The Smarts Behind EMV Smart Cards Part 1 – Online Transaction Processing

Yash Kapadia
CEO
OmniPayments, Inc

Target, the third largest retailer in the U.S., suffered a card-skimming attack during the last months of 2013 in which hackers were able to obtain the magnetic-stripe data off of cards used in Target stores. Stolen was the personal data from 110 million payment cards.[1] Thousands of fraudulent transactions followed. Is there a defense against these data breaches?

The answer is yes – smart cards. A smart card, also called a chip card or an integrated-circuit card (ICC), includes an embedded computer chip that employs cryptographic and risk-management features. In conjunction with a smart-card POS or ATM terminal, these features are designed to thwart skimming, card-cloning, card- counterfeiting, and other fraudulent attacks.

Most card-payment networks include one or more HP NonStop servers. It is therefore important that the NonStop community understand smart-card technology, which is becoming an important component in all financial networks. In this article, split over two issues of The Connection, we describe how smart cards add significant security to payment card transactions. Part 1 covers the methods for authorizing smart-card transactions online with the issuer. In Part 2, we will discuss the procedures for securely authorizing smart-card transactions offline without direct issuer involvement.

The Worldwide Adoption of EMV Smart Cards

A decade or more ago, a consortium of card issuers comprising Europay, MasterCard, and Visa (EMV) began the specification of smart cards or as they are formally known today, EMV cards. EMV card technology has been adopted by most of the countries on all continents in the world. Excluding the U.S., there now exist 2.3 billion EMV cards and 37 million EMV terminals worldwide.

The operative term is "excluding the U.S." The United States is the laggard. Representing almost half of all payment cards and terminals in the world, the U.S. still runs its payment-card services on outdated magnetic-stripe technology. However, this is about to change. The U.S. payment-card industry has mandated that all merchants be EMV-compatible by October, 2015 (except for gas stations, which have until 2017) or face a "liability shift." If a merchant does not process at least 75% of its transactions through an EMV-enabled POS terminal (whether via chip cards or magnetic-stripe cards) and accepts a disputed or fraudulent card payment, the merchant will be liable for the transaction rather than the issuer.

The U.S. is well on its way to EMV acceptance. As of the end of 2013, all acquiring banks and issuing banks were ready and were helping merchants move to EMV technology. Payment-card issuers have distributed 20 million EMV cards and are expected to issue 100 million such cards by the end of 2014. ATM providers are actively deploying EMV-enabled ATMs.

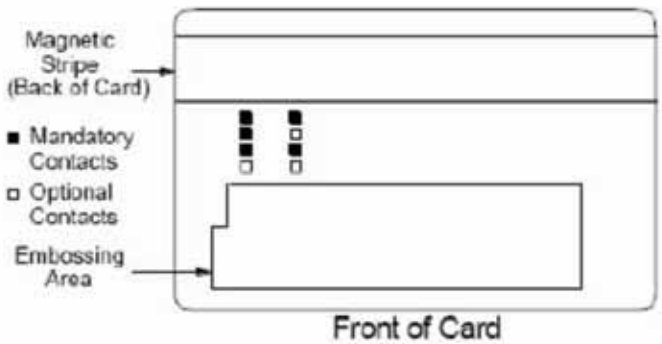
Already, several major retailers are EMV-ready, including Home Depot, Walmart, Best Buy, and Sam's Club.

The Structure of an EMV Card

An EMV card looks like a normal magnetic-stripe card, but in it is embedded a small but powerful microprocessor chip with exposed terminals.

The chip includes a processor, an encryption engine, and the capability to store data. During a card transaction, data flows in both directions between the card and the EMV POS terminal as well as between the card and the issuer. All data transfers are encrypted. However, no personal information is stored on the card other than that stored in a magnetic stripe – the Primary Account Number (PAN), the cardholder's name, and the card's effective and expiration dates.

Ingenico



[1] Target Compromises Millions of Payment Cards, Availability Digest; January 2014.

