A New Security Solution Enhancing the Dynamic Array PIN Protocol

Samir Chabbi, Nour El Madhoun

A New Security Solution Enhancing the Dynamic Array PIN Protocol

Samir Chabbi*, Nour El Madhoun†
* Department of Mathematics and Computer Science, University of Souk Ahras, 41000, Souk Ahras, Algeria
† Security and System Laboratory, EPITA, 14-16 Rue Voltaire, 94270, Le Kremlin-Bicêtre, France
Email: s.chabi@univ-soukahras.dz; nour.el-madhoun@epita.fr

***Abstract*—In order to authenticate a user on an Automated Teller Machine (ATM) using Near Field Communication (NFC) technology embedded on smartphones, we recently proposed a new approach called Dynamic Array PIN Protocol (DAP) that allows a user to enter his PIN code in a secure manner. We proved that the DAP protocol is resistant to 13 different attacks. Furthermore, by comparing it to several existing solutions, we demonstrated that DAP is much better and more cost effective. However, after a thorough analysis, we discovered that the DAP protocol is vulnerable to multiple eavesdropping video or camera records attack. Consequently, in this paper, we aim to address this vulnerability by proposing a new security solution that improves the DAP protocol.**

***Index Terms*—ATM, DAP, NFC, NFC smartphone, Payment, PIN Authentication, PIN, Transaction, Vulnerability.**

I. Introduction

Near Field Communication (NFC) is a technology used to enable contactless communication, within a short distance, between an NFC reader such as an Automated Teller Machine (ATM) or a Point of Sale Machine (PoS) and an NFC bank card or an NFC smartphone [1]. NFC technology transfers data in a frequency of 13.56 MHz using an electromagnetic field [2]. Today, this technology is frequently used and found in many proximity payment products [3]. For example, a client can use his NFC smartphone or his NFC bank card in front of a PoS to make an NFC payment or his NFC smartphone in front of an ATM to withdraw/deposit money [4].

NFC technology is therefore beneficial to both users and banks as it allows fast contactless transactions at any ATM or PoS [5]. Unfortunately, an attacker is able to remotely steal the banking data stored in an NFC bank card or an NFC smartphone, without the knowledge of the client [6]. In addition, the user's password or the Personal Identification Number (PIN) can be stolen through various attacks such as: spyware [7], shoulder-surfing [8], side channel [9], brute force [10], replay [11], smudge [12], camera recording [13], video recording [14] and multiple registration [15].

Indeed, in order to deposit or withdraw money using an NFC smartphone in front of an ATM, the user must authenticate himself. Therefore, the authentication of the user is an important security property that must be confirmed in order to protect access to a resource [16]. By consulting existing literature, there are several solutions proposed to authenticate a user in order to secure his contactless communication with an ATM: some solutions require the entry of a PIN code or a password and others use a biometric modality of the user [17] such as fingerprint recognition, facial recognition, voice recognition, etc. According to existing literature, all of these proposed solutions have their advantages and limits.

In fact, when comparing the PIN authentication technique to those using biometric modalities [18], we found that the burden of the latter outweigh their advantages. In most cases, the major disadvantage of the use of a biometric modality is that when it is stolen, the attacker can use it at any time insofar as the user cannot change it, contrarily to a PIN code. Consequently, since the PIN code is also exposed to various attacks, its security is then very important when an NFC smartphone is used to make a transaction with an ATM [19] and to ensure the security of a user's PIN storage in an NFC smartphone, the PIN code is stored in a hardware circuit called the Secure Element (SE). The latter ensures the security of the data storage and the execution of the sensitive applications it embeds, such as a payment application [20].

In our previous paper [21], we proposed a new PIN entry protocol on a smartphone to authenticate a user during an NFC communication with an ATM. We proved that this protocol, called Dynamic Array PIN (DAP), is more cost effective and secure than several protocols proposed in existing literature. The DAP protocol is executed when the client approaches his NFC smartphone to an ATM to authenticate himself. However, we later discovered that the DAP protocol is vulnerable to the intersection of multiple records (video eavesdropping or camera recording) attack, where the attacker can steal the user's PIN code [22]. This vulnerability is illustrated in our previous paper [23]. Consequently, our objective in this paper is to solve this vulnerability by proposing a new security solution that improves the DAP protocol.

