An Improved Methodology towards Providing Immunity against Weak Shoulder Surfing Attack

Nilesh Chakraborty and Samrat Mondal

Computer Science and Engineering Department
Indian Institute of Technology Patna
Patna-800013, Bihar, India
{nilesh.pcs13,samrat}@iitp.ac.in

Abstract. In a conventional password based authentication system, an adversary can obtain login credentials by performing shoulder surfing. When such attacks are performed by human users with limited cognitive skills and without any recording device then it is referred as weak shoulder surfing attack. Existing methodologies that avoid such weak shoulder surfing attack, comprise of many rounds which may be the cause of fatigue to the general users. In this paper we have proposed a methodology known as Multi Color (MC) method which reduces the number of rounds in a session to half of previously proposed methodologies. Then using the predictive human performance modeling tool we have shown that proposed MC method is immune against weak shoulder surfing attack and also it improves the existing security level.

Keywords: Authentication, Human shoulder surfer, Human performance modeling tool, Session password.

1 Introduction

Authentication is an important component of computer security. Among the different authentication schemes, password based authentication is one of the popular schemes for its efficacy and ease of use. However, the scheme fails to give security against *observation attack* while entering password in a public place (like ATM counter). In this attack, the attacker observes the credentials entered by the user and later may use it illegally for login purpose. This attack is also referred as *shoulder surfing attack*.

Now depending upon the nature of shoulder surfing attack and the types of equipment adversary uses, the attack is divided into two categories – *a) Strong Shoulder Surfing Attack*, where an adversary uses some recording device (like conceal camera) to record a user login session [12] [11] and, *b) Weak Shoulder Surfing Attack*, in which attacker relies on limited cognitive capabilities of human users and does not use any recording devices, though s/he might use pencil and paper to note down session information [18]. Now in general strong shoulder surfing resilient schemes such as [29] [12], [11] require more computational skills from users' end than that of weak shoulder surfing resilient schemes [26], [18].

A. Prakash and R. Shyamasundar (Eds.): ICISS 2014, LNCS 8880, pp. 298–317, 2014.

As system used in public domain (like ATM machine) is handled by all type of users so computational complexity during login is required to be less and thus, weak shoulder surfing resilient schemes become effective over strong shoulder surfing resilient one. In addition of giving protection against observation attack, shoulder surfing resilient schemes provide security against attacks such as- key-logger based attack [14], spreading chemicals on keypad to obtain the keystrokes [7], etc.

To avoid weak shoulder surfing attack, Roth et al. [26] proposed a scheme (we call it as Black-White or, BW method) in 2004 which was considered to be secure against weak shoulder surfers till 2012 [28]. Later Kwon et al. [18] proved that, human shoulder surfers − without equipped with any gadgets like recording device, can break the security of BW method by performing following three operations :

1. Covert attention [21] [18]
2. Perceptual grouping [19]
3. Motor operation [2]

In their work [18] authors proposed an improved methodology (termed as Four Color or FC method in this paper) which overcomes the above three step operations attack, performed by skilled human shoulder surfers. In literature, shoulder surfers capable of performing *Covert Attention*, *Perceptual Grouping* and *Motor Operation* are denoted as *CPM shoulder surfer*. The details of these operations are explained in Section 2. Though FC method is secured against weak shoulder surfing attack but the major problem with this scheme is that a huge number of rounds is required for login. In fact, both BW and FC methods require 16 rounds in a session during login for a PIN of length 4. Thus the user fatigue level becomes high [26] as user needs to face more number of rounds in a session. This may cause human mind inattentive and increase error rate during login [22]. Motivated by this issue we have made two major contributions in this paper.

Contribution 1: We have proposed a new model known as Multi Color or MC model in which user faces 8 rounds for a four digit PIN. Security analysis shows that MC method provides better security against random key selection attack (see Section 4) than of those BW and FC methods.

Contribution 2:We have performed security analysis of our method against CPM shoulder surfers, by using human performance modeling tool as shown by Kwon et al. [18]. We introduce the concept of *hardness factor*, higher value of which shows less vulnerability of a method against weak shoulder surfing attack. We also derive that MC method has higher value of *hardness factor* compared to BW and FC method.

The rest of the paper is organized as follows − in Section 2 we have given a brief overview of the existing work and also discussed some preliminary concepts required to understand our approach. The proposed approach is presented in Section 3. We have performed security analysis in Section 4. Usability analysis of our work is illustrated in Section 5. We conclude and give future direction of our work in Section 6.

2 Overview of Existing Work and Some Preliminary Concepts

Many methods [27], [4], [12], [11] have been proposed since international standard for PIN management, ISO 9564 mandated the fact that PIN entry device should be designed in such a way which can give protection against shoulder surfing attack [1]. Some methodologies such as [12], [11] were developed to resist partially observable shoulder surfing attack. Methods like [4], [29] were proposed to tackle fully observable shoulder surfing attack against strong adversary. However most of these schemes require a lot of computation from the user end.

Schemes proposed to handle weak adversary is relatively easy to use. In 2004 Roth et al. proposed a scheme termed as BW method [26] which is resilient against shoulder surfing attack performed with limited cognitive skill. In this method, the user interface consists of a numeric keypad on which, half of the numeric buttons on the keypad are colored as black and rest are colored as white as shown in Fig 1.

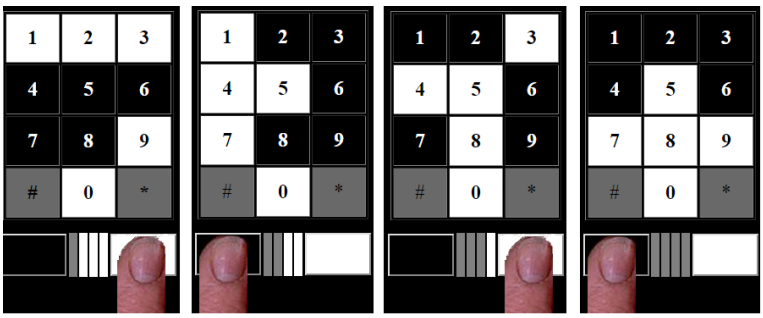

Fig. 1. Above figure shows user response for PIN digit 3. Each time keypad gets partitioned into half of the keys as black and the rest as white. User needs to identify the correct partition in which his/her PIN digit belongs.

