DynamicPIN: A Novel Approach towards Secure ATM Authentication

Jong-Hoon Kim
ATR Lab, Dept. of Computer Science
Kent State Univ., Kent, Ohio, USA
jkim72@kent.edu

Gokarna Sharma
Dept. of Computer Science
Kent State Univ., Kent, Ohio, USA
gsharma2@kent.edu

Irvin Steve Cardenas
ATR Lab, Dept. of Computer Science
Kent State Univ., Kent, Ohio, USA
irvin@irvincardenas.com

Do Yeon Kim
Dept. of Biomedical Engineering
Hanyang Univ., Seoul, South Korea
dkim9681@hanyang.ac.kr

Nagarajan Prabakar
Discovery Lab, SCIS
Florida International Univ.
Miami, FL, USA
prabakar@cis.fiu.edu

S.S. Iyengar
Discovery Lab, SCIS
Florida International Univ.
Miami, FL, USA
iyengar@cis.fiu.edu

***Abstract*—Along with the popularity and widespread use of automated teller machines (ATMs), ATM frauds are also increasing drastically these days. Shoulder-surfing attacks, such as card skimming, PIN capturing using fake machines or fake PIN pads, are the most common methods used by adversaries to capture data from the magnetic stripe on the back of the ATM card. The main problem lies in the existing static PIN-based authentication mechanism which does not provide any security measure (besides displaying asterisks when an user enters a preassigned PIN to the ATM). In this paper, we give a novel approach called DynamicPIN for secure ATM authentication, which is resilient to shoulder-surfing attacks. DynamicPIN is very simple, does not require any hardware changes, and does not pose any significant overhead to the system. A real-time experimental study showed that DynamicPIN improves significantly the ATM authentication compared to the existing static PIN-based authentication mechanism.**

***Keywords*-Authentication; Dynamic Password; Dynamic PIN; ATM authentication; ATM frauds;**

I. Introduction

ATM is a computerized device that provides the customers of a financial institution with access to financial transactions without the need of any human (teller/clerk) assistance round the clock. The most modern ATMs use a plastic ATM card (with a magnetic stripe or a smart card with a chip) to identify the customer. These cards generally contain a unique card number and some security information based on personal identification number (PIN). A PIN is a secret numeric password shared between the user and the system, which is used to authenticate the user to the system. The PINs for ATM cards are usually 4-digit numbers in the range 0000–9999; identical numbers (e.g. 4444, 5555, $\cdots$), consecutive numbers (e.g. 1234, 2345, $\cdots$), or the last 4 digits of SSN or birth date are not allowed.

The main security problem in existing PIN-based ATM authentication is that the PIN is static (users use the same PIN every time they use ATM and it does not change over time; hereafter we call it StaticPIN). When adversaries get to know StaticPIN once by some means, they can use it for subsequent fraud transactions (and also to make fraud ATM cards). The users need to actively take care of securing their PIN input – by hiding the input with second hand, by checking for ATM manipulations or suspected devices attached to the ATM, or by noticing the suspected person(s) nearby – to save their PIN from being captured. Therefore, the PIN security is entirely based on the user's behavior. However, most of the time, users could not take care of such security measures due to different physical and environmental constraints. As several previous studies (e.g. [1]) show, the user's behavior itself opens the security holes in the authentication and adversaries exploit these holes. Thus, a fundamental problem in the PIN-based authentication is to design a PIN entry mechanism in such a way that it does not rely on the user's behavior.

A. Motivation and Contributions

This paper is motivated by a deficit of academic literatures exploring the topic of ATM security from different ATM theft scams. We mainly focus on shoulder-surfing attacks (email phishing or card and cash trapping related scams are outside the scope of paper). A number of research studies has been done to avoid or at least minimize shoulder-surfing attacks (e.g., [2–9, 11, 12]). These techniques have several limitations and some of them have very high overhead and complex that the real world implementation would not be feasible. For example, the PIN entry method of Roth et al. [8] requires several rounds to input a single digit of PIN, which limits its usability and resilience against shoulder-surfing. Similarly, FacePIN [3] and ColorPIN [2] take more login time, RotaryPIN [9] requires additional hardware and many training/practice sessions, graphical password of [7] requires lengthened password selection process, virtual password of [6] requires helper application, and so on.

