Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 13/05/2026 10:15:05

Annales UMCS Informatica AI XIV, 2 (2014) 46–59
DOI: 10.2478/umcsinfo-2014-0006

Annales UMCS
Informatica
Lublin-Polonia
Sectio AI
http://www.annales.umcs.lublin.pl/

One-Time Code Cardholder Verification Method in Electronic Funds Transfer Transactions

Albert Sitek[1]*

[1] *Institute of Telecommunications, Warsaw University of Technology, Nowowiejska 15/19, 00-665 Warsaw, Poland*

Abstract – Card payments are getting more and more popular across the world. The dominant standard used for Electronic Funds Transfer transaction is EMV. It is widely used across Europe and Canada, and currently it is being introduced in the USA. The most frequently used Cardholder Verification Method in EMV transaction is PIN, which requires from the payment terminal to be equipped with pinpad which increases the cost of the whole payment device. In this article I present an alternative Cardholder Verification Method (CVM) that can be used instead of traditional PIN. The key advantage of the presented mechanism is that it can be easily implemented in currently utilized authorization protocols, it does not affect rules of EMV specification and may decrease time of transaction processing.

1 Introduction

Smart cards have almost completely replaced traditional magstripe cards, but not in the USA. Magstripe cards are still very popular there, which has direct impact on level of frauds. It is the only region in the world, where volume of counterfeit card frauds is constantly growing [**1**]. Because of this kind of frauds, the USA issuers accounted for over 26% of global fraud losses in 2013. This is the main reason why there is very high pressure from banks to introduce EMV in the USA. As we can imagine, the whole process will be very costly, so there are still contentious discussions how to make this transition and how the technology standard will be implemented in the PoS (Point of Sale) [**2**]. There are several obstacles to be overcome. Americans are used to making the transactions with the magstripe card verified by signature, and probably it will be hard for them to get used to paying by chip card and entering the PIN

*asitek@tele.pw.edu.pl

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 13/05/2026 10:15:05

code. Luckily, EMV specification allows to verify a cardholder via the signature. On the other hand, switching to EMV will dramatically increase costs of payment system maintenance. Whereas the cost of a single magstripe card is about \$0.25, the cost of a smart card can vary from \$1.25 to \$2.50, depending on card capabilities [**2**]. The setup of the whole EMV-based payment system will also be very expensive. The cost of the standard magstripe reader is about \$20, while a smart card reader can cost about \$40 (without pinpad), and can run up to \$100 for those equipped with pinpad [**2**]. The cost of pinpad is so high because such device must be compliant with various strict security standards like PCI DSS [**3**]. As we may see, it is worth resigning from introducing pinpads at points of sales in new installations. Such approach will dramatically reduce costs of the whole payment system and will speed up rollout of EMV based solutions. The key issue is how to resign from pinpads and do not decrease level of transaction security this can be achieved by utilizing the context-based security management approach [**17, 20, 22, 24**]. In this paper I propose new Customer Verification Method that is based on one time codes entered in the standard keyboard (or any other input device), that is not the PCI DSS compliant because it would not handle sensitive data. The decision to perform Cardholder Verification or not will be made dynamically by the central authorization system that analyzes various context factors gathered from history and present transaction details. The presented approach can even increase transaction security and reduce average transaction processing time.

2 Payment System Architecture

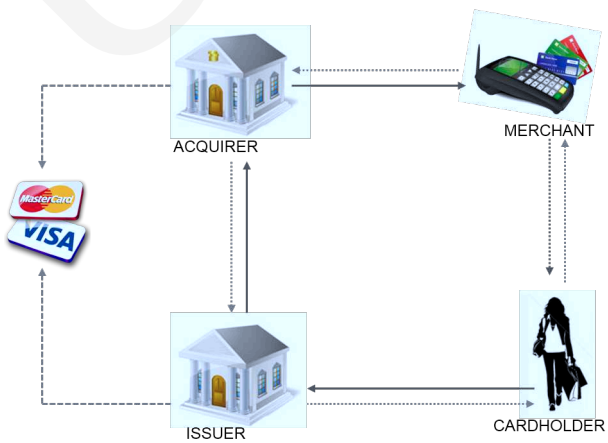

Fig. 1. Payment system architecture four-party card scheme

The standard four-party card scheme is depicted in Fig. 1. Each party can be described as follows:

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 13/05/2026 10:15:05

- Cardholder – a person who wants to buy some goods or services and pays by card,
- Merchant – a person who sells goods or services and operates payment terminal to accept card transactions,
- Acquirer – an institution that processes payment transaction. It is forwarding the authorization request to the Issuer and sends authorization response back to the terminal,
- Issuer – its an institution (bank) that issued the Cardholders card. After receiving the authorization request, it checks if the transaction can be authorized – by checking cardholders account balance and (in the case of EMV transaction) by validating transaction cryptogram generated by the card. It also generates the authorization response and sends it back to the Acquirer.

There are also some fees that are charged from the transaction amount:

- Merchant Service Charge,
- Interchange Fees (IC Fees) – can vary from the transaction type and circumstances under which the transaction has been performed (if there was a fallback to magstripe or not etc.).

The communication between Acquirer and Issuer is processed via MasterCard, Visa or other card company network (depends on what card was used). The communication between Acquirer and Merchants terminal is performed via special authorization protocol and depends on the Acquirer. More information about the authorization protocols can be found in Section 3.

3 Authorization Protocols

The authorization protocol is used in order to communicate between Merchants terminal and Acquirer. It is mainly used for:

- Exchanging authorization requests,
- Performing settlement – usually at the end of day,
- Performing network diagnostics,
- Transferring configuration to the terminal.

For instance, the most popular authorization protocol in Poland is ISO 8583. Its customized versions are used for example by:

- Elavon – one of the most popular acquirer in Europe and USA
- eService – the biggest Polish acquirer, currently exploring European countries

Short description of this protocol can be found in Section 3.1. The other, known for the author, authorization protocols are:

- SPDH – used for example by FirstData Poland,
- EP2 – the most popular protocol in Swiss and neighboring countries, used for example by SIX Payments Services.

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 13/05/2026 10:15:05

3.1 ISO 8583

The whole standard is defined in ISO 8583 Financial transaction card originated messages Interchange message specifications. It is the binary protocol defined by International Organization of Standardization and intended for the systems that exchange electronic transactions made by cardholders using payment cards. ISO 8583 defines a communication flow and a message format so that different systems can exchange these transaction requests and responses [**5**]. Fig. 1 presents the structure of ISO 8583 message.

Fig. 2. Structure of ISO 8583 message

MTI Message type indicator is a 4 digit numeric field which defines the high level function of the message. MTI includes the ISO 8583 version, the Message Class, the Message Function and the Message Origin. For example MTI equal to 0100 means:

- 0xxx - version of ISO 8583 (1987 version),
- x1xx - Authorization Message,
- xx0x Request,
- xxx0 Acquirer.

Detailed description of MTI can be found in [**5**].

Bitmap field indicates which data elements are present in the other part of the message. For example: Bitmap equal to 4210001102C04804 indicates that fields 2, 7, 12, 28, 32, 39, 41, 42, 50, 53, 62 will be present in the message, because 0x42 = 01000010, so bits 2 and 7 are set in byte 1 etc.

After bitmap each field carries transaction information. The main fields in ISO 8583 messages are as follows:

- 4 Transaction amount,
- 7 Transaction date and time,
- 11 Transaction sequence number,
- 35 Track2 data,
- 39 Response code,
- 64 Message Authentication code.

There are also some fields (61-63) reserved for private usage. They are usually utilized by Acquirers for:

- Value Added Services,
- Time Synchronization,
- Stocktaking.

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 13/05/2026 10:15:05

Those private fields can be utilized for the presented new Cardholder Verification Method in order to send one time code to the terminal.

4 EMV Security Mechanisms

The EMV standard (named after Europay, Visa and MasterCard) introduced many security mechanisms that rapidly decreased volume of fraudulent transactions. In fact, the goal of introducing EMV was to shift the costs of dispute from the issuing bank in the following way:

- If a disputed transaction has been authorized by a signature, it would be charged to the merchant,
- If a disputed transaction has been authorized by a PIN then it would be charged to the customer.