This paper is organized as follows. In section II, we discuss the related work. In section III, we describe our new proposed solution to solve the vulnerability of the DAP protocol. In section IV, we prove the security of our proposal by a demonstration. The last section concludes the paper and outlines future work.

II. Related work

In order to provide secure authentication to a user through a password or a PIN code to perform an NFC transaction between an NFC smartphone and an ATM, several solutions have been proposed in existing literature. In this section, we

present the most important and recent solutions and analyze their strengths and weaknesses.

A. BrightPass Solution

The principle of this solution is the generation of a sequence of 0 and 1, called Lie Overhead by running an application embedded in the SE. From this sequence, a series of circles with different luminosity will be displayed on the screen of the smartphone to introduce the PIN code. To enter the PIN code, the user enters a random digit that is not part of the PIN code in a low brightness circle (corresponding to the value 0), and enters a real digit that is part of the PIN code in a high brightness circle (corresponding to the value 1) [24]. The analysis of this solution shows that it protects the PIN code from spyware attacks that discover the typed PIN by capturing the phone screen [25]. Contrarily, it is vulnerable to the shoulder surfing attack and the camera recording attack.

B. Color Wheel PIN (CWPIN) Method

In order to authenticate a user with the Color Wheel PIN (CWPIN) method, the bank server uses a secret consisting of a PIN code and an array of ten randomly generated colors, each of which is indexed by a random number from 0 to 9, for example: 0 corresponds to green, 1 to yellow, etc. This secret will be shared with the user and at each authentication attempt, the server sends the indexed table to the ATM which uses a QR code or NFC radio waves to authenticate the user via his smartphone. After the NFC communication with the smartphone, the ATM screen displays a wheel of 10 colors surrounded by a fixed index from 0 to 9 and a search bar to make it turn. The colors and corresponding numbers are generated randomly. For the smartphone, the SE displays an array of 10 randomly generated colors surrounded by an index of 10 randomly generated digits [26]. In order to authenticate, the user searches the array displayed on the smartphone screen for the color indexed by the first digit of his PIN code, and then on the ATM screen, he turns the wheel to match that color with the second digit of his PIN. In this way, the user enters the first two digits of his PIN. When the user releases the wheel, the wheel rotates randomly until it stabilizes. Then, the user repeats the previous step, finds the color corresponding to the third digit of his PIN code in the array displayed on the smartphone, and then rotates the wheel displayed on the ATM to match this color with the fourth digit of the PIN and doing this, he enters the last two digits of the PIN code [26]. In this way, CWPIN protects the PIN code from smudge attacks and malware. However, it is not applicable if the PIN has an odd number of digits.

C. DAP Protocol

DAP is a solution that we proposed in [21] allowing to authenticate a user by entering a PIN code using a smartphone during an NFC communication with an ATM. In this technique, the PIN code is the only secret shared between the bank server and the user, the smartphone used integrates the SE and the ATM is equipped with a small touch pad covered with a shell to hide the user's finger movements. In order to authenticate, the user presents his smartphone in front of the NFC reader of the ATM. At that moment, the ATM displays on its screen two vertically aligned arrays, each containing 10 random numbers. The digits in the upper array are fixed on the screen during the authentication session, but the digits in the lower array can slide horizontally following the movement of the user's finger on the ATM touchpad, and can rotate in a circular manner so that cells that disappear on one side immediately appear on the opposite side in the same order. The user authentication in the DAP protocol requires the following steps [21]:

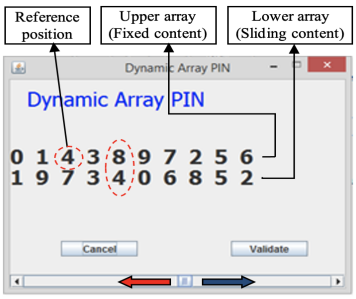

Fig. 1. The DAP Protocol Interface

- The user locates the position of the first digit of his PIN code in the upper array and reads the corresponding digit in the lower array. The user then finds the position of the digit read in the upper array and takes it as a reference to be used as explained in the other steps of the session. Fig. 1 shows an example: to enter the PIN code "8642", the user locates 8 in the upper array and reads the corresponding digit in the lower array, which, in this case, is 4. He then looks up the position of 4 in the upper array and considers the cell found (which is in the third position from the left) as the reference for this authentication session.
- Using the touchpad, the user slides the digits in the lower array to the left or to the right, and when the first PIN digit in the lower array aligns vertically with the reference position in the upper array, the user releases his finger to enter it. We used the horizontal scroll bar shown in Fig. 1 to simulate the ATM touchpad.
- The user repeats the previous step for each remaining digit of the PIN code. To complete this operation, the user presses the "Validate" button.