From these observations, we advocate that an ATM authentication mechanism should fulfill the following require-

ments: (i) It should not require an user to protect the PIN input; (ii) It should be strongly resilient to shoulder-surfing attacks (e.g., card skimming, PIN capturing, fake PIN pads, and fake machines); (iii) It should pose very negligible overhead to the system than the existing StaticPIN-based authentication; (iv) It should not require any significant hardware changes; and (v) It should be user-friendly, reliable, and within the capacity of human's memory. To this end, we propose DynamicPIN, a novel PIN entry mechanism based on the combination of a StaticPIN and some random number that is carefully chosen, that fulfills the aforementioned requirements (Section III). An experimental study confirmed that our approach enhance significantly the security of the ATM authentication compared to the existing StaticPIN-based authentication. The benefit of our scheme is that the mechanism is very simple, it avoids shoulder-surfing attacks, and the overhead incurred is very low.

One proposal that is similar to DynamicPIN is the US Patent by G.T. Wilfong [12]. In his method, the operator challenges the user with a random number explicitly, one after another, for each digit of the PIN that need to be entered by the user. After the user performs modulo 10 arithmetic operation (+, -, or *) for each digit of his StaticPIN with the digits of the random PIN (generated from random numbers given to the user), the operator inverts the calculation by subtracting the random PIN from the entered PIN. In contrast to our DynamicPIN, this method is vulnerable to shoulder-surfing attacks because when an adversary track the random numbers provided by the operator and the PIN entered by the user one time, he can easily calculate the actual PIN.

B. Related Work

A wide range of research that has been done to overcome shoulder-surfing (or at least to minimize) related ATM security problems [2–5, 5–9, 11, 12]. One such research work is graphical passwords [7], in which authors present a technique that demonstrates multiple graphical passwords are substantially more effective than multiple PIN numbers and increase the password memorability such that they cannot simply be stolen by educated random guessing or from shoulder-surfing attacks. Some other studies (e.g., [4, 11]) also considered techniques to provide shoulder-surfing resistant graphical passwords.

One very promising approach to make PIN entry more secure is indirect input (e.g., [2, 8, 10]), which means that some kind of "detour" is used instead of inputting authentication PIN directly. De Luca et al. [2] proposed ColorPIN, an authentication mechanism that uses indirect input to provide security enhanced PIN entry, and showed that it is notably secure than StaticPIN entry. Later, they conducted a field study which showed a big influence of contextual factors on security and performance in PIN-based ATM authentication and need for the design of alternative ATM authentication mechanisms that are resilient to distraction and social compatibility. Another proposal is by Roth et al. [8], where they created a PIN entry mechanism using a cognitive trapdoor game. In their mechanism, four key presses are required for each digit. The spy-resistant keyboard [10] hides the input in the similar way. Two to four clicks are required for each digit in their mechanism. Both systems are resistant to shoulder-surfing except the camera based attacks.

And, some other proposals provide shoulder-surfing resilient PIN entry mechanisms are: a RotaryPIN [9], FacePIN [3], virtual passwords [6], etc. The main problem of indirect input is that most systems that rely on this approach add significant overhead to the input and some are considerably complex to use in real world scenarios.

II. Threat Model

We assume that the number of digits in the PIN is $\ell = 4$ and it is made from any combination of the digits $\{0, 1, \cdots, 9\}$. We also assume that the adversary can have full access to the ATM machine at which the authentication will take place. Additional hardware such as video camera may have been installed to obtain the information stored on the magnetic strip of the ATM card. Additionally, the keypad may also be manipulated. The adversary can also be able to do shoulder-surfing, that is, the attacker can stand close to the ATM to gaze on the user's input. Thus, the protection of the authentication mechanism relies solely on the security of the PIN input. In this scenario, we consider two different threat models as given below:

- *Zero-knowledge adversary model:* In this model, the adversary has only the card but no knowledge about the PIN. Thus the only option for the adversary is to perform random guessing attacks on the ATM.
- *Shoulder-surfing adversary model:* In this model, we assume that the adversary can acquire up to *r* records of the entire authentication process of a user by means of the concealed camera installed over the input screen of the ATM. We also assume that the adversary can capture the card information, including the account number, balance, and PIN number, and the user is unaware about these devices.

