Applied Cryptography for Magnetic Stripe cards

(This is the distributed text of a paper by Andrew Marshall, published in HTML format.

Comments and critique are always welcome, either by traditional methods or by Email here.

1.0 Introduction

The intention of this document is to provide a basic understanding of cryptography and techniques applied to magnetic stripe cards in the financial industry.

This subject is normally approached with some trepidation by the uninitiated, however it is reasonably straightforward once the basic principles are explained.

Cryptography is complex, but its practical application is less so. It is not necessary to understand the mathematics involved in order to successfully use and manage cryptography in a financial environment.

Because of the security implications of card cryptography, it is extremely hard to find information in any form explaining this application, which adds to the somewhat unnecessary shroud of mystery surrounding the topic. In early implementations, a measure of additional security was provided by ensuring that few people knew exactly how these mechanisms

worked and this method of operation has permeated into today's implementations.

However, none of the information provided in this document will compromise security in any way.

Although other, more secure card tokens are becoming available, the magnetic stripe card is significantly cheaper than alternatives, and is by far the most common card type in use. Security techniques for magnetic cards have slowly but steadily improved, and properly implemented can provide perfectly adequate security for financial transactions in a very cost-effective manner.

2.0 Use of cryptography in financial magnetic stripe cards

The most commonly known use of cryptography is in the provision of a Personal Identification Number, or PIN, to allow a magnetic stripe card to be used in unattended environments such as ATM's, or in other situations where traditional signature checking is inappropriate. This applies equally to credit, debit and ATM cards. There are not many financial cards in use today that do not have some kind of PIN capability.

A second common use of cryptography is in providing anti-counterfeit mechanisms for the magnetic stripe. The intention is to prevent fraudulent construction of counterfeit cards by inserting a value on the magnetic stripe that cannot be derived from other card information. Thus when a card is validated online this value can be checked to determine whether the card is genuine or a forgery. Several different standards exist for this mechanism, the most common being the VISA Card Verification Value (CVV) or the Mastercard equivalent, CVC. For the purposes of this document I will refer to this mechanism as CVV as this is the term in most common use.

Other uses of cryptography do not directly relate to the card, they generally relate to the encryption of PIN's and messages whilst being transmitted in a financial environment to prevent their disclosure or alteration.

These items will be discussed in more detail in subsequent sections.

3.0 Basic Cryptography

A basic understanding of cryptographic techniques is required in order to understand this document.

The majority of magnetic card encryption is based on the Data Encryption Algorithm (DEA), usually called DES or Data Encryption Standard. The idea behind DES is that a clear value is passed to the DES algorithm, which can be implemented either as software routines or in dedicated hardware. DES then encrypts the clear value using a key (a secret 64-bit value) and outputs an encrypted value.

The unencrypted input is usually referred to as *Cleartext*, while the encrypted result is referred to as *Ciphertext*. The operation that turns cleartext into ciphertext is known in DES terms as an 'encipher' operation.

Thus:

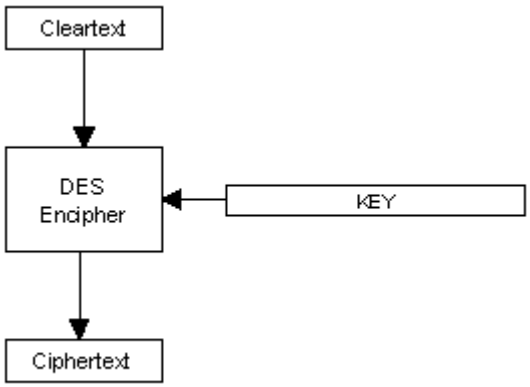

Figure 1 - DES Encipher operation

Note the following:

1. The DES algorithm is NOT secret. It is publicly available. The Key, however, *is* secret.
2. This process is reversible. Executing a DES 'decipher' function using the same key will convert the ciphertext into cleartext.

A value encrypted with a key is generally referred to as being encrypted 'under' that key.

The security and integrity of the whole operation depends on the secrecy of the key used. The key is a random value that is strictly protected and never disclosed or written down. Most of the complexity involved in DES cryptography systems is related to protecting, storing and transmitting keys, and these activities are referred to as *key management*.

