The EMV Payment System: Is It Reliable?

Nour El Madhoun, Emmanuel Bertin, Guy Pujolle

The EMV Payment System: Is It Reliable?

Nour El Madhoun*, Emmanuel Bertin†, Guy Pujolle‡
* LISITE Laboratory, ISEP, 10 Rue de Vanves 92130 Issy-les-Moulineaux, France
†Orange Labs, 42 rue des Coutures BP 6243 14066 Caen, France
‡Sorbonne Université, CNRS, LIP6, 4 place Jussieu 75005 Paris, France
Email: nour.el-madhoun@isep.fr; emmanuel.bertin@orange.com; guy.pujolle@lip6.fr

***Abstract*—EMV (Europay MasterCard Visa) is the technical international protocol implemented to secure the communication, between a client's payment device and a PoS (Point of Sale), during a contact or contactless-NFC purchase transaction. In this paper, we are interested in examining the reliability of this protocol.**

***Index Terms*—EMV protocol, EMV vulnerabilities, NFC technology, security.**

I. Introduction

The bank card is considered today as the fastest and most convenient means of payment in Europe according to the studies [1] [2] [3] [4]. It has much utility and advantages over cash and other payment methods because it [5] [6] [7] [8] [9]: includes insurance and assistance, is simple to obtain by banks, is small in size and easy to carry (we can keep it in a safe place to be protected), stores in a secure manner critical banking data that are needed primarily to:

– Perform online payments, make in-store purchases either by magnetic stripe, with contact or contactless based on the NFC (Near Field Communication) technology, deposit or withdraw cash money from an ATM (Automated Teller Machine).

Accordingly, the clients consider the bank card as a crucial and *"magical"* solution to safely manage their funds. In fact, they assume that this *"magic"* card is very secure and reliable because it securely stores the critical banking data in a smart chip, which is a sophisticated electronic system offering a highly secure environment [6]. However, authors in the studies [10] [11] [12] [13] and [14] prove that this assumption is not completely accurate by demonstrating the following attack: *"A malicious adversary can use an NFC reader to steal the banking data, remotely, from an NFC bank card, without the knowledge of the cardholder and without stealing the physical card"*. So, this kind of attack has raised our attention to ask the question: *(*) How this attack can be produced* ?

In this paper, we are interested to address the aforementioned question *(*)* and for this, it is mandatory to examine how the bank card can send in clear the banking data, using NFC radio waves, to an NFC reader. Indeed, it is the Europay MasterCard Visa (EMV) standard which is by default responsible to secure the communication, during a contact or an NFC payment transaction, between a client's payment device (a bank card or an NFC smartphone emulating a bank card) and a PoS (Point of Sale). Therefore, in order to answer the question *(*)*, we will examine the reliability and safety of the EMV standard by surveying a set of research papers that have identified several security vulnerabilities in this standard.

This paper is organized as follows. In section II, we present a survey of the EMV security vulnerabilities. In section III, we discuss the possible attacks due these vulnerabilities whereas in section IV, we illustrate three experimental tests that we have performed during this work. The last section concludes this paper.

II. EMV Security Vulnerabilities

According to the EMV specifications in [15]–[21], the EMV payment system includes five actors that communicate together in order to secure an EMV contact/contactless-NFC purchase transaction: Payment Scheme (*PS*), Issuing Bank (*IB*), Client's Payment Device (*C*), Acquiring Bank (*AB*) and a Point of Sale (*PoS*). The role of each EMV actor is described in our previous work [6] and [7]. Then, during a payment transaction, these EMV actors exchange with each other a sequence of security messages that are divided into four phases: Initialization (Phase 1), Authentication of *C* to *PoS* (Phase 2), Authentication of the Client (Phase 3) and, Actual Transaction (Phase 4). Each phase is explained in detail in our previous work [5], [6], [7] and [22].

In this section, we examine the reliability of the EMV protocol by presenting the most well-known security vulnerabilities that have been identified in literature for each EMV phase [23] [24]. We note that for any detail or information concerning the EMV phases or EMV actors, it is essential to consult the references [5], [6], [7] and [22].