The data stored on the card comprises public data that can be accessed by the EMV POS terminal and secret data such as encryption keys that cannot be accessed. If any attempt is made to access secret data, the data is erased; and the card is rendered unusable.

Via the exposed contacts on the card's chip, data can be transferred between the POS terminal and the card by inserting the card into an EMV terminal. Contactless and mobile transfers are also supported.

EVMCo

Terminal Capabilities

EMV terminal capabilities are described by a three-byte, bit-encoded designator. These capabilities include:

Byte 1: Card Data Input Capability	Byte 2: Cardholder Verification Method (CVM)	Byte 3: Security Capability
Manual key entry Magnetic stripe IC (integrated circuit) with contacts	Plaintext PIN for ICC verification Enciphered PIN for online verification Signature (paper) Enciphered PIN for offline verification No CVM required	SDA DDA CDA (These capabilities are described later in Part 2 of this article.)

Table 1: EMV POS Terminal Capabilities

Encryption

EMV supports two types of encryption, depending upon whether the card is communicating with the issuer or with the terminal.

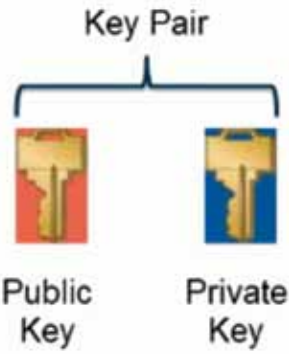
Card/Issuer Communication

Messages sent from the card to the issuer through the POS terminal are encrypted via Triple DES symmetric key encryption. With symmetric key encryption, a secret key is shared between each partner in the communication. The sender encrypts the message with the common key, and the recipient decrypts the message with the same key.

However, to preclude an attacker from obtaining thousands of messages all encrypted with the same key, which may give him the ability to deduce the key, each card has a separate key. The key that the card uses is derived from the issuer's key and the card account number. Communications from the card to the issuer are encrypted with the card's key. Using its own secret key, the issuer can deduce the card's key and decrypt the message. Messages that it returns to the card are encrypted with the card's key.

Though these messages pass through the POS terminal, the POS terminal cannot read these messages because it does not have the symmetric encryption key used to encrypt the message.

Card/Terminal Communication



Symmetric key encryption is not suitable for communication between the card and the POS or ATM terminal. The card would have to know the Triple DES key for every terminal in the world. Rather, public-key cryptography is used for card/terminal communications.

With this technique, a pair of keys is created. One key is a public key, and the other is a private key. The public key for a card can be made available to any POS terminal with which it needs to communicate. Messages encrypted with one key can only be decrypted with the other key.

For instance, a sender of a message can encrypt the message with the sender's private key, and the recipient can decrypt the message with the sender's public key. In this way, the recipient knows that the message has come from the sender (the sender has signed the message). Alternatively, the sender can encrypt the message with the recipient's public key, and the recipient can decrypt the message with its private key. In this way, the sender is confident that only the recipient will be able to read the message. By using double encryption with both keys, the recipient knows with certainty who the sender is; and the sender knows with certainty that only the recipient can read its message.

All public keys are registered with a trusted central Certificate Authority so that they can be verified as valid before using them to encrypt or decrypt messages to or from another device.

The EMV card uses public-key encryption to communicate with the POS terminal. The card has its own public and private keys and allows the terminal to read and authenticate the card's public keys.

Card Data

The card stores three kinds of data – static data, dynamic data, and secret data.