It other words, the banking industry, which designed the whole system, carries less liability for the fraud. This is also called the liability shift [**6**].

4.1 EMV Transaction Steps

Each EMV transaction consists of several processing steps:

- Application Selection – each smart card can have a few payment applications available. Application may be selected manually by the cardholder – from the displayed list, or may be selected automatically by the terminal – the decision is made based on priorities assigned to each application,
- Application Initialization – the terminal sends certain information to the card in order to decide if transaction processing should be continued. If yes, then the card responds to its processing capabilities (AIP) and application file location (AFL), indicating from where to read application data,
- Application Data Read – in this step the terminal reads application data from location pointed by AFL. The terminal retrieves all data necessary to perform payment transaction. Card controls data that can be read – for example it is not possible to read PIN value. In other words it is not possible to clone a smart card
- Card Data Authentication – this step is intended to verify if the presented card in not a counterfeit one. All possible methods to be performed in this step were described in Section 4.2,
- Processing Restrictions – the terminal checks if the requested service is allowed, if the application has not expired etc.,
- Cardholder Verification – the terminal verifies if the person who presents the card is the genuine cardholder. The whole Cardholder Verification step is described in detail in Section 4.3,

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 13/05/2026 10:15:05

- Terminal Risk Management – during this step the terminal can decide if the transaction should go online or not. The decision is made based on the transaction amount (if the so called floor limit were exceeded), transaction history (how many transactions has been authorized offline before) etc.,
- Terminal Action Analysis – the result of the previous steps is saved in the so called Terminal Verification Result – TVR. It is a bitmap that stores various information – for example if the application is expired, if the floor limit is exceeded etc. Its value is compared with Terminal Action Codes (TAC – retrieved from the terminal configuration), and IAC (Issuer Action Codes – retrieved from the card). This comparison can indicate that the transaction should be offline declined, offline approved, or must go online. The result is sent to the card in the GENERATE AC command which requests from the card to generate the application cryptogram. Possible cryptogram types are described in Section 5,
- Card Action Analysis – in this step the card performs its own issuer specific analysis. The card can change the terminal decision only to a stricter one. The card decision is indicated by the type of cryptogram generated in response for GENERATE AC command,
- Online Processing – this step is performed only if the generated cryptogram was Authorization Request Cryptogram (ARQC). In that case the generated cryptogram is sent to the issuer host for authorization. The issuer sends back its Authorization Response Cryptogram (ARPC) which is forwarded to the card in order to make a final authorization decision. Even if the host authorized the transaction, the card can decline it if it failed to authenticate the received cryptogram,
- Issuer Script Processing – this step is optional and is performed only when the issuer sends its script with the authorization response. Such scripts allow to update the card data, for example unlock or change the PIN code, block the card etc.

As can be seen, the EMV standard specifies a lot of security mechanisms that drive transaction processing in a certain way. The whole Risk Management and the Action Analysis procedures are performed on the terminal and card side, so only a few context factors can be taken into account. The presented approach assumes that the key Risk Management processing is performed on the issuer host side, so a lot of context factors can be freely analyzed and a more accurate decision can be made.

4.2 Card Authentication Methods

All card authentication methods (CAM) provided by the EMV standard are based on the asymmetric cryptography - RSA. There are three possible card authentication methods to be performed during the EMV transaction:

- SDA – Static Data Authentication - requires the merchant terminal to verify the digital signature of static card data computed by the issuer. The SDA

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 13/05/2026 10:15:05

cards are not able to perform the RSA calculation, so they are cheap, but this type of cards is vulnerable to a trivial and well-known replay attack in which the certificate is read from a card and written to a counterfeit one (these are often called yes cards because they will respond yes to any entered PIN during the PIN verification phase) [**6**],

- DDA Dynamic Data Authentication in this method the terminal verifies a dynamic digital signature generated by the card over the unpredictable number sent by the terminal. As this algorithm is dynamic, it is better than SDA but it also requires that the card has an RSA private key along with the RSA processing capability so those cards are more expensive than the SDA ones [**7**],
- CDA -Combined Data Authentication - this method can be performed on the cards supporting DDA, and assumes that the terminal verifies a digital signature computed by the card over the full transaction data (including the symmetric cryptogram that can be verified by the issuer). CDA is better than DDA because CDA protects against the use of so-called wedges that, if undetected, could fraudulently modify the transaction data exchanged between the card and the terminal.