The color of the numeric buttons varies in each round. User needs to identify the proper color that appears on his chosen PIN digit by pressing either black or white color button. User chooses a four digit PIN from a set $Q = \{0, 1, 2, ..., 9\}$. User needs to face $r = \lceil log_2|Q| \rceil$ rounds for each PIN digit. So for a l digits (here $l = 4$) long PIN user will face $l \times r$ rounds.

Limitations of BW Method: To explain the limitations of BW method, some prerequisite knowledge is required about the vision and information processing capabilities of human. This will help readers to understand the activities of CPM shoulder surfers and vulnerability of BW method more clearly.

Foveal Vision: It refers to normal vision capability of human while fixing his/her eye at a particular object [23]. For example, in the word $n+1$, by looking

at 'n' a person can understand the whole word. This is because while looking at 'n', character '+' and '1' also come into normal vision angle. It has been observed that people having normal (or correct to normal) vision, can notice objects within 2° of visual angle, by fixing eye at a particular position. 1° visual angle is about 3 normal text from the point of eye fixation.

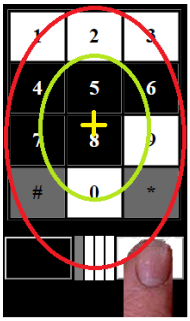

Fig. 2. Above figure shows a foveal and parafoveal vision ranges of human eyes by fixation of eye to a particular point (shown by yellow color +). Inner circle shows the foveal vision range and outer circle shows the parafoveal vision range.

Parafoveal Vision: It signifies the vision region which is hard to see (if not impossible) by fixing eye at a particular point. It starts from the end point of foveal vision region and surrounds within an angle of 5° from the eye fixation point [23]. Readers can assume that, this region starts after 4 to 5 normal text and ends after 8 to 9 texts from there. To gain information from this vision region, human needs saccadic (rapid) eye movements (except skillful video game player). Video game players normally have improved vision capabilities than of normal people [13] and can obtain information from extra foveal vision region even without saccadic movement of eyes. Both foveal and parafoveal vision ranges have been shown in Fig.2.

Covert Attention: Covert attention corresponds to attention not associated with eye movements. Significance of covert attention is, human can store a fair amount of information in visual short term memory (VSTM) [20] from foveal vision range by performing covert attention. By this operation, video game players can obtain the information from both the visual angles (2° and 5°) because of their improved vision skill [5] [13]. Extracted information from the range of foveal vision, helps adversary to perform perceptual grouping which is discussed next.

Perceptual Grouping: Perceptual grouping [19] implies grouping of objects and it depends upon their proximity, similarity, continuation, closure and symmetry. In BW method adversary can group objects (colored numeric buttons) based upon their color from the fovel vision range.

Motor Operation: Motor operation [2] requires a co-ordination between central nervous system and the musculoskeletal (muscular and skeletal) system. Human processes the grouping information by performing covert attention and perceptual grouping which requires effort of human mind. Now if the adversary wants to write down some gained information, his/her hand (comes under musculoskeletal system) must be engaged and thus the co-ordination between hand and mind is required for surreptitious handwriting, without moving the eyes.

Attack on BW Method by CPM Shoulder Surfers: Time required to enter a digit in each round by user is called response time. If the response time of the user allows the attacker to perform the necessary operation to obtain the PIN digit then attacker will proceed successfully. By using CPM-GOMS tool Kwon et al. [18] in their work showed that CPM shoulder surfers can proceed successfully to break the security of BW method. In Fig. 3 we have shown a pictorial presentation of attack scenario on BW method.

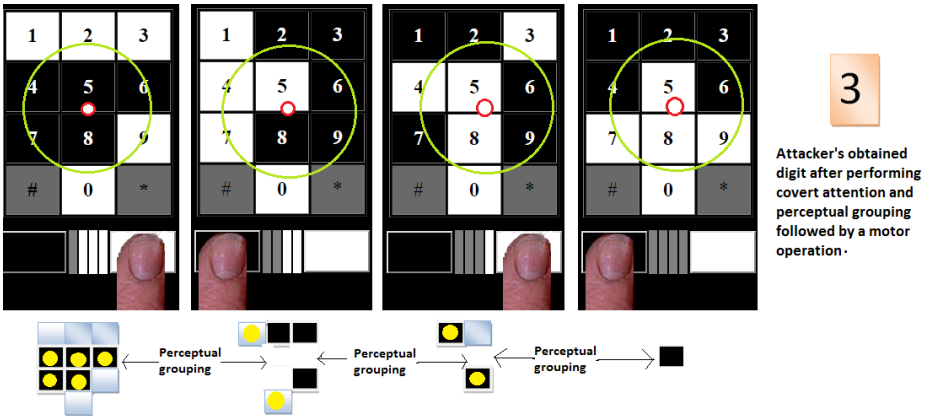

Fig. 3. Above figure shows a foveal vision angle to obtain the perceptual grouping. Parafoveal vision helps attacker to obtain the color chosen by user see (Fig. 2). Based upon the user response attacker discards the group (shown by yellow dot in the picture) from (visual short term memory) VSTM [8] [20] not falling into user color response. Finally attacker obtains a group consisting of single object due to logarithmic decrease of group cardinality in BW method.

In the first round of response of a digit, attacker first groups the black and white objects together. One thing is needed to mention here, while attacker groups those black and white numeric buttons depending on the colors, attacker overlooks the digits on the color buttons. After perceptual grouping attacker sees the user response in the first round corresponding to the first digit and depending upon that attacker discards one of the group from VSTM. For example, if user presses white color button then attacker discards the group of black

objects. Reason behind this is, user PIN has appeared on the button belonging to white color group, so only that information is required by the attacker. In the immediate next round attacker keeps his/her eyes on those part of the keyboard interface which forms the previous color group and has been stored in his/her VSTM. Now in this round attacker finds that color black has appeared on some portion of the group stored in VSTM and color white has appeared on the rest. Now again depending upon user response attacker discards some of the objects from VSTM and stores a smaller perceptual group in VSTM at the end of the second round. This will be continued through out the four rounds corresponding a PIN digit of user. In every round, the cardinality of the perceptual group will be decreased and always it will converge to 1 on or before four rounds. After identifying a single object by the end of fourth round attacker will observe the digit written on it. Then s/he performs hand motor operation to write down the digit.