Static Data

An EMV card stores over five-dozen static data objects available to the EMV terminal. These data objects are static in that they are loaded onto the card by the issuer when the card is first created. Static data includes the following, the use for which will be described later in the article (unless already described above):

- the primary account number
- the cardholder's name
- the card's effective date
- the card's expiration date
- the cardholder's language preference
- issuer identification
- issuer public key
- card public key
- PIN public key
- public key Certificate Authority
- application ID(s) and name(s) (AID)
- Cardholder Verification Method (CVM)
- signed Static Application Data (SDA)
- Issuer Action Codes (IAC)

Table 2: EMV Card Static Data

Dynamic Data

Dynamic data is data that can be modified by card action or by issuer action. Modifications by the issuer are generally changes to the risk parameters used by the card to determine its response to a transaction. Dynamic data includes:

- transaction counter
- offline transaction counter
- maximum offline transactions
- lower offline floor limit
- upper offline floor limit

Table 3: EMV Card Dynamic Data

Secret Data

Secret data is not available to any external entity. Any attempt to access this data causes the data to be erased and the card rendered unusable Secret data includes:

- card symmetric key (for issuer messages)
- card private key (for terminal messages)
- card PIN
- card PIN private key

Table 4: EMV Card Secret Data

EMV Processing Flow

The processing of an EMV card transaction is shown in Figure 1. It comprises several steps, with interaction between the card, the terminal, and the issuer. Processing time for an EMV transaction is comparable to that for a magnetic-stripe transaction, where communication delays account for the majority of the time

EMV transaction processing begins with some preliminary steps that help determine whether the transaction should be handled online with the issuer or offline with no issuer involvement. Once this decision has been made, processing splits into two distinct flows – one for online transactions and one for offline transactions. Part 1 of this article describes the common processing flow and the online processing flow. Part 2 will describe the offline processing flow.

Preliminary Processing

Application Selection

The preliminary processing for a card transaction is shown in the flow chart of Figure 1. Each issuer can define its own application. The issuer's application determines how a card transaction must be handled. This includes under which