A. In Phase "Initialization"

- $Vulnerability\ (1)$:
  To make any *C* device compatible with any *PoS* device, the EMV primary negotiation phase (initialization) is required. The messages exchanged in this phase are neither signed nor enciphered by *PoS* or by *C*. Consequently, the studies [25] and [26] show that this phase presents an important vulnerability in the EMV standard, where it is possible for an attacker (man-in-the-middle) to exploit this vulnerability and modify the abilities of *C* or *PoS* to put *PoS* in a weak position. This type of attack is called $Downgrade\ Attack$. See section III-A: $1st$, $2nd$, $9th$ $Attacks$.
- $Vulnerability\ (2)$:

The studies [10] [11] [13] [12] [14] show that the EMV protocol fails to guarantee two important security properties in the EMV phase 1 during a contact or an NFC purchase transaction. Therefore, two security vulnerabilities are detected:

- $Vulnerability\ (2.1)$: the confidentiality of the banking data is not ensured: the PAN (Primary Account Number) and *ExpDate* (Expiration date) are sent in clear from *C* to *PoS*.
- $Vulnerability\ (2.2)$: with the detection of the $Vulnerability\ (2.1)$, also the authentication of *PoS* is not ensured to *C*. The latter can answer any device without authenticating it, by sending the banking data in clear.

Indeed, only the authentication of *C* to *PoS* which is well ensured in the EMV phase 2 (if it is executed). Consequently, an attacker can exploit the $Vulnerabilities\ (2.1)$ and (2.2) to steal the banking data (see $3rd$, $4th$, $5th\ Attacks$, in section III-A) and use them to harm the victims (see $6th$, $7th$, $8th\ Attacks$, in section III-A).

B. In Phase "Authentication of C to PoS"

- $Vulnerability\ (3)$:
  Because of the $Vulnerability\ (1)$ in the initialization phase, the attack which is called "YES cards" attack may still be applicable at certain offline *PoS* devices despite the integration of DDA (Dynamic Data Authentication)/CDA (Combined Data Authentication) methods to prevent cloning of the SDA (Static Data Authentication) signature [14] [25] [26]. In section III-A, the $9th\ Attack$ illustrates this case of vulnerability.

C. In Phase "Authentication of the Client"

- $Vulnerability\ (4)$:
  In the EMV phase 3, the response of *C*, indicating if the PIN (Personal Identification Number) code entered is correct or not, is not authenticated by *PoS*. Authors in the study [27] show that this lack of authentication is a vulnerability in the EMV phase 3 and demonstrate the $10th\ Attack$ exploiting it: see section III-A. Indeed, the $10th\ Attack$ cannot be detected in the EMV phase 4 neither by *C*, nor by *PoS* nor by *IB*, which leads to a new vulnerability in this phase (see next section II-D).

D. In Phase "Actual Transaction"

- $Vulnerability\ (5)$:
  In the same study [27] that presents $Vulnerability\ (4)$ and the $10th\ Attack$, authors show that this attack cannot be detected in the actual transaction phase because of another EMV vulnerability that is: an ambiguity in the encoding TVR (Terminal Verification Results) and IAD (Issuer Application Data). Indeed, the TVR and the IAD contain the results of the EMV phase 3 respectively carried out by *PoS* and *C*. *C* includes the TVR and the IAD in the cryptogram 'TC (Transaction Certificate) or ARQC (Authorization ReQuest Cryptogram)' intended to '*PoS* or *IB*'. According to EMV specifications [15] [16] [17] [18] [19]:
  - In the TVR, the byte of phase 3 is set, by *PoS*, to 1 only if the PIN verification has been attempted and has failed.
  - In the IAD, the byte of phase 3 is set, by *C*, to 1 only if the PIN verification has been attempted.

  However, the TVR and the IAD do not define their bytes to 1 in the cases: PIN verification has been successful, PIN verification has not been attempted, the handwritten signature has been chosen. Therefore, in order to better understand this vulnerability, see the $10th\ Attack$.