FC Method: In 2013 Kwon et al. [18] proposed a scheme referred as FC Method in which they have used four colors for coloring the numeric keypad. Each numeric button has been divided into two partitions. So there has been a total of 20 partitions (10 numeric buttons each having 2 partitions) which are filled with those 4 colors. The basic principal behind coloring the button are *(i)* each color will appear in exactly 20/4 (i.e. 5) partitions. *(ii)* same color will not appear on a button twice. So in each round user will find that his/her PIN digit posses two colors. User can choose any one of those two colors as his response and will press the color button of his/her chosen color. For giving response there exist four color (which are used to color the numeric buttons) buttons on user interface.

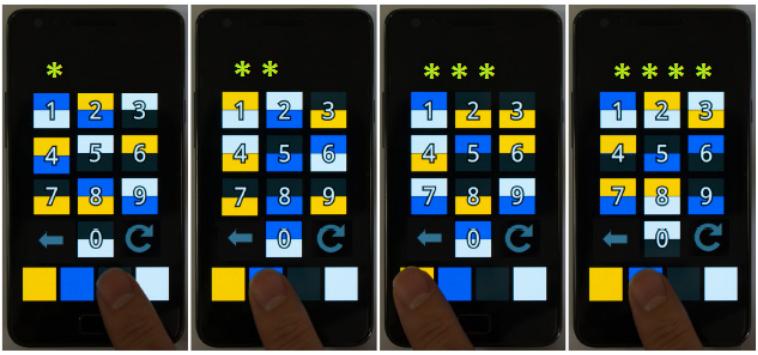

Fig. 4. Above figure shows user response for PIN digit 6. Each time user keypad gets partitioned using four colors. User needs to identify one of the correct color of his/her corresponding PIN digit.

Power of FC Method: There are evidences that human can recognize a visual object in quick time, occurring within $100 - 200$ milliseconds of stimulus presentation and can bring that thing within consciousness in another 100 milliseconds of time [25]. So objects posses similar properties can be perceptually grouped within at most 300 milliseconds. As in BW method attacker needs to perceptually group two different objects (black and white) so it takes 600 milliseconds to perform perceptual grouping. In [18] Kwon et al. showed that login time complexity of BW method in each round, would allow CPM shoulder surfers to get that required time for perceptual grouping operation. Thus security of the BW method was compromised.

In FC method perceptual grouping to identify objects of four colors takes (4×300 or) 1200 milliseconds [25] in each round. But time complexity of each round does not allow the CPM shoulder surfers to get that required time for perceptual grouping, in fact in [18] authors showed that CPM shoulder surfers only get 700 milliseconds for perceptual grouping which is much less that the required time limit and thus reduces the chance of attack.

3 Proposed Multi Color Methodology

The main problem of FC method is that it takes 16 rounds for a four digit PIN. Thus the login process becomes lengthy and as a result is more error prone. In the proposed approach our aim is to reduce the login rounds without compromising with the security. In this section first we will discuss the basic feature and the login principal by using our proposed Multi Color (MC) methodology. Next we will describe how each digit of user PIN gets identified by the system uniquely. In MC method we have used a set *COLORS* consisting of five different colors – here *COLORS* = { *Red, Green, Pink, Yellow, Sky* }. For color blind people the same set can be replaced by *MARKS* = { *Black, White, Dot, Vertical strip, Horizontal strip* }. User PIN consists of 4 digits denoted as *d1,d2,d3,d4*.

3.1 Basic Feature of MC Method

Each numeric button in MC method is subdivided into three partitions namely *Up*, *Middle* and *Down*. So for ten numeric buttons from 0 to 9 there are 30 (3×10) partitions over which the five colors will be distributed. So each color will appear exactly in (30/5 or) 6 places. Now there will be a coloring constraint, by following which those colors will be distributed. The coloring constraint is described as follows:

Each color will appear in six partitions in six different numeric buttons. Among those six partitions, each partition will appear exactly twice.

In Fig. 5 we have shown the distribution of five colors in MC method. Each numeric button holds three different colors. Each color is placed on six different numeric buttons holding each partition exactly twice. Five color buttons shown bellow in Fig. 5 are used for giving response by user. To design the login interface

Fig. 5. A prototype model of MC method

Table 1. Useful notations used in algorithms

Notations	Descriptions
δ(X)	Randomly permute elements of set X
B_k	Numeric button associated with digit k
B_k(p)	p^{th} partition in B_k
cfPosition(S)	Returns the first element from the set S
getEmpty(B_k)	Returns the partitions in B_k, not filled by any color
equCheck(A,B)	Checks whether set A, B are equivalent or not
colr(B_k(p))	Returns color at p^{th} partition of B_k
getValue(p)	Returns the value associated with a partition p
getColorpos(C,B_k)	Returns the partition where Color C placed in B_k
cardinality(S)	Returns the cardinality of set S
exchangeBackColor(B_X,B_S)	Background color of numeric buttons B_X and B_S exchanged
rand(S)	Choose an element from set S randomly
view(Keypad)	Shows the colored numeric buttons on user interface

we have used Algorithm 1. Readers can refer to Table 1 to understand the meaning of the notations used in Algorithm 1 (and also in Algorithm 2 in Section 3.2) in Table 1.

Algorithm 1 takes the permuted color set *COLORS* as its one of the inputs. The other input N is a set of integers from $0-9$. At each iteration, for each color C $\in$ *COLORS* set *FILLED* holds those partitions where C is already placed twice and can not be placed any more. Set S used in Algorithm 1 stores values of k for which numeric button B_k has already been encountered for a particular color. Variable k, used in the algorithm assures that same numeric button B_k for a particular color C, does not get selected more than once.