We assume that the verification system of ATM will keep a record on the number of successive PIN entry failures. Once detecting that this number reaches a predetermined threshold (e.g., 4), it will retain the card and suspend the account, until the PIN is reset through a secure channel (e.g., at a bank branch). Please note that this counter will be automatically reset upon a successful login. Moreover, we assume that upon a successful login the legitimate user will be notified previous PIN entry failures, if any.

III. DynamicPIN

The goal of DynamicPIN entry mechanism is to provide a PIN that is usable for one time input for users without the

Base PIN: 1-2-3-4; **Preselected operation:** plus (+); **The user's preselected target digit:** 3rdSD of Base PIN **User's security information – SSN:** 987-65-4321; **Phone:** 123-456-7890		
First operand: pre-selected target digit **Second operand:** digit on ? mark	**First operand:** random digit (first ? mark) **Second operand:** digit on second ? mark	**First operand:** all base PIN digits **Second operand:** digit on ? mark
Mask question: ###−?#−####(SSN) **DynamicPIN:** 1-2-9-4	**Mask question:** #?#−###−##?#(Phone) **DynamicPIN:** 1-1-3-4	**Mask question:** ###−?#−####(SSN) **DynamicPIN:** 7-8-9-0

Table I: An example of DynamicPIN generation using FixedTarget, RandomTarget, and AllTarget, respectively

need of any significant hardware changes (i.e., installation of extra devices), and at the same time, to achieve simplicity and userfriendliness in use. An advantage of this mechanism is that the PIN overlooked by the adversary using some shoulder-surfing mechanism will not be valid for the next time use. In contrast to the previous proposals (e.g., [2, 9]), our goal is to make the mechanism very simple to use in real world scenarios. In this quest, DynamicPIN achieves overwhelming level of security and remains in one-to-one relationship with the PIN length and the number of key presses required. DynamicPIN is generated from the combination of the following four attributes:

- **Base PIN:** The base is the 4 digit PIN that is same as the StaticPIN, which an user has to remember to make a transaction in the existing ATM authentication system. A StaticPIN is used as the base for the DynamicPIN and it does not change until the card user resets it through a secure channel from a bank branch. Combining it with other attributes given below, we build the PIN that change every time the user performs the PIN entry.
- **Target digit:** This is the digit in the base PIN the user need to change while performing PIN entry. This is the first operand for the arithmetic operation to generate the DynamicPIN. For simplicity, the authentication system can be configured with preselected target digit by the user or can be randomly selected and provided it as a masked security question (details later).
- **Mask digit:** This is the random digit provided to the user as the second operand for the arithmetic operation at the PIN entry time. The random digit is generated from the user's security information (such as his/her SSN, birthday, or phone number). The system randomly chooses a digit among the digits available in the selected information and provide to the user as masked security question (i.e., the digit is not shown) at the time of PIN entry. The user then enter the PIN digit generated from the arithmetic operation. For enhanced security, the security information can also be randomly selected from one of the user's security information at the PIN entry time. If no mask digit is used, DynamicPIN will be equivalent to StaticPIN.
- **Arithmetic operation:** The arithmetic operation for the target digit is either of: plus (+), minus (-), or multiplication ($*$). Considering the complexity of the calculation, we will not use the division operation. Moreover, we use modular arithmetic to assure that the correct input is always a non-negative single digit. Without the use of modular arithmetic, the single-digit input could itself reveal information itself to the shoulder-surfing adversary.

Based on the aforementioned four attributes, we consider following variations of DynamicPIN (different other variations are also possible such as the ones described in Section IV).