4.3 Cardholder Verification Method

During the EMV card personalization each card is equipped with the so-called CVM List. This list presents the issue choice of supported CVMs ordered by priority. This list also indicates what should happen if this failed to perform the current CVM method to go to the next CVM from the list, to decline the transaction etc. Because different types of terminals support different CVMs, multiple CVMs enable the EMV card to be accepted at as many merchant terminals as possible. The card and the terminal use the first matching CVM type for transaction authorization [**8**].

There are a few possible Cardholder Verification Methods defined by the EMV standard:

- Offline Plaintext PIN in this case PIN is entered on the pinpad and sent directly to the card in the plaintext form. Then the card responds if PIN is correct or not. Such PIN number is not send to the authorization host. It is worth mentioning that if this failed to verify Offline PIN then the terminal can switch to another CVM method regarding the CVM List,
- Offline Enciphered PIN this method is similar to Offline Plaintext PIN, but PIN is submitted to the card enciphered under the card public key. This type of CVM can be performed only by cards supporting DDA because of mandatory RSA calculation,
- Online PIN in this method PIN is entered on the pinpad and sent directly to the authorization host in authorization request. The PIN is validated by the issuer and proper response is sent to the terminal. PIN is encrypted using 3DES with the (usually) unique key per each transaction. In order to derive

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 13/05/2026 10:15:05

the unique key, a few key management schemes like DUKPT [**9**] or Master/Session have been proposed. There are also proprietary key management schemes like for example in EP2 [**10**],

- Signature This Cardholder Verification Method is performed always after online of offline authorization. In this case the merchant verifies if the signature on the transaction receipt matches that on the card. If the merchant affirms that the signature is not valid, the automatic reversal for the current transaction is generated and the decline receipt is printed on the terminal. This CVM method can be only performed on the attended terminals,
- NoCVM in this case, the transaction is authorized without the PIN or the signature. It is usually utilized for low-amount transactions, and it is the fastest way to complete the transaction.

As mentioned before, Cardholder Verification Method during the EMV transaction is agreed by the terminal in cooperation with the card. In the EMV terminal configuration there is a special parameter called the CVM limit. This parameter indicates the amount of transaction above which it cannot be authorized with NoCVM. In particular, if this parameter is set to 0, every transaction will require a kind of PIN or signature. There are also terminals (usually unattended ones) that require NoCVM for every transaction because they are not equipped with pinpad and there is no merchant to validate the signature.

Non implemented Cardholder Verification Methods Recently researchers have focused on introducing the biometric-based Cardholder Verification Methods [**15, 16**]. As smart cards are getting more and more powerful such approach is quite reasonable. Those methods assume that during the Cardholder Verification step some biometric data will be gathered from the Cardholder, converted to a special format and sent to the card for validation. The following biometric information can be verified:

- Fingerprint
- Face shape
- Voice
- Signature dynamics
- Iris pattern
- Hand geometry
- Keystroke

Unfortunately, none of those Cardholder Verification Methods have been implemented in real payment system. This can happen because of high costs of biometric devices.

4.4 Attacks on EMV Security

In 2010 a group of researchers from Cambridge proved that stolen cards can be successfully used without knowing PIN code [**6**]. It can happen only when Offline PIN is used. In that case after PIN is entered, the Verify command is sent to the card. Card simply responds with OK or NOK answer, that is not tied to any data nor secured

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 13/05/2026 10:15:05

by any signature. An attack assumes that the whole communication between the card and the terminal is routed via the proxy device that blocks the Verify command and always responds with OK to the terminal. In that situation the terminal thinks that Offline PIN was performed, and the card thinks that NoCVM was forced.