III. Possible Attacks

A. Attacks Due to EMV Vulnerabilities

In fact, due to the EMV vulnerabilities, many attacks are possible. Therefore, in this section, we discuss some attacks for each EMV vulnerability.

- Because of the $Vulnerability\ (1)$:
  - $1st\ Attack$: According to EMV specifications [15] [16] [17] [18] [19], an NFC purchase transaction should not be accepted if an attacker uses a cloned NFC card, because the original secret key of the original NFC bank card cannot be copied. However, in the work [25], authors demonstrate that an attacker who happens to clone an NFC bank card (without copying the original secret key), can modify the capabilities of the cloned NFC card to fool *PoS* into executing a magnetic stripe transaction (used in the United States) rather than an NFC transaction [14]. We specify that this attack is possible only if *C* is a bank card (from MasterCard) and not an NFC smartphone.
  - $2nd\ Attack$: in the research work [26], authors show that an attacker can convince *PoS* that the chosen CVM (Cardholder Verification Method) is *the offline plaintext PIN* rather than *the offline enciphered PIN*. Then , he can easily record the PIN code.
- Because of the $Vulnerability\ (2)$:
  - $3rd\ Attack$: by default, for a contact or an NFC purchase transaction, the professional *PoS* (big merchants) prints two payment proofs: one proof for the client containing the PAN and *ExpDate* truncated for safety and security reasons, the second proof for the merchant containing the PAN and *ExpDate* **in clear**, which serves to ensure the traceability of the client's purchase transaction in the future [7]. However, a thief can easily steal the merchant's receipts and obtain the banking data of several clients insofar as the merchants [28]: - generally do not protect these proofs conscientiously, - will not need these proofs anymore, every 12-13 months, and they will be able to throw them away. In section IV-A, we will illustrate the experiment that we have done in our work to show the content of receipts printed by a *PoS* device.

- $4th\ Attack$: in the case of small merchants, the mobile *PoS* is less secure than the professional *PoS*, because during a contact or an NFC purchase transaction, the merchant's smartphone receives the banking data from the mobile *PoS* without encryption: android's attackers can then retrieve the banking data [29] [30].
- $5th\ Attack$: recently, authors in [31] show an attack by brute force where it is possible to easily and quickly obtain the *ExpDate* and the security code using only the PAN. They carried out this attack first to obtain the *ExpDate* from the PAN as follows: by using a website that does not ask for the security code and does not block this type of brute force attack, they tested all possible combinations month/year taking into account 5 years as a maximum duration of validity (12 months * 5 years = 60 tests). Once the *ExpDate* is known, they proceeded to the second step to retrieve in the same manner the security code. For this, they used other websites that request the PAN, *ExpDate* and the security code. Then, they reviewed all possibilities for the security code, knowing that there are only one thousand: 000 to 999. This operation can be relatively fast if it is not blocked by servers.
- $6th\ Attack$: as seen in [6] [12] [14], an attacker can use the PAN and *ExpDate* to make fraudulent payments on the internet without needing to provide the security code: many websites such as "www.amazon.com", "www.armaniexchange.com", "www.zappos.com" do not request the security code. Additionally, if the attacker manages to obtain the security code as in the 3rd attack, he can then use it on websites that require the security code. In fact, in this type of attack, it is assumed that: the victim has funds in his bank account, the merchant of the website does not use any additional security mechanism as the 3D secure [32] and that the attacker is able to recover his parcels on postal addresses where it will be difficult for the police to follow him. We specify that the cardholder name is never checked by websites and then the attacker can use a random name. In section IV-B, we will show an example of a purchase made on a website that does not verify the name and does not require a security code.
- $7th\ Attack$: an attacker can duplicate the bank card if he manages to obtain the banking data (as shown in the $3rd$, $4th$ and $5th\ Attacks$). Then, he may fraudulently use the duplicated card as in the case of the 1st attack.
- $8th\ Attack$: in fact, an attacker can identify and track the client using the PAN which is by default a sensitive information [6].