1) **FixedTarget:** In this variation, the target digit is preselected by the user. When the user inserts the card the system responds with the masked security question as "###−?#−####". Now the user should figure out the number in the ? position from one of user's security information (that is used for the mask question at that time) and perform the preselected arithmetic operation with the number in the target digit. An example for FixedTarget is given in the left column of Table I assuming the base PIN: 1-2-3-4 (1stSD-2ndSD-3rdSD-LSD)[1], and social security number and phone number as the security information. Moreover, we assume that the user has pre-selected the plus (+) operation as his preferred arithmetic operation and the user's SSN is 987-65-4321[2]. That is, the masked security question looks like "SSN:###−?#−####", where the ? is the digit which the user need to use as the second operand in the arithmetic operation the user will perform. As the first operand is 3 (3rdSD in PIN) and second operand is 6 (4thSD in his SSN masked by ? mark), the DynamicPIN generated for that time using this FixedTarget method is 1-2-9-4 = 1-2-(3 + 6)-4.
2) **RandomTarget:** In this variation, in contrast to the preselected target digit of FixedTarget, target digit is selected at the PIN entry time from the masked security question. For that purpose, there are two digits with ? mark randomly selected from the user's security information. The first ? mark digit is to choose the target digit (the first operand), and the second ? mark gives the mask digit (the second

[1] 1stSD–first significant digit, LSD–last significant digit, and so on from beginning to end.

[2] The security information, such as SSN, phone number, is not explicitly displayed when the system use them as a masked security question.

operand). For example, the masked security question "Phone:#?# – ### – ##?#" means that the first ? mark is target digit in base PIN and second ? mark is mask digit. An example is given in middle column of Table I, where the preselected arithmetic operation (+) is done between 2 (2ndSD in base PIN given by first ? mark in phone number) and 9 (9thSD in phone number as second ? mark). The DynamicPIN generated using this RandomTarget method is 1-1-3-4 = 1-(2 + 9)-3-4. For the cases where the digit at first ? is greater than 4, the modulo 4 arithmetic is used to find the target digit.

3) **AllTarget:** This variation of DynamicPIN generation is similar to FixedTarget with the difference that all base PIN digits are used as target digits. Now the user performs the preselected arithmetic operation between the mask digit with all digits in base PIN. The right column of Table I gives one example where mask digit 6 (4thSD in SSN) is added to all digits of base PIN 1-2-3-4, which gives 7-8-9-0 = (1 + 6)-(2 + 6)-(3 + 6)-(4 + 6) (Note that modulo 10 arithmetic is used in calculation).

The description of DynamicPIN verification procedure is given in Fig. 1. After the user insert card to the ATM, the ATM authentication system retrieves security information of the user and randomly selects one of the security question from either of: SSN, birthday, phone number, or driving license number. Then the system builds and sends the masked security question to the user. The user performs the preselected arithmetic operation ($+$, $-$, or $*$) with the selected target digit and the random digit on the security question according to the DynamicPIN variation used. If the DynamicPIN entered by the user is correct, the transaction will be approved, otherwise it will be rejected. If the DynamicPIN attempts cross some predefined threshold (e.g., 4), the account will be automatically locked.

A. Security Analysis

We analyze the level of security provided by DynamicPIN in both threat models. Compared to StaticPIN, DynamicPIN is resilient to a number of simple attacks. Take for an example the educated guessing attacks, where an adversary knows the entered PIN, there is still the secret information about the target digit, mask digit, and arithmetic operation, which makes it way harder to guess. DynamicPIN is also an effective countermeasure against shoulder-surfing including hidden camera recordings of the keypad or the fake PIN pads. Even if the adversary see or record the whole input, the base PIN remains hidden, which allows the ATM authentication security totally independent of the user. The theoretical comparison of StaticPIN and DynamicPIN properties is given in Table II.