Note also that the DES encipher operation as described above is not foolproof. In theory, a massively parallel processor could derive the key in about a days processing. Much is made of this possibility in discussions on strengthening security, however, additional procedures can be implemented which go some way towards reducing the effect of this limitation.

If we take a simple example to demonstrate this: computer logon passwords.

Passwords used on computer systems are commonly encrypted after they have been set, and they are stored in a file in encrypted format. When a user signs on, the password is entered, usually in a hidden field, in cleartext. It is important to understand that this value is NOT compared against a value that is deciphered from the password file. The cleartext password in enciphered under the same key and compared against the enciphered value stored on the password file. Cleartext, enciphered under the same key, will always provide the same result, and almost all cryptographic validation compares ciphertext to ciphertext to avoid exposing cleartext values inside computer systems that could be compromised by memory dumps and so on.

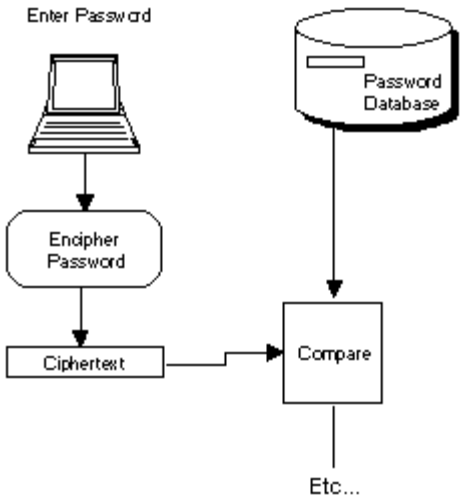

Figure 2 - Password encryption

In this scenario however, a user of a password can always claim that his password can be exposed by deciphering the enciphered value, and that this is not under his control - and this is true.

Dynamic key exchange

Many financial systems implement dynamic key exchange. While not exclusively relating to magnetic stripe cards, it is relevant to include it here.

In dynamic key exchange, two parties change keys 'on the fly' to ensure that one key is not used for an extended period and risks exposure. This is normally used in the financial environment where two hosts are exchanging financial authorisation messages - for example an acquirer bank and an issuer bank. When the acquirer bank forwards the PIN to the issuer bank for validation, it must do so encrypted to avoid disclosure. Obviously, the issuer will need access to the key used to encrypt the PIN so that it may be deciphered for validation. These keys will have been previously agreed, and may be changed using dynamic key exchange where keys are shipped (themselves enciphered under a 'key encryption key') and changed frequently in real time for added security.

It must be stressed that no cryptography system is ever completely secure. There are always weaknesses in any system, both from a technical viewpoint and operationally, where human and operational procedures may be compromised.

4.0 Practical application of cryptography in Magnetic stripe cards.

The intention of this section is to demonstrate how cryptographic principles are (usually) applied to magnetic stripe cards in a practical context.

4.1 PIN Processing

The PIN principle is based on the fact that nobody other than the legitimate cardholder has knowledge of the PIN. Thus when a PIN is provided for a customer:

- It must not be stored anywhere in cleartext (except in the secure PIN mailer destined for the customer)
- It must not be possible to reverse-engineer the PIN from information on the magnetic stripe or from a centrally held database.

Normally, a PIN is a 4-digit numeric value. Other schemes exist, but we will use this format for illustration as it is a common standard.

When a PIN is issued, the sequence of events is as follows:

- A 4-digit random number is generated. This is the PIN.
- The PIN is combined with other information, such as the account number, to create a block of data for input to the cryptography process.
- The input block is triple encrypted using the PIN working keys
- Digits are selected from the ciphertext result. These become the Pin Verification Value or Pin Offset.
- The PIN Offset is stored
- The PIN mailer is printed
- Memory is cleared to binary zeroes to remove all traces of the clear PIN.

At this point, the only place the PIN value exists is inside the PIN mailer. The PIN cannot be derived from the PIN offset.