- Because of $Vulnerabilities\ (1)$ and (3)
  - $9th\ Attack$: in 2006, when all banks cards only supported the SDA authentication method (there were no smartphones for payment), authors in [33] illustrated that it was possible to create cloned SDA bank cards, because the signature provided by the original bank card is static and it is the same for each EMV transaction. A cloned SDA card can be used to perform purchase transactions with an offline *PoS*. Consequently, a cloned SDA card can also be programmed to support the CVM *'offline plain-text PIN'* and to respond with "YES" to any PIN entered by the attacker [21]: the name given to the cloned SDA cards is "YES cards". Indeed, since 2013, most SDA cards have integrated DDA and CDA as authentication methods because they do not allow the cloning of cards. However, despite the integration of DDA/CDA methods, authors in [14] [25] and [26] demonstrate that an attacker can select in the case of an offline *PoS*, which cannot communicate with the banking network, the SDA method and fool both *PoS* and *C* in order to execute the "YES cards" attack.
- Because of $Vulnerabilities\ (4)$ and (5)
  - $10th\ Attack$: an attacker can authorize an EMV purchase transaction by entering an incorrect PIN code in the offline PIN case in the EMV phase 3. He then modifies this phase as follows [27]:
    - $>$ By first sending *PoS* a message indicating that the PIN entered (by the attacker) is well verified by *C* and that it is correct.
    - $>$ Then, by informing *C* that the transaction is verified by a handwritten signature and that no PIN is required.

    In fact, this attack cannot be detected in the actual transaction phase because of: *PoS* believes that the PIN verification has been successful and so it will generate a zero byte in the TVR. *C* believes that the PIN verification has not been attempted but a handwritten signature has been required, so it will accept the zero byte from the TVR and generates a zero byte in the IAD. Finally, neither party can identify the inconsistency between the TVR and the IAD. *IB* will also consider that *PoS* has not been able to solicit a PIN code and that the handwritten signature has been chosen as a CVM.

    In 2012, in France, several criminals were arrested because they had exploited the EMV $Vulnerabilities\ (4)$ and (5) and performed 6000 fraudulent purchase transactions mounting up to more than 500000 euros [14].
- Another Attack:
  - $11th\ Attack$: authors in [34] illustrate that it is theoretically possible to falsify the EMV authentication signatures: DDA or CDA. This attack is not practical because it is necessary to execute 4,639 partial transactions by accessing *C* to generate the falsified DDA/CDA signature. Also, each transaction lasts about 500ms and therefore, in order to achieve this attack, one needs to have access to *C* for 38 minutes [14].

B. Attacks Due to NFC Technology

- By default, NFC technology enables contactless communication between two devices within a short distance 5-10cm [35]. Firstly, before introducing NFC technology into payment applications, the EMV protocol has been designed only to secure contact payments. After the integration of NFC technology, EMV protocol has been adapted without any change for contactless payments. Indeed, EMVCo has assumed that an NFC purchase transaction cannot exceed the short distance of 5-10cm. However, authors in [10] [11] [12] [13] [14] [36] [37] [38], confirm that this assumption is very weak by demonstrating the following attacks:
  - $12th$ *Attack*: a relay attack is presented in [14] [36] [37] [38] where it is possible to perform an NFC purchase transaction using an NFC bank card *C* which is at a distance of several kilometers from *PoS*. Additionally, the attacker has the possibility to eavesdrop this NFC communication during the NFC purchase transaction, and is able to retrieve the banking data that are sent from *C* to *PoS* without encryption because of $Vulnerabilities$ (2.1) and (2.2) (see section II-A). Afterwards, the attacker can use the retrieved data to do a brute force attack or to harm the victim (see $5th$, $6th$, $7th$, $8th$ $Attacks$ in section III-A).
  - $13th$ *Attack* (which is the same attack presented in section I): a skilled attacker in radio-electronics, can attach an amplifier to the NFC antenna of an unauthenticated NFC reader (NFC smartphone, NFC tablet, etc.) in order to reach a distance of NFC reading up to 1.50 meters. The attacker can then remotely steal the banking data (PAN, *ExpDate*) from several bank cards, even if the latter are in the bag, without the knowledge of the victims and without stealing the physical cards [10] [11] [12] [13] [14]. In response to the question *(*)* asked in section I, this attack is also due because of $Vulnerabilities$ (2.1) and (2.2) and the attacker can do a brute force attack or harm the victims as seen in the $5th$, $6th$, $7th$, $8th$ $Attacks$ discussed in section III-A. In section IV-C, we will illustrate an example of reading banking data using an NFC smartphone.
- According to EMV specifications in [19] for NFC payment applications, an NFC purchase transaction can be performed without the need to enter a PIN code if the amount is less than 30 Euros in France. If the transaction amount exceeds the maximum authorized limit, *PoS* then asks the client to enter his PIN code. However, this EMV specification only corresponds to the local currency of *C* and authors in the experimental study [39] demonstrate the following attack:
  - $14th$ *Attack*: an attacker can use a French NFC bank card to execute contactless purchases without entering a PIN code for any amount in a foreign currency: in dollars or pounds for example.