1) Zero-knowledge Adversary Model: In this model, the adversary knows nothing about the base PIN, and thus

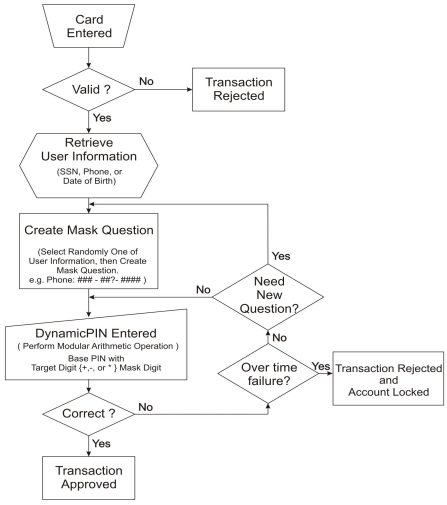


Figure 1: DynamicPIN-based ATM authentication procedure

can only launch a random guessing attack. In this case, the adversary needs to choose each digit from 10 possible choices, that is, the probability of success in one random attack is $1/10$ for each digit of PIN. The probability of success is then goes up to $(1/10)^4$ for all 4 digits. Even if the adversary excludes identical numbers, consecutive numbers, etc., the probability of success in one random attack still remains very low.

2) Shoulder-surfing Adversary Model: In this model, at first, we assume that the adversary can acquire up to one record of the whole PIN entry process, and later extend it for r records.

- **One record:** If an adversary can acquire up to one record of the entire authentication process and the DynamicPIN method used is FixedTarget, the adversary still needs to launch a random guessing attack on all base PIN digits since he may not know the actual target digit from one record. This gives the probability of success $(1/10)^4$, the same as Zero-knowledge Adversary Model. Moreover, even if the adversary knows the target digit, he still needs to choose one mask digit among 10 digits, i.e., the success probability is 1/10. If the DynamicPIN generation method used by the user is RandomTarget or AllTarget, the success probability remains $(1/10)^4$ as target digit is always changing in RandomTarget and all digits are changing in AllTarget (which the adversary is not aware about).
- r **records:** If an adversary can acquire up to r records of the authentication process and the DynamicPIN method used is FixedTarget, the probability of success is 1/10 for any $r >= 2$ (for $r = 2$, see Table II) because the adversary might know the target digit as $\ell - 1$ digits of PIN are static. The success probability of FixedTarget also applies for AllTarget using similar argument. If

	StaticPIN	DynamicPIN
Security depends	on the user	not on the user
Token	digits	Base PIN $\{+,-,*\}$ mask on target digit
Example	2376	2376 + ###$-$?#$-$####
Theoretical password space	10,000 (random guess: 1/10,000)	10,000 (random guess: 1/10,000)
successful attack from one record	1:1	$1:(10)^4$
from two records	1:1	$1:10^*\|1:400^{**}$
from r records	1:1	$1:10^*\|1:40^{**}$

Table II: Theoretical comparison between StaticPIN and DynamicPIN. *This is for FixedTarget and AllTarget methods. **This is for RandomTarget method.

the method used is RandomTarget, the probability of success for the adversary is still $1/40$ as the target and mask digit choice for the current round are completely independent of previous rounds.

If the PIN input as well as the screen is recorded for several authentication sessions of the same user, an intersection analysis may be successful, but the probability is still very low due to the nature of DynamicPIN design. Moreover, we believe recording many authentication sessions is very unlikely in a short period of time.

IV. Experimental Results

For the experimental evaluation, we built a virtual ATM prototype web-server such that it can be accessed from anywhere using any internet browser and used in the backend to record all the information for DynamicPIN analysis. The study set up consisted of a standard desktop PC with a standard commercial keyboard attached (one key per letter).

The study was started by creating the user profile to record their basic information and then they were allowed to login to the system using their username and base PIN. For the experimentation purpose, we consider following six different variations of DynamicPIN:

- **BasePIN:** is the same as StaticPIN.
- **FixedReplace:** is a variation of FixedTarget where preselected target digit of base PIN is replaced with the mask digit (no arithmetic operation used).
- **RandReplace:** is a variation of RandomTarget where base PIN digit at position given by first ? mark of security question is replaced the number at second ? mark.
- **FixedSum:** is FixedTarget method with $+$ as the only arithmetic operation.
- **RandSum:** is RandomTarget method with $+$ as the only arithmetic operation.
- **AllSum:** is AllTarget method with $+$ as the only arithmetic operation.

