the student and staff participants was slightly elevated in the case of the auto-drag scheme. It is probably because of the high prevalence of the gaming activities of the students. This seems to have helped the students to handle the auto-drag scheme effortlessly. The number of wrong entries was also in the same pattern. There were 5 wrong entries and all the four were by staff participants (Figure 7).

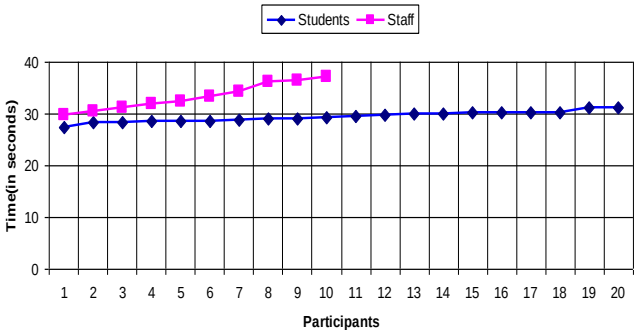

Figure 7. Time taken by participants under auto-sliding scheme.

8.4.1. User Feedback

At the ends of the hands on trial participants were asked two questions. The first question was whether they would prefer to use our scheme in case if it is provided as an option in their e-banking portal and the second question was between the two variants of the schemes, which one they consider as highly secured for their PIN number. Around 89 % of the participants said “yes” for the first question and that conveyed their willingness to adopt the scheme into regular practice. For the second question, all the participants were unanimous in choosing the Auto-drag variant.

Table 3. Results of the User Study. t(studs) and t(staff) are the time(in seconds) for the PIN entry by the students and staff members using the two schemes.

Auto Sliding		**Manual Sliding**	
t(Studs)	**t(Staff)**	**t(Studs)**	**t(Staff)**
27.5	29.8	23.6	27.3
28.3	30.5	24.1	28
28.4	31.2	24.3	28.1
28.5	32	24.5	29.3
28.6	32.3	25.3	29.6
28.6	33.4	25.3	31.3
28.9	34.2	26	33.4
29	36.3	26.1	35.2
29.1	36.4	26.2	35.3
29.2	37.2	26.2	32.5
29.5		26.9	
29.8		27.1	
30		27.3	
30		27.6	
30.2		27.6	
30.2		28.1	
30.3		28.2	
30.3		28.2	
31.3		28.3	
31.3		30	
29.45	33.33	26.545	31

9. Discussion

Through the user study we were able to ascertain the overall usability of our schemes. It was verified that the time taken for completing the PIN entry is within the acceptable limits. Moreover it will improve on regular usage. Time taken is also important from the security perspective. If the user spends more time for entering the PIN, it would give ample time to the shoulder surfing attackers to apprehend the PIN numbers. Not many users erred in their PIN entry. Only few required second attempts. Finally, the user feedbacks proved with certainty that our schemes are secured and usable in the perception of the users. Among the two variations, the manual sliding scheme offers stronger protection against HSSA and reasonable protection against RSSA. The auto sliding variant offers superior resistance against RSSA. But it requires some user practice to avoid wrong entries.

10. Conclusions

In this paper, we proposed and analyzed the design of new PIN entry scheme to secure the PIN numbers against keylogging and shoulder surfing attacks. Two variants of the schemes based on manual and auto dragging were presented. The user studies carried out on the implementations of the schemes have demonstrated their high usability and security. But a user evaluation based on a larger participant group will have to be done to fully ascertain its adoptability before actual usage. Though the proposed schemes are designed and tested on computer usage, they are yet to be tested for their usability on touch screen based mobile phones. There is also scope to extend the proposed approaches for full-fledged password entries.

References

[1] De Luca A., Frauendienst B., Boring S., and Hussmann H., "My phone is my Keypad:Privacy-Enhanced PIN-Entry on Public Terminals," *in Proceedings of the 21st Annual Conference of the Australian Computer-Human Interaction Special Interest Group: Design: Open 24/7*, Melbourne, pp. 401-404, 2009.

[2] De Luca A., Hertzschuch K., and Hussmann H., "ColorPIN: Securing PIN Entry Through Indirect Input," *in Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Atlanta, pp. 1103-1106, 2010.