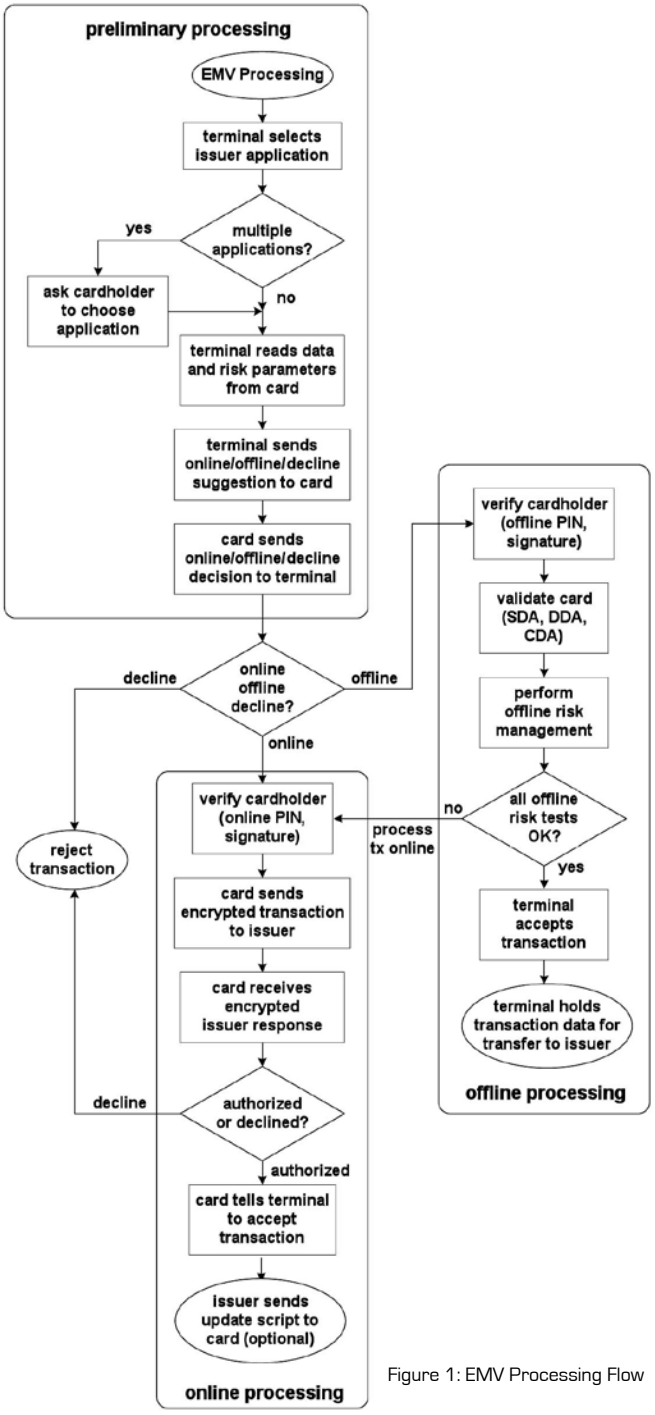


Figure 1: EMV Processing Flow

circumstances a transaction must be authorized online by the issuer, when it can be executed offline without issuer involvement, and when it must be declined. An issuer may have several applications. For instance, an issuer may allow a card to be either a credit card or a debit card.

The applications of all issuers are registered with a central Certificate Authority that assigns a unique application identification number (AID) to each application. In its static data, an EMV card carries a list of AIDs of all applications that can be used with the card (often, there is only one). The terminal has a similar list of all applications it supports. The terminal reads the AID list from the card and creates a list of common AIDs supported by both the terminal and the card. If there is more than one AID in the list, the terminal may ask the cardholder via the PIN pad which application he wants (do you want to use your card as a credit card or as a debit card?).

If there is no common AID, the transaction may be rejected; or it may proceed as a magnetic stripe transaction, depending upon how the terminal is configured. Attended POS terminals such as in stores are required to be magnetic-stripe capable.

Terminal Action Analysis

When an AID is selected, the terminal requests from the card the list of data elements that the terminal needs to process the transaction. The card may also request certain data elements from the terminal.

The terminal checks to see if the card can be used:

- Is the AID on the card a current version?
- Are there restrictions on the card as to where it can be used?
- Is the card within its effective and expiration dates?

If any of these conditions are not met, they are noted for later risk-management processing. However, they do not lead immediately to a transaction rejection. For instance, an issuer may be willing to allow a card to be used for a time period following its expiration date but only for online transactions.

Among the data read from the card are the Issuer Action Codes (IAC). They are bit-mapped conditions that specify criteria imposed by the issuer for how a transaction is to be processed. The terminal combines them with a set of Terminal Action Codes (TAC) (also bit-mapped conditions) to determine if the transaction should be processed online or offline or be declined.

Most transactions will be processed online, sending the transaction data to the issuer for authorization. However, in some cases, the transaction may be processed offline with no issuer involvement. This can take place if there is no communication link (for instance, terminals used in airplanes to sell food, drinks, and other items), if the communication link between the terminal and the issuer should fail (allowing a retailer to continue to service customer payments at its own risk), or for very small transactions.

As the terminal analyzes a transaction with respect to the IAC and TAC criteria, it creates a bit-mapped Terminal Verification Result (TVR) that shows which conditions have been violated (such as card is expired). When the terminal has made a decision as to how the transaction should be processed based on the IAC, TAC, and other parameters (such as transaction amount), it sends the resulting TVR and other transaction data to the card and indicates to the card the terminal's suggestion relative to online processing, offline processing, or rejection. This is only an advisory suggestion to the card.

First Card-Action Analysis

During the first card-action analysis, the card considers the terminal's recommendation and makes its own decision as to whether the transaction should be processed online, offline, or rejected. (The second card-action analysis is taken when the card receives a response from the issuing bank, as described later).

Based on the results that the terminal sends to the card, the card returns an application cryptogram (AC) to the terminal that indicates how the transaction should be processed. The application cryptogram is encoded with the card's private key and is decrypted by the terminal with the card's public key.

There are three ACs that the card can return to the terminal:

- a Transaction Cryptogram (TC) that indicates that the transaction can be approved offline.
- an Application Request Cryptogram (ARQC) that indicates that the transaction must be approved online.
- an Application Authentication Cryptogram (AAC) that indicates that the transaction is to be declined.

