Extending EMV payment smart cards with biometric on-card verification

Olaf Henniger[1] and Dimitar Nikolov[2]

[1] Fraunhofer Institute for Computer Graphics Research IGD
Fraunhoferstr. 5, D-64283 Darmstadt, Germany
olaf.henniger@igd.fraunhofer.de
[2] dsnikolov@gmx.de

Abstract. Nowadays, many bank cards are smart cards (i.e. integrated-circuit cards) based on the EMV specifications for payment systems. This paper specifies how biometric on-card verification can be integrated into EMV debit and credit cards in a backwards-compatible way. The biometric verification does not change the EMV transaction flow outside the cardholder-verification step. The proposed payment system has been prototyped using Java cards and an applet for handwritten signature on-card verification.

1 Motivation

Debit and credit cards used to be magnetic-stripe cards. Today, however, a growing number are smart cards equipped with a microprocessor chip in addition to the magnetic stripe. This significantly improves the capabilities for authenticating the card and the cardholder and enables new protocols for securing payment processes. In the absence of an online connection to the banking network, the card is able to represent the card-issuing bank and to authorise payments on its behalf.

For confirming a debit transaction the cardholder must enter a secret personal identification number (PIN). Unlike magnetic-stripe card based debit transactions, not every smart-card based debit transaction requires a paid-for online connection. The PIN entered may be verified offline against the PIN stored in the chip and, in case of a match, the card may authorise the payment. An online account validation and online transaction authorisation are carried out only under certain conditions determined by the card issuer, e.g. if the credit limit stored on the card has been used up or if more time than permitted has passed since the last online connection.

Some payment methods require the cardholder to confirm the payment with a handwritten signature on a slip of paper (e.g. credit-card payment and a form of direct debit transaction popular in Germany called “elektronisches Lastschriftverfahren”). These payment methods do not take advantage of the opportunities that the microprocessor on card offers for raising the confidence in cardholder authentication. The cashier may only visually compare the image of a given signature with the signature image on the back of the card. The handwritten signature dynamics could be verified inside the smart-card chip just as the PIN is verified inside the chip in case of offline PIN verification. This would make the comparison more objective and improve security provided that the required levels of attack resistance and usability are achieved.

The authorisation of payment transactions is not the only field of application of biometric on-card verification. Biometric on-card verification has been proposed before for protecting other smart card functions access to which should be restricted to the legitimate cardholder [1]. Instead of handwritten signatures also other biometric characteristics could be compared on bank cards in order to improve the binding of payments to the cardholder. However, the major advantage of handwritten signatures over other biometric characteristics is that people are used to presenting their signatures and that signatures are evidence of deliberate decisions.

The EMV specifications [2] (which are named after the organisations Europay, Mastercard, and Visa, who created the first version) specify, based on [3], requirements and building blocks for smart-card based payment systems. The EMV specifications are the basis for several EMV-compliant payment systems. Section 2 reviews how EMV transactions work. International standards [4, 5] specify several approaches how to achieve personal verification through biometric methods, but has not been integrated into the EMV specifications yet. The challenge is to extend the EMV specifications in a way that is in compliance with the requirements and side conditions imposed by [3, 4]. The main contribution of this work is to specify how biometric on-card verification can be integrated into EMV payment smart cards in a backwards-compatible way. Section 3 extends the cardholder-verification process of the EMV specifications with biometric on-card verification of the cardholder. Figure 1 illustrates the sequence of steps to be carried out for offline EMV transactions on a point-of-sale (POS) terminal equipped with a biometric capture device. Section 4 discusses security and usability issues. Our prototype of the proposed payment system is described in Section 5. Section 6 summarises the results and gives an outlook.

2 EMV transactions

2.1 Overview

An EMV transaction requires that both, the debit card of the customer who wishes to pay for some goods and the POS terminal, conform to the EMV specifications. An EMV transaction begins with the insertion of the card into the terminal. Afterwards, the terminal determines what applications are installed on the card and the cardholder selects a payment method. The end result of the process is a cryptogram issued by the card. This cryptogram is a transaction certificate (TC) in case of an accepted and authorised transaction or an application authentication cryptogram (AAC) if the transaction is rejected for some reason.