However, in our previous paper [23], we discovered that the DAP protocol is vulnerable to the intersection of multiple records (video eavesdropping or camera recording) attack where the attacker can steal the user's PIN code [22]. This attack is usually applied to discover the PIN entered by the user when using a smart technique (PIN digits are not entered directly). It is usually performed by a hidden camera that records the steps of the PIN-based authentication protocol.

III. New Proposed Solution: DCP

In order to protect the user against the intersection multiple records attack mentioned above (see also section I), in this section, we propose a new solution called Dynamic Cloud PIN

(DCP) which is an improvement for the DAP protocol. This solution improves the authentication mechanism on ATMs in order to strengthen the security system of NFC transactions with the ATM. It uses only the PIN code and a digit as secrets shared between the user and the bank server.

A. Solution Phases

To perform an NFC transaction with the ATM, our new solution includes the following phases:

1) Cloud Phase: it includes the following steps:
   - When the user is away from the ATM, he will use his own account by running an application installed on his smartphone that has been provided by the bank server. This application will communicate with the SE to obtain its encrypted identifier.
   - The server uses the encrypted identifier to reference the user's record in a database, then generates a randomly indistinguishable number from 0 to 9, stores it in the user's record in the database and sends it to the user's smartphone.
   - The server places the user's records in a cash memory to optimize the access time and sends a dynamically generated number to the user.
   - The user receives from the Cloud (online) the dynamic number (secured thanks to TLS) in his smartphone, stores it and destroys it.

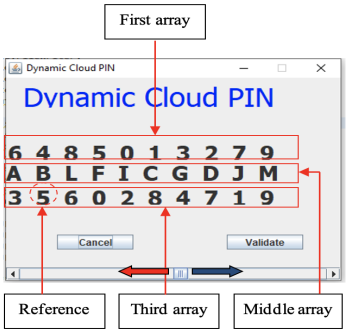

Fig. 2. The DCP Protocol Interface

2) Authentication phase: it includes the following steps:
   - The user runs the secure application that belongs to the bank server using his smartphone.
   - The user approaches his smartphone to the NFC reader of the ATM.
   - The smartphone sends encrypted information to the ATM: identifier of the SE, banking information stored in the SE or in the virtual bank card integrated in the smartphone.
   - The ATM sends the information received from the smartphone to the bank server.
   - The server looks for the appropriate record in the cash memory. If this record is not found, the protocol is interrupted and the transaction is ignored. Otherwise, the server will proceed to authenticate the user (the owner of the smartphone) by inviting him to enter the PIN code as described in the next steps.
   - The server generates three arrays, each one contains 10 randomly indistinguishable elements. The first one (the highest in Fig. 2) or the last (third) one (the lowest in Fig. 2) is composed of 10 digits numbered from 0 to 9 and they are displayed randomly. The position of each digit in each array is fixed during the authentication session. The middle array is composed of 10 alphabetical characters from A to Z. These characters are displayed in a random order where the position of each char is dynamically assigned during the authentication session as the user can drag the characters to the right or to the left.
   - The server sends the three arrays to the ATM.
   - The ATM displays the three arrays on the screen with the "Cancel" and "Validate" buttons. The characters of the middle array can be moved by the user to the right or to the left using a touchpad.
   - The user locates the position of the digit received online by his smartphone in the third array from the top (see Fig. 3). This position is called the reference.
   - In order to authenticate, the user must slide the character that corresponds to the reference into the middle array and release his finger each time the character coincides with a digit of the PIN code in the first array. Finally, he presses the "Validate" button to indicate the end of the PIN code entry.
   - When the "Validate" button is pressed, the ATM sends the user's data to the server.
   - The server compares the PIN code entered with the one retrieved from the database. If they are equal, then the user is well authenticated.