Algorithm 1. Color.NumericButtons()

```
Input: This algorithm will take set COLORS and set N = {0, 1, ..., 9} as input.
Output: This algorithm colors ten numeric buttons by following coloring constraint.
COLORS ← δ(COLORS)                                   /* randomly permute the color set */
foreach (C ∈ COLORS) do
    Initialize: up ← 0; mid ← 0; down ← 0; /* variable up, mid and down are associated
    with partition Up, Middle and Down respectively */
    FILLED ← empty ; S ← empty;
    while (1) do
        k ← rand(N-S);                    /* selects a random number from the set N-S */
        P ← getEmpty(B_k); /* holds those partitions in B_k not filled by any color */
        if (equCheck(P,FILLED) = false AND P ≠ empty) then
            pos ← cfPosition(δ(P − (P∩FILLED)));
            if (pos ≠ empty) then                    /* condition false if P ⊆ FILLED */
                B_k(pos) ← C;
                getValue(pos)++;                  /* increases value of up, mid, down */
            end
            if (up=2) then
                FILLED ← Up;
            end
            if (mid=2) then
                FILLED ← Middle;
            end
            if (down=2) then
                FILLED ← Down;
            end
        end
        add digit k to set S
        if (cardinality(S) = 10) then     /* if color C can not be placed in any numeric
        button by maintaining the coloring constraint */
            for (t = 0 to 9) do
                if (C ∉ B_t) then                          /* if B_t not posses color C */
                    for (r = 0 to 9) do
                        pos ← getEmpty(B_r); /* pos initially holds partitions in B_r
                        not filled by any color */
                        pos ← cfPosition(δ(pos−(pos∩FILLED)));
                        if (pos ≠ empty) then
                            if (colr(B_t(pos)) ∉ B_r ) then
                                B_r(pos) ← colr(B_t(pos)); B_t(pos) ← C; /* swap colors
                                between the partitions */
                                getValue(pos) ++; break;
                            end
                        end
                    end
                end
                if (up = 2 AND mid = 2 AND down = 2) then
                    break;
                end
            end
        end
        if (up = 2 AND mid = 2 AND down = 2) then
            break;
        end
    end
end
return (ColorNumericKeypad);
```

3.2 Login Procedure and Evaluation of User Response

Using our proposed methodology user will give response twice for each of his/her PIN digit. As user PIN is 4 digit long so user will face $2 \times 4 = 8$ rounds in each session. In the first round user selects one color out of three colors from the numeric button corresponding to the first digit of the PIN and presses the corresponding color button. While choosing the color in the first round, user needs to remember the chosen partition. For the subsequent responses in that session user will look for the numeric button corresponding to his/her PIN digit and will select the color from the same partition.

Fig. 6 and Fig. 7 show user response for first two digits of PIN "d1 d2 d3 d4"(where d1 = 2, d2 = 3, d3 = 4, d4 = 1 taken as an example here). User gives his/her response in the first round by choosing a color from the middle of the numeric button corresponding to his/her first PIN digit 2 and thus user will always select a color from the middle of his/her corresponding PIN digit in that session. User enters his /her response for the first PIN digit in the first and second round, then for the second PIN digit in third and fourth round and so on. With the notion of the above discussion we define session partition next.

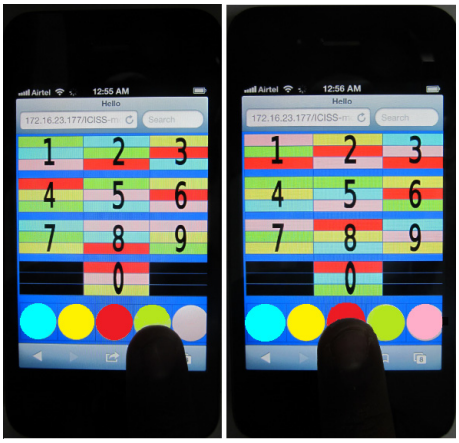

Fig. 6. Above figure shows first and second round responses for digit 2 with *session partition* selected as middle

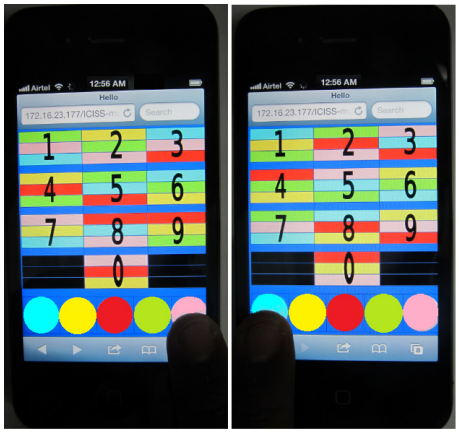

Fig. 7. Above figure shows third and fourth round responses for digit 3 while *session partition* remains middle

Definition 1 ***Session partition:*** *It represents an arbitrary partition SP* $\in$ *{Up, Middle, Down} on the numeric button corresponding to the first PIN digit of the user from where user chooses the first color for giving response in first round. For giving rest of the responses, user will choose the color from same session partition SP corresponding to the numeric button of the PIN digit.*

Validation of User Response: Our scheme ensures that valid responses of genuine user will uniquely be identified. This means by guessing a different PIN an attacker will never be able to access the valid user's account. Total 8 rounds complete the MC method and user has to response in two consecutive rounds for a single PIN digit. So by the end of each even round (2,4,6,8) every single PIN digit of user should uniquely get identified by the system.

To achieve this we have used the following strategy. After giving the first color response by the user, system will track the session partition (for that entire session) with the help of user color response and first PIN digit of the user. Next in each odd (1,3,5,7) round system will record the color "C" appeared on session partition corresponding to the user PIN digit d1,d2,d3,d4 respectively. Then system will look for all other numeric buttons (say "tracked buttons") along with partition (say "tracked partition") where color "C" has appeared. In each even round (2,4,6,8) system will ensure that color "$\overline{C}$", which has appeared on the session partition on the numeric button corresponding to the user PIN digit, will never appear in those "tracked buttons", on the "tracked partitions". Though "$\overline{C}$" may appear in those "tracked buttons" on different partitions. One thing needed to mentioned here that "$\overline{C}$" may same as "C". If user fails to choose the session partition properly then system will return numeric buttons with arbitrary color combination by maintaining coloring constraint in each round and will block the user at the end of the session.