IV. Tests and Diagnostics

In this work, we have performed three experimental tests illustrating the EMV $Vulnerabilities$ $\{(2.1), (2.2)\}$.

A. 1st Test: Content of Payment Receipts

As presented in the $3rd$ $Attack$ in section III-A, the client's receipt contains the PAN and *ExpDate* truncated, whereas the merchant's receipt contains the PAN and the *ExpDate* **in clear**. In Figure 1, we illustrate an experiment that we have done in agreement with a merchant who agreed to give us the receipts printed by his *PoS*. The banking data are marked by a red rectangle. In the client's receipt, we can notice that the banking data are truncated whilst in the merchant's receipt they are in clear (the blue band is to hide our data).

Fig. 1. *PoS* Proofs for Contact/NFC Purchase Transactions

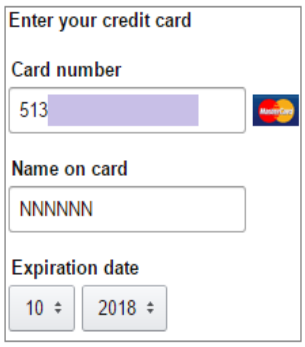

Fig. 2. Adding Banking Data (Amazon) [6]

B. 2nd Test: Website Without Security Code

In the $6th$ $Attack$ in section III-A, we have seen that an attacker can perform fraudulent payment transactions on the internet without the need to provide the security code and the exact cardholder's name. In Figure 2, we show a test that we performed on the Amazon website. We only

added the banking data: PAN, *ExpDate* and a random name to "www.amazon.com" and we succeeded in buying many items.

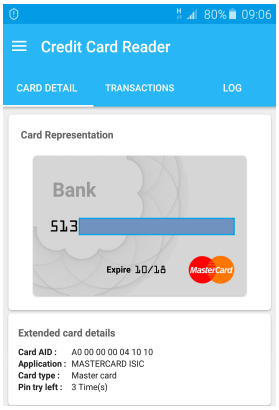

Fig. 3. Data of an NFC Bank Card Read by an NFC Smartphone [6]

C. 3rd Test: Reading Banking Data

In section III-B, we referred that the assumption of EMVCo in the case of NFC payment is very weak, where in the $13th$ *Attack* it is possible for an attacker to remotely steal the banking data from several bank cards by using an unauthenticated NFC reader. In Figure 3, we were able to read the PAN and *ExpDate* of a personal NFC bank card by using an NFC smartphone *'Samsung S5'* with a free Android application called NFC card reader [40]. We performed this test over a distance of 1 cm because the antenna of our smartphone is not efficient. Indeed, it is sufficient to attach a signal amplifier to this smartphone to reach a distance of 1.50 meters.

V. Conclusion

EMV vulnerabilities represent major risks for our day to day safety. The attacks presented in this paper threaten and harm clients and merchants with the loss of their revenues: the *C* device of a victim client can be fraudulently used without his knowledge, and a merchant can sell his products while there are no funds to receive. One of our objectives in this paper is to answer to the question *(*)* asked in section I. In response, the attack introduced in section I (which is the same as the $13th$ *Attack*) is due to the $Vulnerabilities$ (2.1) and (2.2) (see section II-A). These vulnerabilities lead also to several attacks and dangerous risks: the $3rd$, $4th$, $5th$, $6th$, $7th$, $8th$ and $12th$ *Attacks*.