Another interesting attack was proposed by Adam Laurie, Andrea Barisani, Daniele Bianco and Zac Franken and it is called the CVM Downgrade attack [**11**]. This attack assumes that the CVM List sent by the card is changed by the proxy device in order to force the Offline PIN verification. Such change may cause Offline Data Authentication to fail, but they proved that it may not lead to decline the transaction (because they have also changed IAC Issuer Action Codes [1]). Moreover, in most cases such modified transaction will be successfully authorized by the Issuer.

5 Proposed Solution

Nowadays almost everyone has its own mobile phone and always carries it in his pocket. Mobile phones are used not only for texting and voice call, but also for example for authentication. One time passwords sent via SMS or generated from the dedicated application are currently very frequently used for confirmation of online fund transfers. The presented approach assumes that the transaction is confirmed by entering one time code if the issuers system decides to perform Customer Verification. The whole diagram of transaction that involves the new Context-based Cardholder Verification Method is depicted in Fig. 3.

The transaction flow can be described as follows:

- Payment terminal is configured to accept only NoCVM transactions,
- After the amount and card entry steps, the transaction process regarding the NoCVM rules. The results of such processing can be:
  - Offline Approved Transaction Certificate (TC) has been generated by the card,
  - Offline Decline - Application Authentication Cryptogram (AAC) has been generated by the card,
  - Request for online authorization Authorization Request Cryptogram (ARQC) has been generated by the card,
- Let us assume that the card requested for online authorization. In that case Authorization Request is generated by the terminal and sent via the Acquirer to the Issuer,
- The Issuer checks if the card is active, sufficient funds are present on the Cardholders account and if the ARQC cryptogram is genuine. If any of those conditions fail, the transaction is declined and the proper Authorization Request is sent back to the terminal,

[1] More information about EMV processing can be found in [**13**] and [**14**]

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 13/05/2026 10:15:05

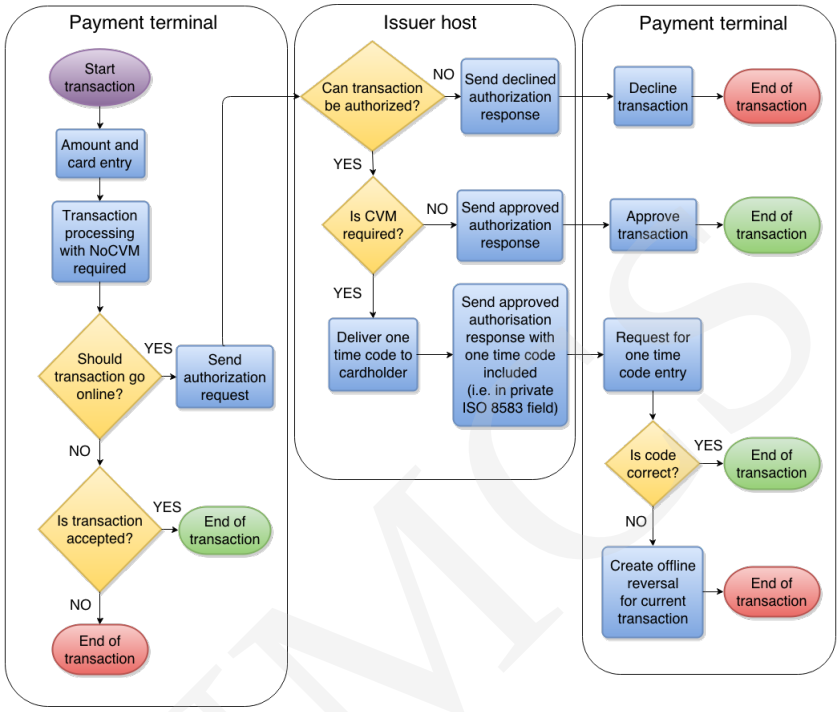

FIG. 3. Transaction flow diagram involving Context-based Cardholder Verification Method