For instance in Fig.6 user has responded with color button "Green" in the first round. System finds that user has identified *session partition* as Middle by identifying the color "Green" on the user PIN digit 2. Next system locates "Green" color in all other "tracked buttons" along with partitions. In the immediate even round (second round) system allocates color "Red" in the session partition Middle, of numeric button 2. System ensures that "Red" color has not appeared on the "tracked partitions" of those "tracked buttons".

Significance of this is, if an attacker guesses a PIN digit wrongly, say 7 and identified session partition as middle in the first round, then also his response will match with the valid response of the user (see Fig. 6). This is because, same color has appeared in both the partitions (middle of numeric button 7 and 2). Now as system ensures that, color appear in middle of numeric button 2 will not appear in the previously tracked partitions (including middle of numeric button 7) in immediate even round (here second round), so attacker finds a different color (green) in middle of numeric button 7 which is not a valid color response, as color "Red" has appeared in middle of numeric button 2. Thus if an attacker proceedes successfully in any of those odd rounds (by guessing a wrong PIN digit or wrong session partition), our system ensures that, in immediate even round followed by an odd round, attacker will give a wrong response if either of PIN digit or session partition is wrong.

We have used Algorithm 2 to evaluate user response. Two array data structures "but" (abbreviation of button) and "poss" (abbreviation of position) are used in that algorithm which will hold the information about "tracked buttons" and the partition information respectively. In Fig.6 user responds by pressing

color button "Green" in the first round. So array "but" and "poss" will hold the information about color "Green" that has appeared on the other numeric buttons. Table 2 shows the content of both the arrays for the above described situation. We have presented Algorithm 2 which will evaluate the user response. *The user will only get authenticated if array Resp in Algorithm 2 holds value* 1 *at all it's indices.* In Algorithm 2 "response" indicates the color entered by user as his/her response.

Table 2. Information stored in "but" and "poss"

index	but	poss
0	1	up
1	4	down
2	5	up
3	7	middle
4	9	down

4 Security Analysis

On discussing the security analysis of MC method first we will show the attack scenario by CPM shoulder surfers against skilled user login. Skilled users [18] are those who can minimize the login duration by suppressing rapid eye movement. We have used CPM-GOMS tool to perform security analysis. To prove the validity of the theoretical analysis we have performed an experimental analysis in support. Both the results show that MC method is more secure than BW and FC method.

Modeling the Security and Usability Trade-Off Using CPM-GOMS Tool for MC Method: Though it is quite feasible for CPM shoulder surfers to perform shoulder surfing attack on BW method but in FC method [18] authors have shown using CPM-GOMS tool that the same attack is infeasible. In our work we have used the same tool to show that our method is even slightly better than FC method. One thing can be noticed that, while performing the experimental analysis for user login and attacker activity we have tried our best to keep both user set and adversary set as mentioned in [18] in terms of their background and ability to perform. This helps us to compare better with the previous technologies.

The reason behind modeling execution time using CPM-GOMS [16] [15] (stands for cognitive perceptual motor and goals, operators, methods, and selection rules) is, it can model overlapping actions by interleaving cognitive, perceptual and motor operators and thus can predict the skilled behavior. Next we will introduce different functionality of CPM-GOMS.

Algorithm 2. Evaluation of user response

Input: This algorithm takes color keypad as input generated by Algorithm 1. **Output:** This algorithm will check user response in each round.

```
for (r=1 to 8) do
    Keypad ← Color.NumericButtons();   /* Keypad holds colored keypad returned by Algorithm 1 */
    if (r = 1) then
        view(Keypad); flag ← 0;
        if (response ∈ B_d1) then
            SP ← getColorpos(response,B_d1);                    /* sets session partition */
            flag ← 1; Resp[r] ← 1; k←0;
            for (i=0 to 9) do
                if (response ∈ B_d⌈r/2⌉ and i ≠ d⌈r/2⌉) then
                    but[k] ← i; poss[k] ← getColorpos(response,B_d⌈r/2⌉); k++;
                end
            end
        else
            SP ← null; Resp[r] ← 0;
        end
    else
        if (SP ≠ null) then                                     /* If SP correctly identified */
            if (r.Isodd() = true) then
                flag ← 1; view(Keypad);
                if (SP = getColorpos(response,B_d⌈r/2⌉)) then
                    Resp[r] = 1; k←0;
                    for (i=0 to 9) do
                        if (response ∈ B_d⌈r/2⌉ and i ≠ d⌈r/2⌉) then
                            but[k] ← i;
                            poss[k] ← getColorpos(response,B_d⌈r/2⌉); k++;
                        end
                    end
                else
                    flag ← 0; Resp[r] ← 0;
                end
            else
                if (flag = 1) then                              /* If SP correctly identified */
                    pColor ← colr(B_d⌈r/2⌉(SP));                /* holds valid color response */
                    for (t=0 to 4) do
                        X ← but[t]; Y ← poss[t];
                        if (colr(B_X (Y)) = pColor) then
                            for (S=0 to 9) do
                                if (poss[t] ≠ getColorpos(pColor, B_S)) then
                                    exchangeBackColor(B_X ,B_S); break;
                                end
                            end
                        end
                    end
                    view(Keypad);
                    if (response = pColor) then
                        Resp[r] = 1;
                    else
                        Resp[r] = 0;
                    end
                else
                    view(Keypad); Resp[r] = 0;
                end
            end
        else
            view(Keypad); Resp[r] = 0;
        end
    end
end
```