In [6] an EMV transaction is described as consisting of the following steps:

Initiate application processing: This step
- Informs the smart card that the processing of a new transaction is beginning,
- Exchanges transaction-related information between terminal and smart card,
- Determines whether the transaction can be processed.

Read application data: The terminal reads the data from the card that are needed to process the transaction.

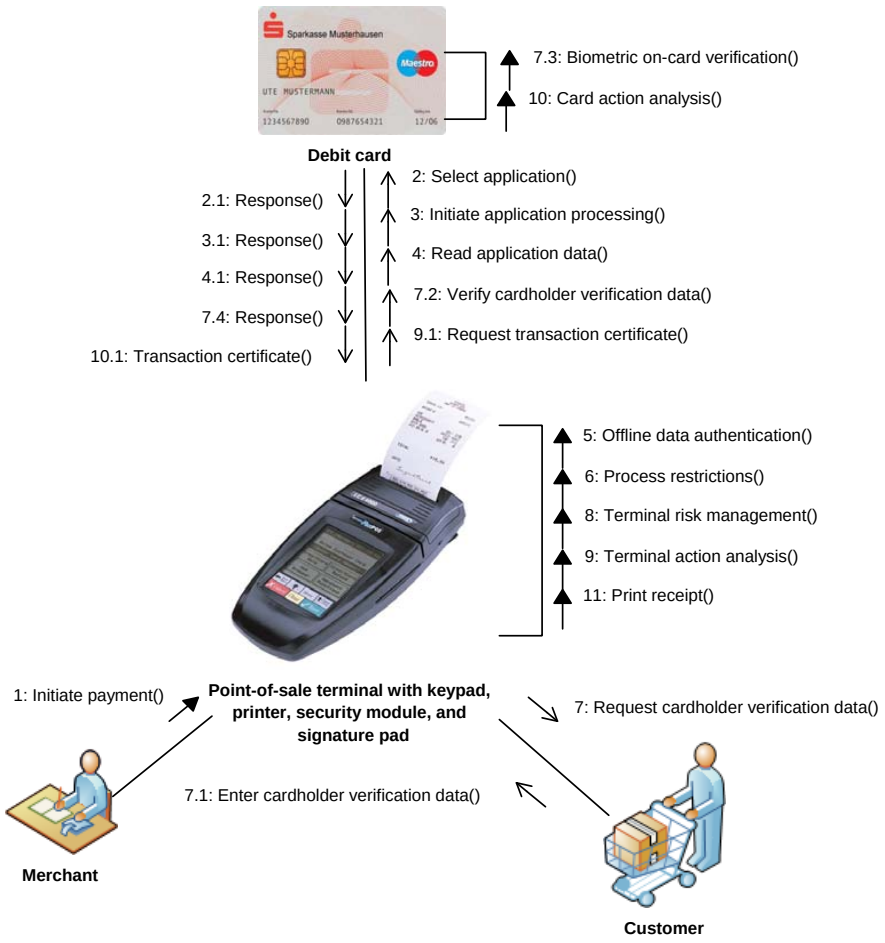

Fig. 1. Communication diagram for offline EMV transactions with biometric on-card verification

Offline data authentication: This step is performed if both the terminal and the card support offline checking of the validity and integrity of the data stored on the chip. Different methods based on public/private key pairs are defined for this purpose:

- The static data authentication method verifies the digital signatures of the card-issuing bank on data stored in the card;
- The dynamic data authentication methods preclude counterfeiting of EMV cards by not only verifying the digital signatures on static data, but also verifying signatures created by the card on challenges received from the terminal.

Processing restrictions: The terminal determines the degree of compatibility of the application in the terminal with the application in the smart card and makes nec-

essary adjustments, possibly rejecting the transaction. It is possible to restrict the application geographically or only to certain types of transactions.

Cardholder verification: This step is performed to ensure that the person presenting the smart card is the person to whom the card was issued. It is up to the card issuer to choose which cardholder verification methods (CVMs) to apply under what conditions. [6] provides for the following CVMs or combinations thereof:

- Offline PIN,
- Online PIN, and
- Handwritten signature on paper.