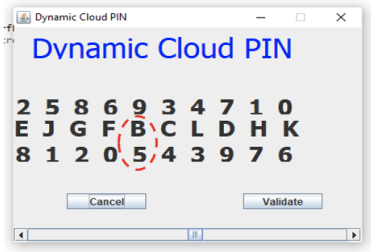

Fig. 3. The Reference in DCP Interface

B. Solution Example

In this section, we use an example to illustrate our solution. We assume that the dynamic number sent by the Cloud is "5" and that the PIN code is "1234". As shown in Fig. 3, to enter this PIN code, the user looks for the position of the digit "5" in the third array which represents the reference for this time (position five in the third array from the left). Then, using a small touchpad installed on the ATM, the user slides the middle array horizontally to the left or to the right and stops it by releasing his finger whenever a PIN digit in the first array matches the character in the middle array that corresponds to the reference in the third array (the character for entering the first digit is "B"). The X-axis scroll bar in Fig. 3 is used in our desktop implementation to simulate the ATM touchpad. the rest of the steps followed by the user are as follows:

- To enter the first digit of the PIN code, the user drags in our example the "B" until it becomes opposite the first digit ("1") in the first array as shown in Fig. 4.

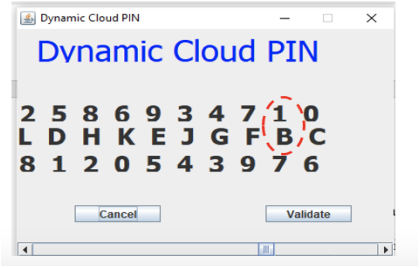

Fig. 4. Enter of the First Digit (‘1’)

- Then the user notices that the "E" is now opposite the reference ("5") in the third array, he must drag the "E" until it becomes opposite the second digit ("2" in our example) in the first array. This is illustrated in Fig. 5.

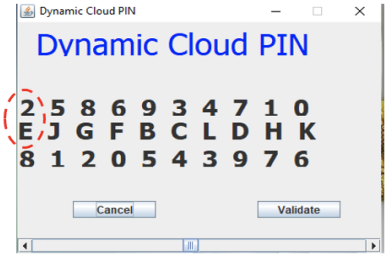

Fig. 5. Enter of the Second Digit (‘2’)

- Then the user notices that the "B" is now opposite the reference ("5") in the third array, he must drag the "B" until it becomes opposite the third digit ("3" in our example) in the first array. This is illustrated in Fig. 6.

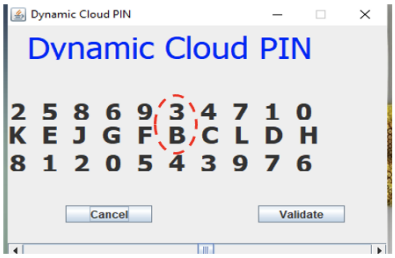

Fig. 6. Enter of the Third Digit (‘3’)

- Finally, the user notices that the "F" is now opposite the reference ("5") in the third array, so he must drag the "F" until it becomes opposite the fourth digit ("4" in our example) in the first array. This is illustrated in Fig. 7.
- To validate, the user clicks on the "Validate" button and obtains the status shown in Fig. 8.

Indeed, the basic idea of our proposed solution is that the attacker can see the user’s input, but he cannot reveal the PIN code because he cannot detect the reference in the third array that is the position of the dynamic digit shared between the server and the user ("5" in our example). The attacker also cannot detect the character in the middle array that corresponds to the reference, especially because the character changes after each entry of a PIN digit. The dynamic digit shared between the server and the user and its position in the third array are changeable during each authentication session and are unknown to the attacker.