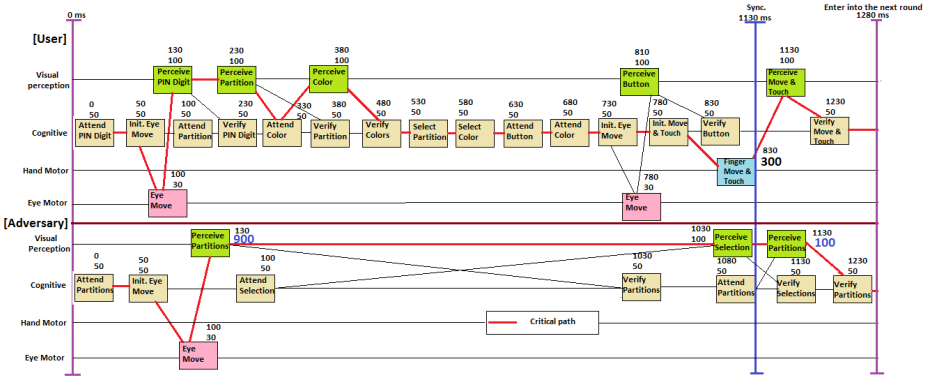

Fig. 8. Modeling and synchronization of MC method. (Each round takes 1280 ms. All rounds finish in 10.24 sec.) Skilled user is modeled.

Descriptive Operators and Functionalists of CPM-GOMS: Every task has been represented by a box with a duration in milliseconds (ms). According to the architecture of production system, cycle time of each cognitive operator is 50 ms [3] which is considered here. The cycle time to visually understand the presence or absence of an object is taken as 100 ms [9] but it may vary with the complexity of visual perception. The eye motor operation is set to 30 ms [17] which follows the conventional eye movement time in all CPM-GOMS models proposed after 1992 [10]. Time required for hand motor is reported as 300 ms in [18] and it has been evaluated empirically in their work. In our observation we also find the same. The reason behind this, in [6] author shows that "touch on screen" may take around 450 ms, though it may vary depending upon the screen distance and width. But estimated 450 ms time includes visual perception (requires 100 ms) of button and cognitive operation "initiation of move and touch" (requires 50 ms). So time required to perform hand motor is 300 ms, which is justifiable. The synchronization point is set after user presses a color button as his/her response.

Basic Idea Behind Overcoming the Attack: The basic idea behind overcoming the attack performed by CPM shoulder surfers is to increase the time required to perform the attack in such a manner so that it exceeds the user login time. If attacker does not get the required time to process the information then s/he will definitely fail. There are enough evidences that human can recognize objects within a time range of $100 - 200$ ms and takes another 100 ms to bring this information into awareness [25], thus perceptual grouping of same type of objects take $100 + 200 = 300$ ms. So in BW method adversary requires around 600 ms for perceptual grouping (300 ms each for recognizing group of black object and group of white object). Now in MC method as we have used five overlapping colors so total time required for perceptual grouping is $300 \times 5 = 1500$ ms. We modeled skilled user login time (see Fig. 8) by CPM-GOMS, which

shows that by suppressing saccadic eye movement user can give response in 1280 ms time in each round (which is very close to the actual login time by skilled user 1275 ms in each round discussed in Section 5). So attacker needs to accomplish the attack within 1280 ms. But human performance modeling tool shows due to other activities like, *Attend Partition, Initiate (Init.) Eye Move and so on* s/he only gets at most 900 ms (see Fig. 8) time to perform perceptual grouping which is never been enough to perform the attack. Attacker needs 600 ms more to perform the attack. Thus like FC method [18] and unlike BW method [26] CPM shoulder surfers fail in MC method to perform the attack. With the notion of above discussion we will define "hardness factor" next.

Definition 2 ***Hardness factor:*** *It is the ratio of actual time needed by CPM shoulder surfers to get the PIN digit and skilled user login time. Higher value of this shows less vulnerability of a methodology against the shoulder surfing attack performed by CPM shoulder surfers.*

In our proposed method user login time (or the time CPM shoulder surfers gets) is 1280 ms. But to perform the attack, it requires $1280 + 600 = 1880$ ms. So hardness factor becomes 1880/1280 or, 1.468.

Experimental Analysis of Shoulder Surfing Attack: To see whether the theoretical acceptance of the attack model is valid in reality or not we have selected 15 participants (12 male and 3 female) as attacker having average age of approx 24 years and (correct-to) normal eyesight. They all were right handed. As suggested in [18], we have selected only those people who like to play fast video games. Our experimental analysis comprises of two phases − a) *Training Phase* in which we introduced three methods to the attackers and gave a demonstration on how attack can be performed by CPM shoulder surfers on BW method. For each of the methods we employed 5 skilled users (total 15) who we believe, can achieve reasonably faster login time. We also split the participants (who will perform the attack) in 3 groups (each having 5). Then we asked them to learn how the attack can be performed. We allow each group to perform the attack on a single method in each day (first day on BW method, second day on FC method and third day on MC method). There were one to one interaction between the skilled users and participants. It took three days to complete the training phase. Each participant faced around 15 rounds for each of the methods in training phase. Next in b) *test phase* fourth, fifth and sixth day we asked the participants to perform the attack on BW, FC and MC method respectively. The attack was performed against 20 login (by skilled user) sessions for each participant. So we have collected total $(20 \times 15) = 300$ results from the participants for each of the methods. We have used smart phones for login.

We have seen that 68.3% of the attackers have been able to identify all the four digits of PIN for BW method. Three of them were able to do it in 18, 16 and 16 sessions (out of 20 sessions). The duration of each login session was about $15 - 16$ seconds (skilled user login time) in BW method. In the next two days of experiment, (meant for FC and MC method) there was a severe degradation in attackers performance. None of them was able to retrieve all

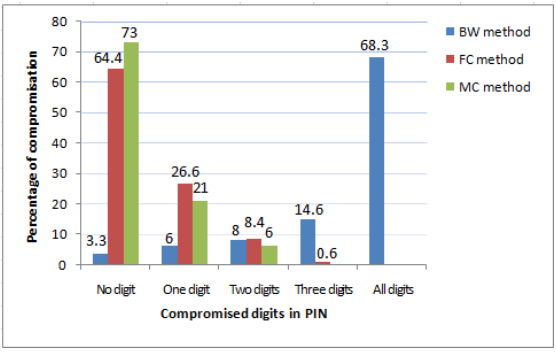

Fig. 9. Above figure shows that CPM shoulder surfers in 68.3% cases can get all PIN digits entered by user using BW method. All digits in the PIN are secure only in 3.3% of cases of BW method and that of 64.4% for FC and 73% for MC method.

four digit successfully in any single session. Many of them even failed to detect a single digit. Fig. 9 shows the performance graph of attacker for BW, FC and MC method. Duration of each session for FC method was $17-18$ seconds (skilled user login time) and that of MC method was $10-11$ seconds (skilled user login time). While performing the attack by video game players, we have observed that non-video game players can achieve the same capabilities as video game players by several practices [13].