Section 3 of this paper proposes to add biometric on-card verification to this list.

Terminal risk management: This is the portion of risk management performed by the POS terminal to protect the acquiring and card-issuing banks from fraud. It provides positive issuer authorisation for high-value transactions and ensures going online regularly to protect against threats that are undetectable in an offline environment.

Terminal action analysis: The terminal takes a first decision as to whether the transaction should be approved offline, declined offline, or transmitted online.

- If the decision is to proceed offline, the terminal sends a GENERATE AC command asking the smart card to return a TC. However, the smart card may return an authorisation request cryptogram (ARQC) or an AAC instead of a TC as a result of card action analysis.
- If the decision is to reject the transaction, the terminal sends a GENERATE AC command asking the smart card for an AAC.
- If the decision is to go online, the terminal sends a GENERATE AC command asking the smart card for an ARQC. The ARQC is a cryptogram generated by the card from transaction data using an issuer key stored in the card and known in the issuer authorisation system.

Card action analysis: The smart card performs risk management on behalf of the card-issuing bank to protect the card-issuing bank from fraud or excessive credit risk. Details of the risk management algorithms are specific to the card issuer and outside the scope of the EMV specifications. The smart card may decide to complete a transaction online or offline or to reject the transaction. The smart card may also decide that an advice message should be sent to the card issuer to inform the issuer of an exceptional condition.

Online/offline decision: Depending on the card's answer to the previous GENERATE AC command, the terminal decides whether to continue the processing online.

Online processing: This step may be performed to ensure that the card issuer can review and authorise or reject transactions that are outside acceptable limits of risk defined by the card issuer, the payment system, or the acquirer. The ARQC generated by the card is sent to the card issuer in an authorisation request message. The issuer uses his key to authenticate the ARQC and thereby to authenticate the card. The authorisation response message sent from the issuer authorisation system to the terminal may contain issuer authentication data. This is a cryptogram generated using an issuer key from selected data included in the authorisation response or already known to the card. The terminal forwards the issuer authentication data to the smart card in an EXTERNAL AUTHENTICATE command or a second GENERATE AC command. The smart card may use the issuer authentication data to authenticate that the response message originated from the issuer.

Issuer-to-card script processing: A card issuer may provide command scripts to be delivered to the card by the terminal to perform functions that are important for the continued functioning of the application in the card, e.g. unblocking an offline PIN.

Completion: This step closes the processing of a transaction. The smart card indicates willingness to complete transaction processing by returning either a TC or an AAC to either the first or second GENERATE AC command issued by the terminal. If the terminal decides to go online, completion is done when the second GENERATE AC command is issued.

2.2 Details of cardholder verification

The terminal uses a CVM list read from the card to determine the cardholder verification method to be performed. A CVM list is a composite data object consisting of:

- X value,
- Y value,
- Cardholder verification (CV) rules list.

The X and Y values are threshold amounts in the application currency of the card application that can be used by the CV rules. The CV rules list is a variable-length list of two-byte data elements. Each CV rule describes a CVM and the conditions under which that CVM is to be applied.

The terminal processes the card's CV rules list entry by entry attempting to perform each applicable CVM until either

- A CVM is performed successfully,
- A CVM required to be performed successfully is performed unsuccessfully,
- The CV rule requires the cardholder verification to fail under the given conditions,
- Or the CV rules list is exhausted.

If a CVM is performed successfully, EMV transaction processing continues with the next step. Otherwise, the cardholder verification is unsuccessful and transaction processing is terminated.

3 Proposed extensions for biometric cardholder verification

The proposed extensions to the EMV specifications concern only the cardholder verification. From among the options for personal verification through biometric methods described in [4], for reasons of compatibility we have chosen that option that imposes the least changes to the EMV specifications. Given the fact that EMV payment systems are already widely used, the more extensive the changes, the less acceptable they would be. The main ideas of the proposed changes are the following:

1. Extend the commands used for PIN processing (CHANGE REFERENCE DATA, VERIFY, and GET DATA) to support biometric on-card verification by introducing new allowed values for the command parameters P1/P2 and introducing command chaining [7] for CHANGE REFERENCE DATA and VERIFY.