Overall 30 graduate and undergraduate students participated in the experimental study and the results are based on several test sessions (> 50) of each participant. In the study, the time is measured starting the user press start button to enter the PIN until he/she presses enter to signify the end of the PIN input. The average time taken to enter the PIN is given in Fig. 2a, where average time for the BasePIN is around 3s which is the least. We use this as the base for comparing the performance of other variations. RandSum takes the highest average time which is around 17s. After that RandReplace which gives 15s followed by AllSum, FixedSum, and FixedReplace each of average authentication time 11s, 10s, and 8.5s, respectively.

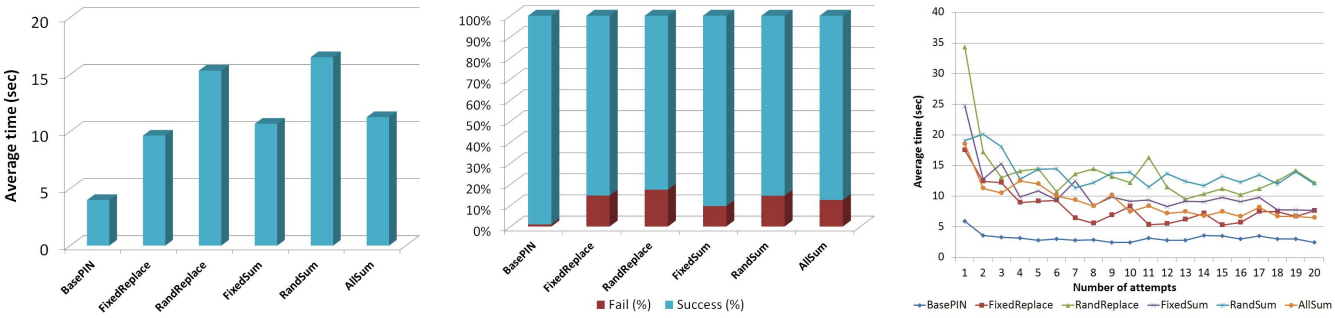
Fig. 2b shows the error rate comparison of the variations of DynamicPIN used in the experiments. BasePIN has the best performance with negligible error rate. The most error prone variations are RandomTarget variations (RandReplace and RandSum) due to the complexity in first choosing the target digit and then perform the selected operation. FixedTarget variations (FixedReplace, FixedSum) have relatively high success rate due to less complex operation than RandomTarget variations. AllSum method achieves similar performance as of FixedTarget variations.

We also plot authentication speed over number of attempts in Fig. 2c. The motivation was to verify how the average time changes after users have some experience of the environment. We answer it affirmatively since the average time required to authenticate using each DynamicPIN variations is decreasing with the increasing number of attempts.

Our study showed that, in comparison to the level of security provided to shoulder-surfing attacks by DynamicPIN against StaticPIN, the overhead due to average error rate and authentication time is considerably negligible. Moreover, DynamicPIN showed improved performance (sometime comparable, e.g., Hayashi et al. 12.4s [4]) compared with the time needed on the mechanisms proposed on previous studies: Tan et al. 50s [10], and Roth et al. 23.3s [8].

V. Applications

DynamicPIN can be used for enhancing authentication on many offline as well as online security applications. One of the prominent offline applications is the gate security system that uses StaticPIN-based authentication. The system can be made resilient to shoulder-surfing related attacks by simply upgrading to DynamicPIN. The benefit of our approach is easily upgradable in considerably low cost and without the



(a) Average authentication time (b) Error rate (c) Authentication speed over number of attempts

Figure 2: Evaluation of StaticPIN and different variations of DynamicPIN

need of any major hardware changes (for new systems as well as existing systems).

One example of the online applications is the market place where credit card transactions are verified from user's credit card number, expiration date, and CVV number (all of them static). Security in such applications can be enhanced by adding DynamicPIN-based authentication mechanism, which hides the secure code to both the website and the possible adversary. In addition, it will also secure the user security code from accidental release via website. DynamicPIN can also be used for enhancing security in login procedure for any web application that is vulnerable to wiretapping via backdoor programming. We believe that there should be many more challenges where security can be enhanced using our approach.