Security against Random Key Selection Attack: In each of the BW and FC methods, attacker can proceed successfully with a probability 1/2 in each round by randomly guessing the color buttons. Thus security against this kind of attack in both the method are $(1/2)^{16}$ or, 15.25×10^{-6}. In case of MC method attacker might success in the first round with a probability 3/5 (as 3 colors will appear on user's PIN digit and choosing of anyone is valid for the first round). But in subsequent rounds, the probability of success will get reduces to 1/5 and thus probability of success by the attacker will be $(3/5) \times (1/5)^7$ or, 7.68×10^{-6}, which is further reduced from both BW and FC method.

5 Usability Analysis

While performing usability analysis we have incorporated total 30 participants (21 male and 9 female) whose ages were between $24-45$ years. They all were habituated with touch screen technology and having (correct-to-) normal eye sight. We made three groups and randomly assign ten users to each group. We demonstrated how BW, FC and MC methodologies work and uploaded all three of those in a server so that they can use it to train themselves for login. We have set a global PIN, which is same for all participants for all three methods. During the demonstration we show them by suppressing saccadic eye movement, how one can achieve faster response time and encourage them to do so while login. We also gave all participants 2 days of time to get familiar with the login modules.

During test period we collected the data for one day. We randomly pick each group and assign a random chosen login method to each group so that each group gets one method out of three. No methods were distributed to more than one group. Participant were asked to perform the login using smart phones. Each participant in a group were requested to login ten times using the login methodology allocated to that group. Thus for each methodology we have obtained (10×10) 100 tested data. There after we have performed an analysis regarding login time and percentage of error during login. Fig. 10 shows how login time varies for BW, FC and MC methods ($10.2 - 14.8$ sec for MC method, $17.1 - 24$ sec for FC method and $16 - 22$ sec for BW method). We have taken average login time for each participant to perform statistical significance test. In t test ($t(18) = 0.228$, $P<0.05$) [24] shows no significant difference between BW and FC method in terms of login duration. A one way ANOVA test suggest ($F(2,27) = 35.3$, $P<0.05$) [24] among BW, FC and MC method, at least one method significantly reduces login duration. So cumulative result of both the test suggests, using MC method one can achieve faster login time.

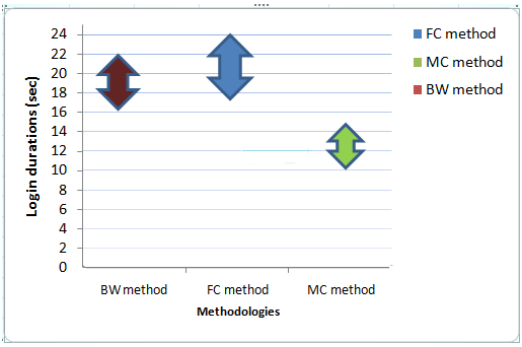

Fig. 10. Above figure shows a comparison of login duration among BW, FC and MC method

Result in Fig.10 shows that login time increases for some users and this is because, few users make a random eye movement during login and do not follow the strategy of "suppression of saccadic eye movement" during login. In this paper we have shown that, by following some smart work user can reduce the login time and thus can avoid the attack while using MC or, FC method. But using BW method no user can avoid such attack and thus this method is vulnerable to shoulder surfing attack performed by CPM shoulder surfers.

Percentage of error occurs during login was estimated as 0.16 for BW method, 0.15 for FC method and that of 0.07 for MC method. Student t test ($t(18) = 0.80$, $P < 0.05$) [24] suggests there exists no significant difference between BW and FC method. A one-way ANOVA test shows ($F\ (2,\ 27) = 3.45$, $p < 0.05$) [24] at least one method among BW, FC and MC method significantly reduces error rate while login. Cumulative result of both the test shows MC method is less prone in terms of login by user.

User Feedback: After performing the usability analysis, we gave all the users a feedback form. Almost all (above 80%) agree that our methodology takes a bit more time ($2-3$ login session) in terms of learning initially. But most of them prefer MC method due to less number of rounds and better security. They also agree that fatigue level using our methodology is much less.

In [18] using CPM-GOMS tool authors showed that skilled user login time using BW method is 960 ms and that of 1080 ms for FC method in each round. They also informed that CPM shoulder surfers require 960 ms to perform the attack on BW method and 1580 ms for FC method (each result was derived using CPM-GOMS tool). Hardness factor greater than 1 suggests a method is secure against CPM shoulder surfers and it increases monotonically. In Table 3 we have presented a summary of comparative analysis among all three methodologies. Pr[SRKS] in Table 3 denotes *probability of success by selecting random keys* for giving response.

Table 3. The outline of comparative features among BW, FC and MC method

	BW method	FC method	MC method
PIN length	4	4	4
Rounds	16	16	8
Hardness factor	1	1.462	1.468
Pr[SRKS]	$1/2^{16}$	$1/2^{16}$	$3/5^{8}$
Login time	More	More	Less

6 Conclusion and Future Work

Strong shoulder surfing attack resilient schemes (that resist recording attack) often require more computational skills from users end and so they are not very commonly used in public domain. Authentication system used in public domain are often targeted by human shoulder surfers and for those systems a better alternative is to use schemes which can resist attack performed by human adversaries. Schemes which can resist such attack are known as weak shoulder surfing resilient schemes.