- Let us assume that checks from the previous step went OK. Next, the Issuer runs its risk management algorithm to decide if the transaction can be authorized with NoCVM, or if one time code entry will be required. The algorithm can take into account the following context factors:
  - Amount of transaction,
  - Time of transaction,
  - Point of Sale localization,
  - Cardholder transaction history,
  - Cardholder reputation (if there were some disputes in the past).
- If the Issuer decides to authorize the transaction without CVM, then the standard Authorization Response is sent back to the terminal,
- If the Issuer decides to use one time code verification, then a proper code is sent somehow (i.e. in the private field 61 ISO 8583 protocol message) to the terminal and at the same time delivered to the Cardholders mobile device,
- When the terminal receives such Authorization Response, it prompts the cardholder to enter its one time code on the standard keyboard, touchscreen, or any non-secure input device,
- If the entered code matches the received one in Authorization Response, the transaction is finally accepted,

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 13/05/2026 10:15:05

- If the entered code does not match the received one in Authorization Response, the terminal generates the offline reversal of current transaction, and the cardholder receives the printout of the declined transaction.

5.1 Security Analysis

In the presented approach the processed transaction is fully compliant with the EMV standard, it means that:

- Card authentication is verified,
- In the case of Contact EMV card, the issuer authentication is verified.

Moreover, our approach is not prone to attacks described in chapter 5.3 because:

- It prevents from using Offline PIN,
- It utilizes the less secure CVM defined by EMV, so the CVM List downgrade attack is not applicable.

The behaviour of payment terminal during one time code verification will be similar to that during the signature check, it means that:

- If one time code is invalid, then the transaction is automatically declined,
- If there is a cut in the power supply on one time code verification step, the non-verified transaction will be automatically reversed just after the next terminal boot.

What is more, the signature verification is prone to accidental acceptance (because there is only simple question: Is signature valid? with possible answers YES or NO). In our approach, one time code is only known by the terminal application, so the only way to accept the transaction is to enter valid one time code. Also communication between the terminal and the Acquirer is secured usually with TLS, so there is no chance to eavesdrop on one time code during the transmission.

Paradoxically, our solution can increase security of low-amount transactions. Due to the fact that the decision about CVM is made by the Issuer, fraudulent transactions can be revealed earlier and CVM can be requested also for low-amount transaction which is not possible in the standard EMV approach.

One time code can be delivered to the Cardholders device for example via:

- SMS,
- Dedicated application that will download one time code from the Issuers server,
- Dedicated application that will generate RSA tokens.

In order to unlock mobile device usually there is a need to enter PIN code, draw special pattern on the screen and so on. Because of that, in the case of stolen or lost device, the security of the whole solution still holds high level. We must consider that in such cases fraudulent transactions are possible only till the Cardholder blocks his card.

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 13/05/2026 10:15:05

6 Summary and Future Work

The key thing in introducing any change in such big systems as Payment Systems is to utilize as many present mechanisms as it is possible. Introducing any big change does not make sense because it will be not profitable and will take a lot of time to rollout. In this paper I presented the new Context-based Cardholder Verification Method which can be easily adapted to the present Payment Systems and it utilizes mechanisms provided by EMV specification. What is more, the presented CVM reduces costs of payment devices (because it does not use a pinpad) which could be a real accelerator for rollouts in new countries like the USA. Thanks to the risk management algorithm running on the Issuer side, there is possibility to optimize the transaction processing time ask for CVM only if there is real need to do so. The decision about CVM will be made based on transaction context, so the result will be accurate.

Moreover, contrary to signature verification, the presented Cardholder Verification Method can be used in unattended solutions, which are getting more and more popular across the world. Such solutions are exposed to acts of vandalisms and hard weather conditions, so in fact they are more expensive than the attended ones. Utilizing the presented CVM method can also reduce price of unattended devices.

The next step in the future research could be design and validation of the risk management algorithm running on the Issuer side. It is also worth to prove, that currently configured in the payment terminals CVM limit (equal to 50 PLN in Poland) could be easily increased without major impact on the transaction security. Such approach could allow to reduce average transaction processing time diametrically, because large majority of authorized transactions have the amount less than 100 PLN.

Another interesting topic for future research could be deployment of one time codes.