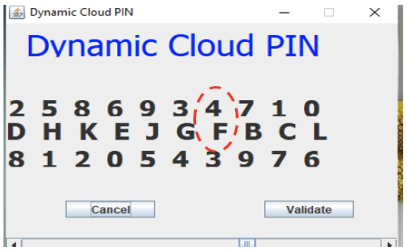

Fig. 7. Enter of the Fourth Digit (‘4’)

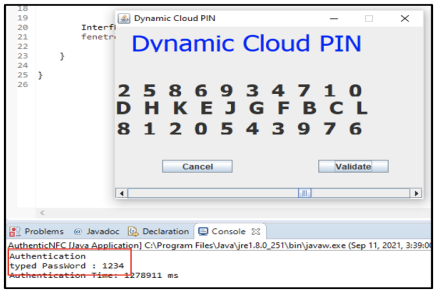

Fig. 8. The Result After the Click on Validate Button

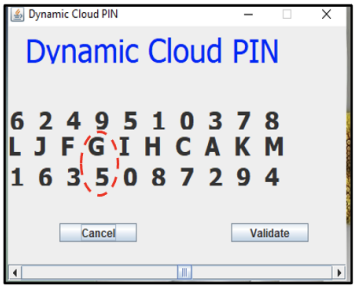

Fig. 9. The Start of DCP Application

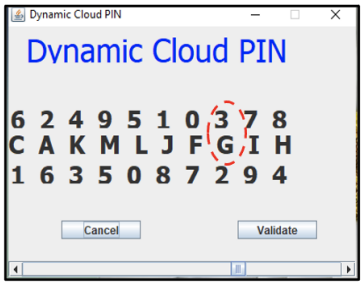

Fig. 10. The Enter of the First Digit

IV. Security Analysis

In this section, we prove that our proposal is protected against the multiple records intersection attack. To achieve this goal, we consider that an attacker, using a camera that records the authentication operation, is able to detect when the user’s finger is released after entering each digit. For example, we assume that the PIN code is "3582" and the shared dynamic

digit is "5". We assume that the DCP interface is as illustrated in Fig. 9. The rest of the steps used to analyze the security of our proposal are as follows:

- To enter the first digit of the PIN code ("3"), the user slides the character "G" under the digit "3" of the first array as shown in Fig. 10.
- The attacker records the state of the first and middle arrays just after the user's finger is released as shown in Fig. 11.

6	2	4	9	5	1	0	3	7	8
C	A	K	M	L	J	F	G	I	H

Fig. 11. Records Made by the Attacker (1)

- To enter the second digit of his PIN code ("5"), the user drags the character "M", that corresponds to the reference "5" in the third array, under the digit "5" in the first array as shown in Fig. 12.

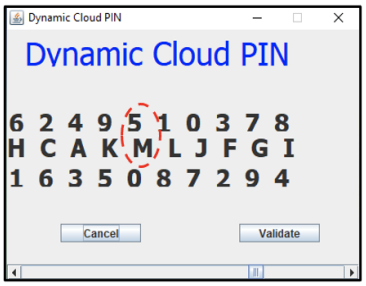

Fig. 12. The Enter of the Second Digit

- The attacker records the state of the first and middle arrays just after the user's finger is released as shown in Fig. 13.

6	2	4	9	5	1	0	3	7	8
H	C	A	K	M	L	J	F	G	I

Fig. 13. Records Made by the Attacker (2)

- To enter the third digit of his PIN code ("8"), the user drags the character "K", that corresponds to the reference "5" in the third array, under the digit "8" in the first array as illustrated in Fig. 14.

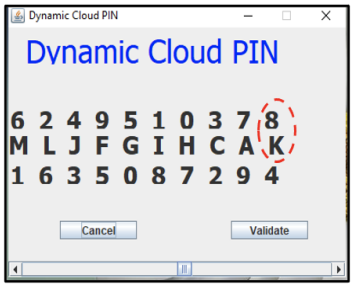

Fig. 14. The Enter of the Third Digit

- The attacker records the state of the first and middle arrays just after the user's finger is released as shown in Fig. 15.
- To enter the fourth and the last digit of his PIN code ("2"), the user drags the character "F", that corresponds to the reference "5" in the third array, under the digit "2" in the first array as illustrated in Fig. 16.

6	2	4	9	5	1	0	3	7	8
M	L	J	F	G	I	H	C	A	K

Fig. 15. Records Made by the Attacker (3)

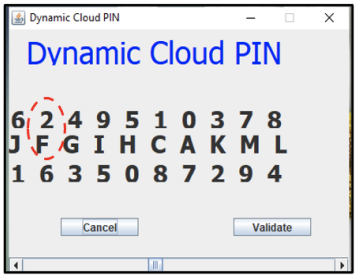

Fig. 16. The Enter of the Fourth Digit

- The attacker records the state of the first and middle arrays just after the user's finger is released as shown in Fig. 17.