The card can accept the terminal's suggestion, or it can force the transaction to be processed online or to be rejected. However, the card cannot force a transaction to be processed offline if the terminal has indicated that it must be processed online (it cannot return a TC if the terminal has asked for an ARQC).

Online Transactions

Cardholder Verification

If the transaction is to be processed online, the next step is to verify that the person presenting the card is the legitimate cardholder. This is accomplished via the Cardholder Verification Method (CVM) that the issuer has specified in its application stored on the card (see Table 1). Either of three methods can be used to verify the cardholder for an online transaction:

- online PIN
- signature
- no CVM.

Whatever methods the issuer selects may be arranged in priority order, or they may be selected according to other transaction parameters. For instance, an ATM transaction may require a PIN. A POS terminal with no PIN pad may call for a cardholder signature. A small transaction may require no cardholder authentication. A transaction at an attended POS terminal may require both a PIN and a signature.

If a PIN is selected, the cardholder enters the PIN into the terminal's PIN pad. The terminal encrypts the PIN with the card's public key and sends it to the card. The card decrypts the PIN with its private key and compares the PIN entered by the cardholder to the PIN value stored in its secret data. If the PIN is wrong, the terminal is informed. According to a PIN counter, the cardholder may be given additional opportunities to enter his PIN. If he reaches a specified limit, the transaction is rejected; and the card is blocked and no longer can be used.

Online Transaction Authorization

When all data for an online transaction has been assembled by the card, the card asks for transaction authorization by sending a message to the issuer. The card encrypts the message with the card's symmetric key shared with the issuer. This encryption represents a digital signature by the card and guarantees to the issuer that the

transaction comes from a valid card.

The issuer responds to the card with an encrypted ARPC (an Application Reply Cryptogram) that indicates whether the transaction is accepted or declined.

Second Card-Action Analysis

When the card receives the issuer's ARPC, it decrypts it using the issuer symmetric key. It is possible for the card to reject an issuer authentication and to decide instead to abort the transaction, in which case a reversing transaction is sent to the issuer. Based on the ARPC and the card's final decision, the card will inform the terminal to accept or decline the transaction. If a signature is required for an authorized transaction, the cardholder will be asked to sign a copy of the transaction receipt.

Following the transaction, the issuer may optionally send a script to the card to change certain parameters in order to update the issuer's risk-management processing. For instance, the card's PIN can be changed, the card can be blocked or unblocked, and other risk parameters can be modified.

The OmniPayments Financial Transaction Switch

OmniPayments (www.omnipayments.com) from Opsol Inc. (www.opsol.com) is an HP NonStop-based financial transaction switch that interconnects POS terminals, ATMs, acquiring banks, and issuing banks via any of the various financial transaction networks. OmniPayments supports all features required to process EMV smart-card transactions, from support of EMV POS terminals and ATMs to the protocols required to communicate with the issuing and acquiring banks. OmniPayments is currently handling EMV transactions with its international banking installations and is ready to handle these transactions with U.S. systems as EMV technology takes hold in the United States.

With successful implementations at many customer sites, OmniPayments is just one member of the Opsol family of solutions for the financial industry. Opsol Integrators specializes in NonStop mission-critical applications and is HP NonStop's largest system integrator.

Summary

In Part 1 of our article on EMV technology, we have described EMV smart cards and EMV terminals and how they manage payment transactions that are to be approved online by the issuer. In Part 2, we extend this discussion to how EMV transactions are securely processed offline with no direct issuer involvement.

The OmniPayments financial transaction switch supports EMV POS terminals and ATMs, EMV protocols, and other EMV requirements necessary to bring the enhanced EMV security capabilities to the financial payment industry.

Yash Kapadia is the founder and CEO of OmniPayments Inc., a leading HP NonStop System Integrator for Telco and Financial Services. Opsol's OmniPayments solution is used by Banks and Retailers for Base24 replacement. Yash and his team provide several products and remote managed services for NonStop customers. Yash can be reached at Yash@OmniPayments.com and +14086669927.