When the card is used and the PIN entered, the PIN offset is calculated again from the entered PIN, using the PIN working keys and compared to the stored offset value to determine if the correct PIN was entered. Clearly this means that when a PIN is validated, the validating system must have access to the PIN working keys used during initial PIN issue or subsequent PIN change.

It should be re-emphasised that the offset comprises *selected digits* from the ciphertext. Typically this would be 4-6 digits. It is not possible to recreate the keys or derive the PIN from this value.

Notes:

I.In some implementations, the PIN offset is stored on the magnetic stripe on the card. This is intended to be used in terminals which can perform local PIN validation. However, this technique is becoming rare as it prevents deployment of user-selectable PIN's.

II. Where the user is given the option to change PIN, the new offset is calculated in realtime and written to the database. Note that if the PIN is forgotten, it cannot be recreated.

III. The method described above is generic. There are many variations, such as the IBM3624

Method-A, Diebold method, and so on, however the principle remains the same.

IV. In many methods, the framework exists for using different key pairs based on an index value, usually stored on the magnetic stripe. This is a single digit value denoting the index of the key pair to be used. The intent is so that a) the same keys are not used across the entire cardbase, and c) that new keys can be used on re-issue without affecting existing cards.

4.2 CVV processing

It was quickly understood that the proliferation of financial cards exposed institutions to risk from counterfeiters. In the credit card world, this came from manufacture of cards with or without magnetic stripe encoding that possessed valid numbers and seemingly valid names and logos. In the ATM card arena, attackers observed PIN number entry 'over the shoulder', collated these PIN's with information from discarded receipts and so on, and constructed their own magnetic stripes on dummy cards for use at their leisure with observed PIN numbers.

These threats and others led to the introduction of the Card Verification Value, a non-derivable sequence of digits constructed by cryptographic process and written to the magnetic stripe of the card. This means that electronic capture of transactions (either at ATM or Point of Sale) are effectively protected against counterfeiters.

A combination of static data such as account number is triple encrypted using a special Card Verification key pair. Selected digits from the result are used to create the CVV, and this is written onto the magnetic stripe.

Similar comments apply to CVV as those for Pin Offset; As the CVV consists of few digits, and triple encryption is used, the CVV keys and values are highly secure and presence of a valid CVV provides an added level of confidence that the card is not counterfeit.

It should be noted that CVV is simply an additional protection method; it is not foolproof. It does not, for instance, protect against fraudulent captures of magnetic stripe data using, say, fake ATM's.

A further development of CVV, CVV2, is used for telephone authorisations. A similar (although not identical) calculation is performed as for CVV, and selected digits from the result are physically printed on the back of the card. These digits can then be requested by a call centre wishing to determine if the caller is really in possession of the card. Once again, this is an additional check, and not foolproof.

4.3 Key management

Key management relates to the storage, protection and transmission of keys. A single financial installation will have many DES keys, and these require careful management if they are not to become compromised or confused. One of the worst forms of debugging of computer faults is when cryptography is involved as traces and dumps are meaningless, and it can be very hard to discover that the wrong cryptography keys are being used!

Keys are normally managed in hierarchies. Keys that are actually used for computation, such as PIN validation [working keys] are themselves stored in enciphered format under a key encryption key. Other key sets will exist for transporting keys from one location to another,

such as two nodes in a network. These are known as transport keys.

In good key management systems, working keys are never stored or exposed in clear format. Even when they are initially created, they are frequently created by automated process and never known to individuals.

When initial keys are created, the 64 bits are split between two or more individuals, who then toss a coin once for each bit required. The two or more individuals then key in their segment of the random key alone, and thus no one individual ever has sight of a whole key. This method is normally used for initial master key generation.

Although a simple concept, key management can become quite complex in implementation.

In a simple ATM network for instance, a terminal master key is used to encipher working keys in transit. A terminal master key (TMK) is generated for each terminal, split into two halves and printed (or sometimes encoded on a special magnetic card). Each TMK is then installed at their respective ATM's. The host system will then download terminal working keys, enciphered under the respective terminal master key, to each ATM. The terminal working key is then used to encipher PIN data in transit to the host during normal processing. If required, the terminal working key can be changed at regular intervals or through dynamic key exchange - but this process requires careful management.