6	2	4	9	5	1	0	3	7	8
J	F	G	I	H	C	A	K	M	L

Fig. 17. Records Made by the Attacker (4)

- Then, the user clicks on the "Validate" button and obtains the result illustrated in Fig. 18.

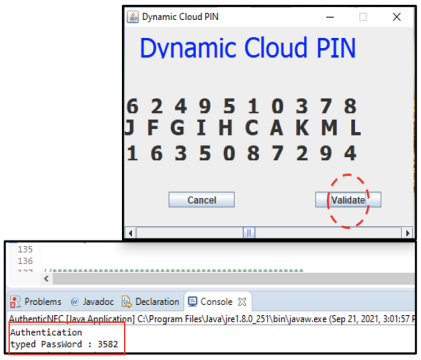

Fig. 18. The End of DCP Authentication

- Finally, the attacker groups the arrays that have been recorded to detect the PIN as illustrated in Fig. 19.

6	2	4	9	5	1	0	3	7	8
C	A	K	M	L	J	F	G	I	H
6	2	4	9	5	1	0	3	7	8
H	C	A	K	M	L	J	F	G	I
6	2	4	9	5	1	0	3	7	8
M	L	J	F	G	I	H	C	A	K
6	2	4	9	5	1	0	3	7	8
J	F	G	I	H	C	A	K	M	L

Fig. 19. Arrays Recorded by the Attacker

We can conclude that the intersection multiple records attack has no effect on our DCP protocol. This is because the attacker

is unable to fix a set of possible combinations representing the PIN code, thanks to the following:

1) The digit shared in the Cloud between the server and the user is a random secret (it changes at each session).
2) The positions of the digits in the first and last arrays are random at each session (they change at each session).
3) The characters displayed in the middle array are random at each session (they change at each session and knowing that they are 10 among 26 characters, this increases the number of probabilities).
4) Thanks to the latter, the character corresponding to the shared digit in the third array will be random at each session and will also change after each PIN digit is entered, which makes tracking too difficult.

V. Conclusion and Future Work

In this paper, we proposed a new security solution that improves the DAP protocol which is vulnerable to multiple eavesdropping video or camera records attack. Our proposal improves the PIN authentication mechanism on ATMs in order to strengthen the security system of NFC transactions with the ATM. We proved the security of our proposal by a demonstration where the attacker cannot find the correct PIN code by applying the intersection multiple records attack. In future work, we intend to experiment with our proposal to find a better balance between its increased security and increased authentication time. One possible improvement would be to relieve the user of the burden of searching for the character that matches the reference and dragging it to the PIN digit in the first array. We believe that this part could be automated without re-exposing the improved DCP to the multiple record attack.

References

[1] M. Badra and R. B. Badra, "A lightweight security protocol for nfc-based mobile payments," *Procedia Computer Science, Elsevier*, vol. 83, pp. 705–711, 2016.

[2] D. Giese, K. Liu, M. Sun, T. Syed, and L. Zhang, "Security analysis of near-field communication (nfc) payments," *arXiv preprint arXiv:1904.10623*, 2019.

[3] N. K. Gyamfi, M. A. Mohammed, K. Nuamah-Gyambra, F. Katsriku, and J.-D. Abdulah, "Enhancing the security features of automated teller machines (atms): A ghanaian perspective," *International Journal of Applied Science and Technology*, vol. 6, no. 1, 2016.

[4] J. Merkus, "Security evaluation of the nfc contactless payment protocol using model based testing," *(Master's thesis, Open University Nederland)*, 2018.

[5] E. Wadii, J. Boutahar, and S. Ghazi, "Nfc technology for contactless payment echosystems," *International Journal Of Advanced Computer Science And Applications*, vol. 8, no. 5, pp. 391–397, 2017.

[6] N. El Madhoun, E. Bertin, M. Badra, and G. Pujolle, "Towards more secure emv purchase transactions," *Annals of Telecommunications, Springer*, vol. 76, no. 3, pp. 203–222, 2021.