VI. Discussions and Future Directions

We proposed a novel concept of DynamicPIN for significantly enhancing the security on existing ATM authentication systems. Our mechanism does not need complex operations and also does not need to exploit the user's additional memory. This is because memorization of other numbers such as telephone numbers (besides base PIN) involved in the computation of DynamicPIN does not put extra overhead to the user since such information is presumed to be known by the user all the time.

We conclude from experimental study that DynamicPIN will provide notably secure ATM authentication compared with the standard StaticPIN entry. But, at the same time, it will be little bit slower due to the extra load on arithmetic operations at the time of PIN entry. We advocate that the system becomes significantly faster after repeated use of DynamicPIN for a certain period of time as shown in Fig. 2c. Moreover, DynamicPIN improves on the previous studies (e.g., [2, 7]) as it does not require complex interaction based on colors and graphics; the only need is the basic knowledge of arithmetic operations.

However, we observe from the experiments that a few participants had difficulty even with the simplest substitution or addition operations (as some people are generally bad at mental arithmetic). To accommodate these scenarios, the system should provide options to choose from either the basic StaticPIN-based authentication or the enhanced DynamicPIN-based authentication in the beginning of the authentication session.

For future directions, we plan to investigate many different applications (as given in Section V) of DynamicPIN in detail, and also to explore its formal performance aspects. Moreover, we plan to consider the effects of multiple passwords that apply this scheme, that is, when the users have multiple PINs for different banks and they have to transform the PINs in unique ways such that our scheme can be applicable with different banks.

References

[1] Anne Adams and Martina Angela Sasse. Users are not the enemy. *Commun. ACM*, 42:40–46, December 1999.

[2] Alexander De Luca, Katja Hertzschuch, and Heinrich Hussmann. Colorpin: securing pin entry through indirect input. In *Proc. of CHI '10*, pages 1103–1106. ACM, 2010.

[3] Paul Dunphy and Jeff Yan. Is facepin secure and usable? In *Proc. of SOUPS '07*, pages 165–166, New York, NY, USA, 2007. ACM.

[4] Eiji Hayashi, Rachna Dhamija, Nicolas Christin, and Adrian Perrig. Use your illusion: secure authentication usable anywhere. In *Proc. of SOUPS '08*, pages 35–45. ACM, 2008.

[5] G. Renee Jebaline and S. Gomathi. A novel method to enhance the security of atm using biometrics. In *Proc. of ICCPCT '15*, Nagercoil, India, 2015. IEEE Computer Society.

[6] Ming Lei, Yang Xiao, Susan V. Vrbsky, and Chung-Chih Li. Virtual password using random linear functions for on-line services, atm machines, and pervasive computing. *Comput. Commun.*, 31:4367–4375, December 2008.

[7] Wendy Moncur and Grégory Leplâtre. Pictures at the atm: exploring the usability of multiple graphical passwords. In *Proc. of CHI '07*, pages 887–894. ACM, 2007.

[8] Volker Roth, Kai Richter, and Rene Freidinger. A pin-entry method resilient against shoulder surfing. In *Proc. of CCS '04*, pages 236–245. ACM, 2004.

[9] Peipei Shi, Bo Zhu, and A. Youssef. A rotary pin entry scheme resilient to shoulder-surfing. In *Proc. of ICITST '09*, pages 1–7, 2009.

[10] Desney S. Tan, Pedram Keyani, and Mary Czerwinski. Spy-resistant keyboard: more secure password entry on public touch screen displays. In *Proc. of OZCHI '05*, pages 1–10, Australia, 2005.

[11] Daphna Weinshall and Scott Kirkpatrick. Passwords you'll never forget, but can't recall. In *CHI EA '04 extended abstracts on Human factors in computing systems*, pages 1399–1402. ACM, 2004.

[12] Gordon Thomas Wilfong. Method and apparatus for secure pin entry, patent number: 5940511, August 1999.