2. Extend the terminal verification result (TVR) and the associated terminal action code (TAC) and issuer action code (IAC) data elements in the data elements dictionary of [6] by one byte to hold information about biometric cardholder verification.
3. Extend the data elements dictionary of [6] by adding the following data elements to support biometrics:
   Biometric information template (BIT): The BIT contains information about the mode and format of biometric data and possibly further information [4, 8]. Prior to biometric cardholder verification, the BIT for on-card verification is retrieved in order to inform the terminal application about properties of the biometric on-card verification method. The terminal application uses the format owner and format type identifiers from the BIT for identifying the required format of the biometric probe. The comparison algorithm parameters in the BIT provide special parameters of the biometric on-card verification algorithm, e.g. the maximum number of minutiae expected in a biometric probe in case of finger minutiae data or the maximum number of sample points in case of handwritten signature time series data. If the card holds several biometric reference data objects, then a group BIT containing several BITs is used to describe the kind of biometric data to be sent to the card.
   Biometric reference: The biometric reference is one or more stored biometric samples, biometric templates or biometric models attributed to the cardholder and used for biometric comparison.
   Biometric retry counter: The offline PIN in the EMV specifications is associated with a retry counter indicating the number of remaining allowed PIN verification attempts. Its initial value indicates the supported maximum number of PIN verification attempts. A similar counter is needed for every biometric verification method as well.
4. Extend the CVM processing to include biometric cardholder verification consisting of the following steps:
   (a) Read the biometric retry counter and if the CVM is not blocked, continue.
   (b) Retrieve the BIT using a GET DATA command and if it is understandable and supported, continue.
   (c) If the biometric capture device is operational, continue.
   (d) Attempt to capture the biometric data and if successful, format them according to the BIT, and send them to the card within a VERIFY command and get the result of the on-card verification.
   (e) Continue depending on the obtained result.

Figure 2 shows the biometric CVM processing flow in the POS terminal. The diagram is built following the example of the PIN CVM processing flow diagram of [6].

4 Security and usability issues

To cut costs, retailers like using direct debit transactions without online authorisation, only requiring a handwritten signature of the customer on a slip of paper. However, such payments do not establish a guarantee of payment. In case of fraud, the cardholder