It should be noted that the biggest single security exposure to DES based cryptographic subsystems is in the exchange of keys, thus good key management procedures are paramount.

4.4 Physical implementation

Cryptographic processing and key management is normally performed in specialised, dedicated secure hardware. Although DES can be implemented entirely in software (using products such as IBM's PCF), it is less secure, and the DES algorithm can be quite processor intensive.

There are companies that specialise in dedicated cryptographic units, such as Racal and Atalla. They are commonly called HSM's (Host Security Module) although this is the Racal proprietary name for the unit.

When using these devices, the intent is that all encipher and decipher activity takes place in the secure unit, and that clear keys and cleartext values are never exposed outside the unit.

Physically, HSM's are tamper proof and intended for installation in secure computer rooms. Attempts to open them will result in the destruction of keys contained in the devices.

HSM's are also capable of generating new random keys and random numbers for use as PIN's in a secure manner.

Some applications use physical telecommunications line encryption for added security, and there are a variety of manufacturers of this type of device. They are effectively 'black box' and require no special knowledge.

5.0 Examples

5.1 Cryptography in a normal ATM withdrawal

Consider a common ATM transaction:

1. A customer inserts his card in the ATM
2. The customer enters his PIN
3. The customer requests cash
4. The transaction is approved, cash is dispensed

There's an awful lot of cryptography going on in this process. For simplicity, we'll assume the acquiring and issuing bank are the same.

The cryptography activity is identified in italics in the sequence:

1. A customer inserts his card in the ATM

The magnetic stripe is read and stored in a buffer in the ATM

2. The customer enters his PIN

The PIN is entered into a tamper-proof PIN pad The stored PIN is stored in a security module in hardware

3. The customer requests cash

The message is constructed in the ATM The PIN (and possibly more) is enciphered under the Terminal key

The message is sent to the host, possibly enciphered in comms hardware.

On receipt at the host, the comms level encryption is deciphered The CVV is calculated and compared to the value on the magstripe The PIN under the Terminal key is deciphered The PIN offset or PVV is calculated The PIN offset or PVV is compared to the database of PVV's

4. The transaction is approved, cash is dispensed

Note: all the host cryptography functions are normally performed in the Host Security module. No Cleartext values are exposed to application programs or outside the secure environment.

5.2 Cryptography in an EFTPoS transaction

Even in a signature authorised environment, the CVV from the magnetic stripe can be validated at the host system to detect counterfeit cards. Clearly this only works in online environments as the CVV validation requires a cryptographic calculation to be performed at the host.

[Note: It is possible, and some manufacturers support, local key storage on EFTPoS devices and distributed terminals. Because of the key management complications, these devices are not considered here]

A more common use of cryptography in EFTPoS environments (and, increasingly in ATM and other traffic) is the MAC (Message Authentication Code). The MAC check can be thought of as a value calculated from the contents of all the critical fields in a message (such as card number and amount) and passed through a cryptographic algorithm. Although the message is carried over transmission lines in clear, the validation of the MAC field at the recipient will determine whether fields have been tampered with. [for the technically minded, MAC can be thought of as an encrypted LRC field]. The overhead of MAC is quite small. (The MAC is defined as 16 bytes in ISO8583).

5.3 Other financial cryptography applications

As well as traditional uses of cryptography as described above, interbank networks (such as SWIFT) have historically been large users of cryptographic techniques.

A plethora of new delivery mechanisms and far wider distribution of advanced technology to the public has increased both the interest in and the use of cryptographic techniques.

In cases where cryptography is required for widespread dissemination to the public (such as PC based home banking) ordinary DES is too complex to manage securely. More appropriate and more secure algorithms such as RSA (A "public key" encryption system) have evolved and been deployed in these environments - they are outside the scope of this paper but review of public key algorithms is especially encouraged where appropriate.

Some corporate, EDI and treasury applications use highly secure DES with a combination of techniques - MAC, physical encryption, dynamic key exchange, smart card key storage and so on. In one implementation reviewed, the working key is changed every transaction by the result of a MAC key calculation residue (a so-called "one time" key system).

END OF DOCUMENT
