EXHIBIT 1

AMERICAN NATIONAL STANDARD

DEVELOPED BY
ACCREDITED STANDARDS COMMITTEE
X9—FINANCIAL SERVICES
PUBLISHED BY
AMERICAN BANKERS ASSOCIATION
X9—SECRETARIAT

Financial Institution Retail Message Authentication

X9.19

Approved August 13, 1986

Developed by the Accredited Standards Committee on Financial Services, X9, operating under the procedures of the American National Standards Institute.

Published by the X9 Secretariat, American Bankers Association, 1120 Connecticut Avenue, N.W., Washington, D.C. 20036.

AMERICAN BANKERS ASSOCIATION
1120 Connecticut Avenue, N.W.
Washington, D.C. 20036

American National Standard

Approval of an American National Standard requires verification by ANSI that the requirements for due process, consensus, and other criteria for approval have been met by the standards developer.

Consensus is established when, in the judgment of the ANSI Board of Standards Review, substantial agreement has been reached by directly and materially affected interests. Substantial agreement means much more than a simple majority, but not necessarily unanimity. Consensus requires that all views and objections be considered, and that a concerted effort be made toward their resolution.

The use of American National Standards is completely voluntary; their existence does not in any respect preclude anyone, whether he has approved the standards or not, from manufacturing, marketing, purchasing, or using products, processes, or procedures not conforming to the standards.

The American National Standards Institute does not develop standards and will in no circumstances give an interpretation of any American National Standard. Moreover, no person shall have the right or authority to issue an interpretation of an American National Standard in the name of the American National Standards Institute. Requests for interpretations should be addressed to the secretariat or sponsor whose name appears on the title page of this standard.

CAUTION NOTICE: This American National Standard may be revised or withdrawn at any time. The procedures of the American National Standards Institute require that action be taken to reaffirm, revise, or withdraw this standard no later than five years from the date of approval. Purchasers of American National Standards may receive current information on all standards by calling or writing the American National Standards Institute.

CONTENTS

FOREWORD

(This Foreword is not part of American National Standard X9.19-1986.)

Today, billions of dollars in funds and securities are transferred by telephone, wire services, card interchange networks, and other communications media. Transactions are often entered remotely, off-premises from the institutions, by non-institution personnel or by customers directly, and are transmitted over potentially insecure media. The vast range in value size and volume of such transactions expose institutions to severe risks (which may be uninsurable) both from accidental and deliberate alteration of messages and from the introduction of fraudulent messages. A uniform process for institutions' transactions is required which facilitates:

- o Verification that selected contents of the message have not been altered in transit,
- o Verification of the identity of the originator and intended recipient,
- o Use by both large and small organizations, and
- o Implementation in automated systems and in manual systems, where applicable.

This standard defines a process for authentication of messages between originator and recipient which, in some applications, are referred to as sender and receiver. This process is independent of the communications media, protocols, character codes and transaction stages.

The authentication process includes the computation, transmission, and validation of a Message Authentication Code (MAC). The American National Standard, Data Encryption Algorithm (DEA) (See Reference 1), a modern algorithm which is not designed for hand computation and which requires a secret key, is used to generate a MAC.

The MAC is based either on the complete message text or on selected elements of the text. The MAC is added to the message by the originator and is transmitted to the recipient. The message or message elements are accepted as authentic by the recipient if the same algorithm and secret key produce a MAC identical to the received MAC. Bogus or altered messages will fail such tests.

Since the DEA algorithm is in the public domain, the security of the authentication process is directly dependent on the security afforded to the secret key.

To support the varying operational demands of and among different institutions, their networks, and their customers, the authentication algorithm can be implemented either through special equipment or computer programs. The authentication process can be performed using independent devices or as part of a computerized system.

It is the responsibility of the institution to put an overall security process in place with the necessary controls to assure that the process is implemented under secure procedures and physical security. Further, the controls should include application of appropriate audit and sensitivity tests in order to ensure compliance.

This document provides both standards and guidelines for use as part of that overall process. It applies to selected message elements from point of MAC computation to the MAC check. Validity of the MAC computation input values and the use of the MAC computation to check output results must be a responsibility of the overall process provided by the institution.

The use of this technique in no way ensures that the overall process or even the application of the technique as a part of the process will, in itself, ensure secure results. The institution must assure that the overall process is secure.

Suggestions for the improvement or revision of this standard will be welcome. They should be sent to X9 Committee Secretariat, American Bankers Association, 1120 Connecticut Avenue, N.W., Washington, D.C. 20036.