In this paper we have presented MC Method which is immune to weak shoulder surfing attack performed without any recording device. Our proposed methodology minimizes user effort during login by a large margin. That is a major advantage we have achieved here. However, to achieve this we have not compromised with the security aspect. On the contrary we are able to increase the security level. These two advantages combined together have made the proposed MC scheme to a powerful scheme. We have also shown the comparative study with two existing techniques and found that the proposed technique performs well with respect to those techniques both in terms of usability and security point of view. In future we will try to extend this shoulder surfing resilient scheme against the adversaries with recording device.

Acknowledgments. This work is partially supported by a research grant from the Science & Engineering Research Board (SERB), Government of India, under sanctioned letter no. SB/FTP/ETA-226/2012. Authors also like to thank Mr. Subho Shankar Basu for providing helpful suggestions.

References

1. Banking–Personal Identification Number (PIN) Management and Security–Part 1: Basic Principles and Requirements for Online PIN Handling in ATM and POS Systems, Clause 5.4 Packaging Considerations, ISO 9564-1:2002 (2002)
2. Allen, G., Buxton, R.B., Wong, E.C., Courchesne, E.: Attentional activation of the cerebellum independent of motor involvement. Science 275(5308), 1940–1943 (1997)
3. Anderson, J.R., Matessa, M., Lebiere, C.A.-R.: A theory of higher level cognition and its relation to visual attention. Human-Computer Interaction 12(4), 439–462 (1997)
4. Bai, X., Gu, W., Chellappan, S., Wang, X., Xuan, D., Ma, B.P.: PAS: predicate-based authentication services against powerful passive adversaries. In: Annual Computer Security Applications Conference, ACSAC, pp. 433–442. IEEE (2008)
5. Bavelier, D., Achtman, R., Mani, M., Föcker, J.: Neural bases of selective attention in action video game players. Vision Research 61, 132–143 (2012)
6. Bi, X., Li, Y., Zhai, S.: FFitts law: modeling finger touch with fitts' law. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 1363–1372. ACM (2013)
7. Blonder, G.: Graphical passwords. lucent technologies, inc., murray hill, nj. US patent, ed. United States (June 1996)
8. Brady, T.F., Konkle, T., Alvarez, G.: A review of visual memory capacity: Beyond individual items and toward structured representations. Journal of Vision 11(5), 1–34 (2011)
9. Card, S.K., Moran, T.P., Newell, A.: The psychology of human computer interaction hillsdale. LEA, NJ (1983)
10. Carroll, J.M.: HCI models, theories, and frameworks: Toward a multidisciplinary science. Morgan Kaufmann (2003)
11. Chakraborty, N., Mondal, S.: Color Pass: An intelligent user interface to resist shoulder surfing attack. In: IEEE Students' Technology Symposium (TechSym), pp. 13–18 (2014)
12. Chakraborty, N., Mondal, S.: SLASS: Secure login against shoulder surfing. In: Martínez Pérez, G., Thampi, S.M., Ko, R., Shu, L. (eds.) SNDS 2014. CCIS, vol. 420, pp. 346–357. Springer, Heidelberg (2014)
13. Green, C.S., Bavelier, D.: Action video game modifies visual selective attention. Nature 423(6939), 534–537 (2003)
14. Holz, T., Engelberth, M., Freiling, F.: Learning more about the underground economy: A case-study of keyloggers and dropzones. In: Backes, M., Ning, P. (eds.) ESORICS 2009. LNCS, vol. 5789, pp. 1–18. Springer, Heidelberg (2009)
15. John, B.E.: Extensions of GOMS analyses to expert performance requiring perception of dynamic visual and auditory information. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 107–116. ACM (1990)
16. John, B.E., Gray, W.D.: CPM-GOMS: an analysis method for tasks with parallel activities. In: Conference Companion on Human Factors in Computing Systems, pp. 393–394. ACM (1995)

17. John, B.E., Kieras, D.E.: The GOMS family of user interface analysis techniques: comparison and contrast. ACM Transactions on Computer-Human Interaction (TOCHI) 3(4), 320–351 (1996)
18. Kwon, T., Shin, S., Na, S.: Covert Attentional Shoulder Surfing: Human Adversaries Are More Powerful Than Expected. IEEE Transactions On Systems, Man, and Cybernatics: Systems 44(6) (2013)
19. Lowe, D.G.: Perceptual Organization and Visual Recognition. Tech. rep., DTIC Document (1984)
20. Luck, S.J., Vogel, E.K.: The capacity of visual working memory for features and conjunctions. Nature 390(6657), 279–281 (1997)
21. Posner, M.I.: Orienting of Attention*. Quart. J. Experimental Psychology 32(1), 3–25 (1980)
22. Rabinbach, A.: The human motor: Energy, fatigue, and the origins of modernity. Univ of California Press (1992)
23. Rayner, K., White, S.J., Kambe, G., Miller, B., Liversedge, S.P.: On the processing of meaning from parafoveal vision during eye fixations in reading. In: The Minds Eye: Cognitive and Applied Aspects of Eye Movement Research, pp. 213–234 (2003)
24. Rosenkrantz, W.A.: Introduction to Probability and Statistics for Science, Engineering, and Finance. CRC Press (2011)
25. Treisman, A.M., Kanwisher, N.G.: Perceiving visually presented objects: Recognition, awareness, and modularity. Current Opinion Neurobiol. 8(2), 218–226 (1998)
26. Roth, V., Ritcher, K., Freidinger, R.: A PIN-entry method resilient against shoulder surfing. In: ACM Conf. Comput. Commun. Security, pp. 236–245 (2004)
27. Wiedenbeck, S., Waters, J., Sobrado, L., Birget, J.-C.: Design and Evaluation of a Shoulder-Surfing Resistant Graphical Password Scheme. In: ACM Working Conference Advance Visual Interfaces, pp. 177–184 (2006)
28. Yan, Q., Han, J., Li, Y., Deng, R.H.: On Limitations of Designing Leakage-Resilient Password Systems: Attacks, Principles and Usability. In: 19th Internet Social Network Distributed System Security (NDSS) Symposium (2012)
29. Zhao, H., Li, X.: S3PAS: A scalable shoulder-surfing resistant textual-graphical password authentication scheme. In: 21st International Conference on Advanced Information Networking and Applications Workshops, pp. 467–472 (2007)