References

[1] Extract from The Nielsen Report - global Credit, Debit, and Prepaid Card Fraud Losses Reach $11.27 Billion in 2012 - Up 14.6% Over 2011 http://www.businesswire.com/news/home/20130819005953/en/Global-Credit-Debit-Prepaid-Card-Fraud-Losses#.U4Y3Ryj69SE

[2] Groenfeldt T. American Credit Cards Improving Security With EMV, At Last, http://www.forbes.com/sites/tomgroenfeldt/2014/01/28/american-credit-cards-improving-security-with-emv-at-last/

[3] Payment Card Industry (PCI) Data Security Standard, Requirements and Security Assessment version 3.0

[4] Berat, C. Cross-border Interchange Fees: Why the General Court Got it Wrong in the MasterCard v. Commission Case http://www.europeanpaymentscouncil.eu/index.cfm/newsletter/article/?articles_uuid=567705BA-5056-B741-DB26318F233469A0

[5] ISO 8583 protocol description, en.wikipedia.org/wiki/ISO_8583

[6] Murdoch S.J., Drimer S., Anderson R., Bond M., Chip and PIN is Broken, IEEE Symposium on Security and Privacy (2010).

[7] Ward M., EMV card payments An update, information security technical report II (2006): 89–92.

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 13/05/2026 10:15:05

[8] TSYS People-Centered payments Cardholder Verification Method - Considerations In A Changing Payments Landscape, http://www.tsys.com/acquiring/engage/white-papers/Cardholder-Verification-Method.cfm
[9] ANSI, ANS X9.24-1:2009 Retail Financial Services Symmetric Key Management Part 1: Using Symmetric Techniques
[10] SIX Payments, EP2 Terminal Security - Technical Requirements
[11] Laurie A., Barisani A., Bianco D., Franken Z., CVM Downgrade Attack
[12] Laurie A., Barisani A., Bianco D., Franken Z., Chip & PIN is definitely broken, Credit Card skimming and PIN harvesting in an EMV world, Inverse Path S.r.l. (2011).
[13] EMVCo, EMV Integrated Circuit Card Specifications for Payment Systems Book 3: Application Specification v. 4.3 (2011).
[14] EMVCo, EMV Integrated Circuit Card Specifications for Payment Systems Book 4: Cardholder, Attendant, and Acquirer Interface Requirements v. 4.3 (2011).
[15] Sanchez-Reillo R., ETSI Telecommun., Ciudad Univ., Spain, Securing information and operations in a smart card through biometrics Security Technology, 2000. Proceedings. IEEE 34th Annual 2000 International Carnahan Conference
[16] Struif B., Use of Biometrics for User Verification in Electronic Signature Smartcards, LNCS 2140 (2001): 220–227.
[17] Ksiezopolski B., Kotulski Z., Adaptable security mechanism for dynamic environments, Computers & Security 26(3) (2007): 246–255.
[18] Gerstel O., Sasaki G., Quality of Protection (QoP): A Quantitative Unifying Paradigm to Protection Service Grades, OptiComm 2001, Optical Networking and Communication Conference (2001).
[19] Jovanovikj V., Gabrijelcic D., Klobucar T., A conceptual model of security context, International Journal of Information Security (2014).
[20] Baldauf M., Dustdar S., Rosenberg F., A survey on context-aware systems, Int. J. Ad Hoc and Ubiquitous Computing 2(4) (2007): 263–277.
[21] Hayashi E., Das S., Shahriyar A., Owusu E., Han J., Hong J., Oakley I., Perrig A., Zhang J., CASA: A Framework for Context-Aware Scalable Authentication, SOUPS'13: Ninth Symposium on Usable Privacy and Secrecy (2013).
[22] Wrona K., Gomez L., Context-aware security and secure context-awareness in ubiquitous computing environments, Annales UMCS Informatica AI 4 (2006): 332–348.
[23] Siljee B., Bosloper I., Nijhuis J., A Classification Framework for Storage and Retrieval of Context, KI-04 Workshop on Modelling and Retrieval of Context, CEUR 114 (2004).
[24] Chen H., An intelligent broker architecture for context-aware systems, A PhD. Dissertation Proposal in Computer Science at the University of Maryland, Baltimore County (2003).
[25] Smirnov A., Pashkin M., Chilov N., Levashova T., Operational Decision Support: Context-Based Approach and Technological Framework, Proceedings of the 5th International and Interdisciplinary Conference CENTEXT (2005).