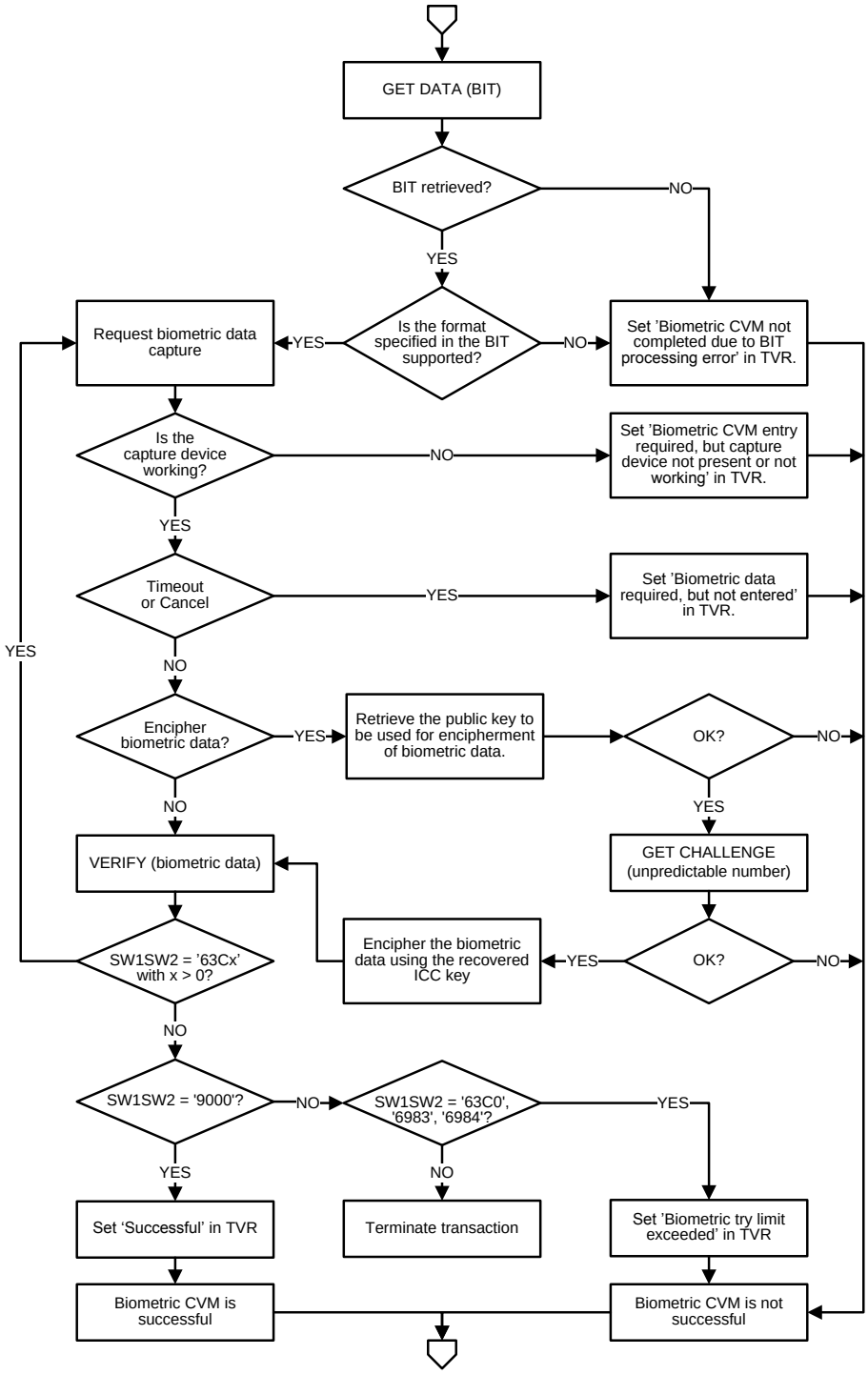

Fig. 2. Biometric CVM processing flow in the POS terminal

may deny the payment and initiate a chargeback. In general, only EMV payments with PIN entry are considered undeniable (even though [9] demonstrated that a particular implementation of the EMV specifications, "Chip and PIN", is vulnerable to a man-in-the-middle attack).

Biometric on-card verification will make fraud more difficult in case of payments without PIN entry, not least because EMV transactions include verifying the authenticity of the card. However, biometric comparisons are susceptible to false accept and false reject errors. As before, the cardholder should get the debited amount refunded when objecting within a time limit in case of a false accept error. If the card errs, a cardholder whose card has been stolen cannot be blamed. The biometric reference should be of high quality and hard to be forged and the verification threshold should be set such that false acceptances (i.e. misuse of the bank card by means of biometric on-card verification) are very rare.

In case of false reject errors, the card should offer yet another CVM (e.g. offline PIN verification) as a fallback. Therefore, the last entry in the CV rules list should not refer to the biometric on-card verification. A fallback is also necessary because not every payment terminal will be equipped with a signature pad.

5 Prototype

5.1 Java-card applet

We have created a prototype of an EMV payment application incorporating the proposed extensions for biometric cardholder verification on a Java card (JCOP 31 v2.2 or similar). The prototype uses the handwritten signature on-card verification applet presented in [10] with improvements discussed in [11]. The applet provides the Biometric Application Programming Interface (API) for Java cards [12].

5.2 Enrolment terminal

We have built a demonstrator for the personalisation of the card and handwritten signature enrolment based on a PC connected with a card reader and a graphics tablet.