References

[1] Etude de l'institut de sondages d'opinion CSA, "Les français et les moyens de paiement," *https://www.economie.gouv.fr/files/sondagecsa_synthese.pdf*, last connection (30/04/2018).

[2] Féderation bancaire française, "Les moyens de paiement," *http://www.fbf.fr/fr/files/AC3CBC/Les%20Moyens%20de%20Paiement.pdf*, last connection (30/04/2018).

[3] La finance pour tous, "La carte bancaire," *http://www.lafinancepourtous.com/Banque-au-quotidien/Moyens-de-paiement/La-carte-bancaire/*, last connection (30/04/2018).

[4] Delphine Cuny, "Carte, virement, chèque ou cash : comment paie-t-on en europe ?" *https://www.latribune.fr/entreprises-finance/banques-finance/carte-virement-cheque-ou-cash-comment-paie-t-on-en-europe-750879.html*, last connection (30/04/2018).

[5] N. El Madhoun and G. Pujolle, "Security enhancements in emv protocol for nfc mobile payment," *The 15th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (IEEE TrustCom-16)*, pp. 1889–1895, 2016.

[6] N. El Madhoun and E. Bertin, "Magic always comes with a price: Utility versus security for bank cards," *The 1st Cyber Security in Networking Conference (CSNet'17), IEEE*, pp. 1–7, 2017.

[7] N. El Madhoun, E. Bertin, and G. Pujolle, "An overview of the emv protocol and its security vulnerabilities," *The Fourth International Conference on Mobile and Secure Services (MobiSecServ), IEEE*, pp. 1–5, 2018.

[8] NFC Technology, "History of near field communication," *http://nearfieldcommunication.org/history-nfc.html*, last connection (30/04/2018).

[9] Centre National RFID, "Introduction au nfc," *http://www.centrenational-rfid.com/introduction-au-nfc-article-132-fr-ruid-17.html*, last connection (30/04/2018).

[10] M. Emms and A. van Moorsel, "Practical attack on contactless payment cards," *HCI2011 Workshop Heath, Wealth and Identity Theft*, 2011.

[11] Benjamin Cohen, "Millions of barclays card users exposed to fraud," *https://www.channel4.com/news/millions-of-barclays-card-users-exposed-to-fraud*, 2012, last connection (30/04/2018).

[12] R. Lifchitz, "Hacking the nfc credit cards for fun and debit," *Hackito Ergo Sum conference*, April 2012.

[13] Gerard Tubb, "Contactless cards: App reveals security risk," *https://news.sky.com/story/contactless-cards-app-reveals-security-risk-10443980*, 2013, last connection (30/04/2018).

[14] M. J. Emms, "Contactless payments: usability at the cost of security?" *Ph.D.Thesis, Newcastle University*, 2016.

[15] EMV - Integrated Circuit Card Specifications for Payment Systems,, *Book 1: Application Independent ICC to Terminal Interface Requirements, Version 4.3, EMVCo, November*, 2011.

[16] ——, *Book 2: Security and Key Management, Version 4.3, EMVCo, November*, 2011.

[17] ——, *Book 3: Application Specification, Version 4.3, EMVCo, November*, 2011.

[18] ——, *Book 4: Cardholder, Attendant, and Acquirer Interface Requirements, Version 4.3, EMVCo, November*, 2011.

[19] EMV - Level 1 Specifications for Payment Systems,, *EMV Contactless Interface Specification, Version 3.0, EMVCo, February*, 2018.

[20] J. De Ruiter and E. Poll, "Formal analysis of the emv protocol suite," *Springer Theory of Security and Applications*, pp. 113–129, 2012.

[21] J. van den Breekel, D. A. Ortiz-Yepes, E. Poll, and J. de Ruiter, "Emv in a nutshell," *Technical Report*, 2016.