[3] Kim C. and Lee M., "Secure and User Friendly Pin Entry Method," *in Proceedings of International Conference on Consumer Electronics*, Las Vegas, pp. 203-204, 2010.

[4] Kölsch M. and Turk M., Keyboards Without Keyboards: A Survey of Virtual Keyboards, Technical Report, 2002.

[5] Kumar M., Garfinkel T., Boneh D., and Winograd T., "Reducing Shoulder-Surfing by Using Gaze-Based Password Entry," *in Proceedings of the 3rd Symposium on Usable Privacy and Security*, Pittsburgh, pp. 13-19, 2007.

[6] Kwon T. and Na S., "SwitchPIN: Securing Smartphone PIN Entry with Switchable Keypads," *in Proceedings of the IEEE International Conference on Consumer Electronics*, Las Vegas, pp. 23-24, 2014.

[7] Lee M., "Security Notions and Advanced Method for Human Shoulder-Surfing Resistant PIN-Entry," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 4, pp. 695-708, 2014.

[8] Malek B., Orozco M., and El Saddik A., "Novel Shoulder-Surfing Resistant Haptic-based Graphical Password," *in Proceedings of EuroHaptics Society*, vol. 6, pp. 179-184, 2006.

[9] Mohaisen A., Nyang D., and Kang J., "Keylogging-Resistant Visual Authentication Protocols," *IEEE Transactions on Mobile Computing*, vol. 13, no. 11, pp. 2566-2579, 2014.

[10] Perković T., Čagalj M., and Saxena N., "Shoulder-Surfing Safe Login in a Partially Observable Attacker Model," *in Proceedings of Financial Cryptography and Data Security*, Canary Islands, pp. 351-358, 2010.

[11] Por L., "Frequency of Occurrence Analysis Attack and its Countermeasure," *The International Arab Journal of Information Technology*, vol. 10, no. 2, pp. 189-197. 2013.

[12] Raddum H., Nestås L., and Hole K., "Security Analysis of Mobile Phones used as OTP Generators," *in Proceedings of Information Security Theory and Practices. Security and Privacy of Pervasive Systems and Smart Devices*, Passau, pp. 324-331, 2010.

[13] Rajarajan S., Maheswari K., Hemapriya R., and Sriharilakshmi S., "Shoulder Surfing Resistant Virtual Keyboard for Internet Banking," *World Applied Sciences Journal*, vol. 31, no. 7, pp. 1297-1304, 2014.

[14] Roth V., Richter K., and Freidinger R., "A PIN-Entry Method Resilient Against Shoulder Surfing," *in Proceedings of the 11th ACM Conference on Computer and communications Security*, Washington, pp. 236-245, 2004.

[15] Roth V. and Richter K., "How to Fend off Shoulder Surfing," *Journal of Banking and Finance*, vol. 30, no. 6, pp. 1727-1751, 2006.

[16] Shi P., Zhu B., and Youssef A., "A PIN entry Scheme Resistant to Recording-based Shoulder-Surfing," *in Proceedings of Third International*

Conference on Emerging Security Information, Athens, pp. 237-241, 2009.

[17] Shi P., Zhu B., and Youssef A., "A Rotary Pin Entry Scheme Resilient to Shoulder-Surfing," *International Conference for Internet Technology and Secured Transactions*, London, pp. 1-7, 2009.

[18] Tari F., Ozok A., and Holden S., "A Comparison of Perceived and Real Shoulder-Surfing Risks Between Alphanumeric and Graphical Passwords," *in Proceedings of the Second Symposium on Usable Privacy and Security*, Pittsburgh, pp. 56-66, 2006.

[19] Wilfong, G., "Method and Apparatus for Secure PIN Entry," U.S. Patent 5,940,511, 1999.

[20] Wu T., Lee M., Lin H., and Wang C., "Shoulder-Surfing-proof Graphical Password Authentication Scheme," *International Journal of Information Security*, vol. 13, no. 3, pp. 245-254, 2014.

Rajarajan Srinivasan received M.Tech in CSE from SASTRA University. He is presently pursuing part time PhD at SASTRA University, India. He is an Assistant Professor at the same university. His area of research interest includes computer security, e-banking and graphical passwords.