During enrolment the user is required to sign five times. The five captured signatures are sent to the card in chained CHANGE REFERENCE DATA commands. After all signatures have been received by the card, each is compared with each of the others. They all are required to be similar enough to each other (satisfy a predefined threshold) for the process to continue. The results computed for the comparisons of each signature are then compared to the results of the other signatures. The signature with the least distance to the others is stored as biometric reference. The worst distances between the signatures are used to compute a decision threshold that is to be used in verification. Storing a signature of high quality as reference and setting a proper threshold for allowed signature variety are of crucial importance for ensuring low errors rates.

During personalization apart from the signature, personal data, cryptographic data, the CVM List and the BIT are stored on the card. After the phase has been completed successfully, the applet is marked as ready to use in transactions and all functionality for storing or changing data that is expected to be written only once is disabled.

5.3 POS terminal

To drive the Java-card applet and to show its capabilities, we have also built a terminal demonstrator simulating an offline POS terminal where an EMV transaction occurs and a tool for cryptogram verification. The demonstrator keeps a detailed log of the APDUs exchanged between terminal and card.

6 Summary and outlook

This paper describes a solution for extending the EMV specifications [6] to include biometric on-card verification methods. This allows deploying biometric on-card verification on debit and credit cards. If sufficiently resistant against direct and indirect attacks and if easy to use, biometric cardholder verification methods can strengthen the binding to the legitimate cardholder. The extensions proposed are fully in compliance with the requirements and side conditions imposed by [3, 4].

In order to demonstrate the feasibility of the design, the proposed payment system has been prototyped on Java cards [13] using a Java-card applet for online signature on-card verification [10]. Before this can be applied in real bank cards, it is advisable to thoroughly evaluate the security of the biometric on-card verification product based on officially recognized criteria like the Common Criteria for IT security evaluation [14].

References

1. B. Struif, "Use of biometrics for user verification in electronic signature smartcards," in *Proc. of the International Conference on Research in SmartCards: Smart Card Programming and Security*, ser. Lecture Notes in Computer Science, I. Attali and T. Jensen, Eds., vol. 2140. Springer, 2001.
2. "EMV integrated circuit card specifications for payment systems," Version 4.2, June 2008.
3. "Identification cards – Integrated circuit(s) cards," International Standard ISO/IEC 7816.
4. "Identification cards – Integrated circuits – Part 11: Personal verification through biometric methods," International Standard ISO/IEC 7816-11, first edition, 2004.
5. "Identification cards – Identification cards – On-card biometric comparison," International Standard ISO/IEC 24787, 2010.
6. "EMV integrated circuit card specifications for payment systems – Book 3: Application specification," Version 4.2, June 2008.
7. "Identification cards – Integrated circuit(s) cards with contacts – Part 4: Interindustry commands for interchange," International Standard ISO/IEC 7816-4, second edition, 2005.
8. "Information technology – Common biometric exchange formats framework – Part 3: Patron format specifications," International Standard ISO/IEC 19785-3, 2007.
9. S. Murdoch, S. Drimer, R. Anderson, and M. Bond, "Chip and PIN is broken," in *2010 IEEE Symposium on Security and Privacy*, 2010.
10. O. Henniger and K. Franke, "Biometric user authentication on smart cards by means of handwritten signatures," in *Proc. of the First International Conference on Biometric Authentication (ICBA 2004)*, ser. Lecture Notes in Computer Science, D. Zhang and A. Jain, Eds., vol. 3072. Springer, 2004, pp. 547–554.

11. O. Henniger and S. Müller, "Handwritten signature on-card matching performance testing," in *Proc. of the International Conference on Biometric ID Management and Multimodal Communication*, ser. Lecture Notes in Computer Science, J. Fierrez, J. Ortega-Garcia, A. Esposito, A. Drygajlo, and M. Faundez-Zanuy, Eds., vol. 5707. Springer, 2009.
12. *Biometric Application Programming Interface (API) for Java Card*, NIST/Biometric Consortium Biometric Interoperability, Assurance, and Performance Working Group, August 2002, version 1.1.
13. D. Nikolov, "Debit and credit cards with handwritten signature on-card matching," Master's thesis, Technische Universität Darmstadt, 2012.
14. "Information technology – Security techniques – Evaluation criteria for IT security," International Standard ISO/IEC 15408.