[22] N. El Madhoun, E. Bertin, and G. Pujolle, "For small merchants: A secure smartphone-based architecture to process and accept nfc payments," pp. 403–411, 2018.

[23] K. Shrikrishna, N. N. Kumar, and R. Shyamasundar, "Security analysis of emv protocol and approaches for strengthening it," *International Conference on Distributed Computing and Internet Technology, Springer*, pp. 69–85, 2018.

[24] D. Singh, R. Ruhl, and H. Samuel, "Attack tree for modelling unauthorized emv card transactions at pos terminals," *4th International Conference on Information Systems Security and Privacy*, pp. 494–502, 2018.

[25] M. Roland and J. Langer, "Cloning credit cards: A combined pre-play and downgrade attack on emv contactless," *WOOT - 7th USENIX conference on Offensive Technologies*, 2013.

[26] A. Barisani, D. Bianco, A. Laurie, and Z. Franken, "Chip and pin is definitely broken," *Presentation at CanSecWest Applied Security Conference, Vancouver*, 2011.

[27] S. J. Murdoch, S. Drimer, R. Anderson, and M. Bond, "Chip and pin is broken," *IEEE Symposium on Security and Privacy*, pp. 433–446, 2010.

[28] Les experts Ooreka, "Ticket de carte bancaire," *https://carte-bancaire.ooreka.fr/astuce/voir/515831/ticket-de-carte-bancaire*, last connection (30/04/2018).

[29] T. Luo, H. Hao, W. Du, Y. Wang, and H. Yin, "Attacks on webview in the android system," *Proceedings of the 27th Annual Computer Security Applications Conference, ACM*, pp. 343–352, 2011.

[30] L. Davi, A. Dmitrienko, A.-R. Sadeghi, and M. Winandy, "Privilege escalation attacks on android," *International Conference on Information Security, Springer*, pp. 346–360, 2010.
[31] M. A. Ali, B. Arief, M. Emms, and A. van Moorsel, "Does the online card payment landscape unwittingly facilitate fraud?" *IEEE Security & Privacy*, pp. 78–86, 2017.
[32] Y. Li and Z. Ying, "The developing tendency of electronic commerce payment: 3d-secure technology [j]," *Sci-Tech Information Development & Economy*, vol. 14, 2007.
[33] R. Anderson, M. Bond, and S. J. Murdoch, "Chip and spin," *Computer Security Journal*, vol. 22, pp. 1–6, 2006.
[34] J. P. Degabriele, A. Lehmann, K. G. Paterson, N. P. Smart, and M. Strefler, "On the joint security of encryption and signature in emv," *Cryptographers' Track at the RSA Conference, Springer*, pp. 116–135, 2012.
[35] ISO 14443 Contactless Integrated Circuit Cards,, *International Standards Organisation*, 2011.
[36] L. Francis, G. Hancke, K. Mayes, and K. Markantonakis, "Potential misuse of nfc enabled mobile phones with embedded security elements as contactless attack platforms," *International Conference for Internet Technology and Secured Transactions (ICITST), IEEE*, pp. 1–8, 2009.
[37] K. Markantonakis, L. Francis, G. Hancke, and K. Mayes, "Practical relay attack on contactless transactions by using nfc mobile phones," *Radio Frequency Identification System Security: RFIDsec*, vol. 12, p. 21, 2012.
[38] M. Emms, L. Freitas, and A. van Moorsel, "Rigorous design and implementation of an emulator for emv contactless payments," *Computing Science, Newcastle University*, 2014.
[39] M. Emms, B. Arief, L. Freitas, J. Hannon, and A. van Moorsel, "Harvesting high value foreign currency transactions from emv contactless credit cards without the pin," *Proceedings of ACM SIGSAC Conference on Computer and Communications Security*, pp. 716–726, 2014.
[40] Julien MILLAU, "Nfc credit card reader (lecteur de carte bancaire nfc)," *https://play.google.com/store/apps/details?id=com.github.devnied.emvnfccard*, last connection (30/04/2018).