[7] T. I. Shammee, T. Akter, M. Mou, F. Chowdhury, and M. S. Ferdous, "A systematic literature review of graphical password schemes," *J. Comput. Sci. Eng*, vol. 14, pp. 163–185, 2020.

[8] S. A. Alsuhibany, "A camouflage text-based password approach for mobile devices against shoulder-surfing attack," *Security and Communication Networks, Hindawi*, 2021.

[9] D. Chen, Z. Zhao, X. Qin, Y. Luo, M. Cao, H. Xu, and A. Liu, "Magleak: a learning-based side-channel attack for password recognition with multiple sensors in iiot environment," *IEEE Transactions on Industrial Informatics, IEEE*, vol. 18, no. 1, pp. 467–476, 2020.

[10] S. Kurnaz, A. H. Mohammed *et al.*, "Secure pin authentication in java smart card using honey encryption," *2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), IEEE*, pp. 1–4, 2020.

[11] J. Shang and J. Wu, "Lightdefender: Protecting pin input using ambient light sensor," *2020 IEEE International Conference on Pervasive Computing and Communications (PerCom), IEEE*, pp. 1–10, 2020.

[12] H. Shin, S. Sim, H. Kwon, S. Hwang, and Y. Lee, "A new smart smudge attack using cnn," *International Journal of Information Security, Springer*, pp. 1–12, 2021.

[13] J. Shubhra, "Atm frauds: Detection & prevention," *International Journal of Advances in Electronics and Computer Science*, vol. 4, no. 10, 2017.

[14] M. Guerar, M. Migliardi, F. Palmieri, L. Verderame, and A. Merlo, "Securing pin-based authentication in smartwatches with just two gestures," *Concurrency and Computation: Practice and Experience, Wiley Online Library*, vol. 32, no. 18, p. e5549, 2020.

[15] K. Kobayashi, T. Oguni, and M. Nakagawa, "A series of pin/password input methods resilient to shoulder hacking based on cognitive difficulty of tracing multiple key movements," *IEICE TRANSACTIONS on Information and Systems, The Institute of Electronics, Information and Communication Engineers*, vol. 103, no. 7, pp. 1623–1632, 2020.

[16] P. M. Chanal and M. S. Kakkasageri, "Security and privacy in iot: a survey," *Wireless Personal Communications, Springer*, vol. 115, no. 2, pp. 1667–1693, 2020.

[17] *Smart payment association. Biometrics in Payment: Breaking down barriers with high value payments*, 2018 May.

[18] *Promontory an IBM Company. Biometric authentication in payments: Considerations for Policymakers*, 2017 November.

[19] K. Yadav, S. Mattas, L. Saini, and P. Jindal, "Secure card-less atm transactions," *2020 First IEEE International Conference on Measurement, Instrumentation, Control and Automation (ICMICA), IEEE*, pp. 1–4, 2020.

[20] *GSMA. NFC Functions and Security Certification overview*, 2018 May.

[21] S. Chabbi, R. Boudour, F. Semchedine, and D. Chefrour, "Dynamic array pin: A novel approach to secure nfc electronic payment between atm and smartphone," *Information Security Journal: A Global Perspective, Taylor & Francis*, vol. 29, no. 6, pp. 327–340, 2020.

[22] D. Nyang, H. Kim, W. Lee, S.-b. Kang, G. Cho, M.-K. Lee, and A. Mohaisen, "Two-thumbs-up: Physical protection for pin entry secure against recording attacks," *computers & security, Elsevier*, vol. 78, pp. 1–15, 2018.

[23] S. Chabbi and D. Chefrour, "Vulnerability of the dynamic array pin protocol." *Ingénierie des Systèmes d'Information*, vol. 27, no. 1, 2022.

[24] M. S. Deshpande *et al.*, "Multifactor authentication on mobile phones using existing brightpass," *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, vol. 12, no. 12, pp. 948–953, 2021.

[25] M. Guerar, M. Migliardi, A. Merlo, M. Benmohammed, F. Palmieri, and A. Castiglione, "Using screen brightness to improve security in mobile social network access," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 4, pp. 621–632, 2016.

[26] M. Guerar, M. Benmohammed, and V. Alimi, "Color wheel pin: Usable and resilient atm authentication," *Journal of High Speed Networks, IOS Press*, vol. 22, no. 3, pp. 231–240, 2016.