The standard was processed and approved for submittal to American National Standards Institute by the Accredited Standards Committee X9 - Financial Services. Committee approval of the standard does not necessarily imply that all committee members voted for its approval. At the time it approved the standard, the X9 committee had the following members:

Robert Kaminski, Chairman
Donald Monks, Vice Chairman
Cynthia Fuller, Secretariat

Organization Represented	Representative
American Express Company	Dave Siegal
Bank of America	John Coombs
Burroughs Corporation	Stanley Fenner
Chase Manhattan Bank, N.A.	John McKessy
Citibank, N.A.	Seymour Rosen
Continental Illinois National Bank & Trust	Joseph Coriaci
Dollar Dry Dock Savings Bank	John Petrusky
Electronic Funds Transfer Association	Michael Strada
Federal Reserve Bank of Dallas	Johnny Johnson
First Interstate Services	Charles Pecharka
IBM Corporation	Dan Sundberg
Irving Trust Company	Dennis Beuchler
MasterCard International	Alice Droogan
Mellon Bank	Eugene Cooney
Moore Business Forms, Inc.	Delmer Oddy
NCR Corporation	A. R. Daniels
Security Pacific National Bank	Janice Hardina
U. S. League of Savings Institutions	O. Tom Thomas
Valley National Bank	Gene Saunders
VISA, USA	Jean McKenna
XEROX Corporation	Glenn Mulligan

The X9A Subcommittee - Electronic Retail Financial Transactions had the following members:

O. Tom Thomas, Chairman

Organization Represented	Representative
A. O. Smith Data Systems	Honora Norton
American Express Company	Bonnie Howard
AT&T Information Systems	Al Brown
Bank of America	Arnold Birenbaum
Chase Manhattan Bank, N.A.	Joel Bloom
Cirrus System	Jay Levy
Citicorp - Transaction Technology Inc.	Grant Laney
City National Bank	Lemont Southworth
Financial Interchange	Steven VanFleet
IBM Corporation	Alan Stein
Instabank	Mark Zalewski
Interlink	Morgan Whitener
J. C. Penney Company	Melvin Benovitz
MasterCard International	Alice Droogan
NCR Corporation	Susan Howorth
National Bank of Detroit	Michael Dion
Navy Federal Credit Union	Russell Thompson
National Automated Clearing House Assn.	Richard Brandt
Southeast Bank	John Gaughan
The Exchange System	Jack LaBounty
Tyme Corporation	Jack Derr
U. S. League of Savings Institutions	O. Tom Thomas
VISA, USA	Susan Crawford

The standard was developed by the X9A3 Working Group - Security in Financial Networks. X9A3 consisted of the following dedicated members:

Joel Bloom, Chairman

Organization Represented	Representative
Atalla	Frank Piedad
Bank of America	Geoffrey Turner
Burroughs Corporation	Daryl Carey
Chase Manhattan Bank, N.A.	Joel Bloom
Citicorp - Transaction Technology Inc.	Grant Laney
Consultant	Howard Zeidler
Diebold Incorporated	Ron Robinson
IBM Corporation	Dennis Abraham
National Bureau of Standards	Miles Smid
Network Controls International	John Collins
Racal-Vadic	Henry Beker
	Ramon Campo
State of Wisconsin	Robert Patrick
Tyme Corporation	Jack Derr
VeriFone, Inc.	Mohammad Khan
VISA, USA	Carl Campbell

1. INTRODUCTION

1.1 PURPOSE

A financial institution is an establishment responsible for facilitating customer-initiated transactions or transmission of funds for the extension of credit, or the custody, loan, exchange, or issuance of money. This standard establishes a universally applicable method to authenticate financial messages for retail transactions.

Financial institutions participating in EFT networks need standards to ensure the common operability of the countermeasures they use to safeguard EFT transactions from computer fraud. Such standards improve operating flexibility and provide for an overall level of security protection. Figure 1.1 illustrates how EFT transactions are comprised of individual messages between originators and recipients in EFT networks. This standard describes a message authentication technique for protecting EFT systems against:

- message alteration and
- fraudulent insertion of messages.

Although not covered in this standard, use of this technique should be part of an overall security process to protect against:

- fraudulent replay of messages and
- fraudulent deletion of messages.

Examples for detection of fraudulent replay and fraudulent deletion are described in Appendix B.

EXAMPLES OF MESSAGES AND TRANSACTIONS,
SENDERS AND RECEIVERS, ORIGINATORS AND RECIPIENTS

Example 1:

	TERMINAL	INSTITUTION A
THE TRANSACTION REQUEST FOR A:	originator	→ destination
is this message:	originator	→ recipient
and this transmission:	sender	→ receiver
THE TRANSACTION RESPONSE:	destination	← originator
is this message:	recipient	← originator
and this transmission:	receiver	← sender

Example 2:

	TERMINAL	INSTITUTION A	SWITCH	INSTITUTION B
THE TRANSACTION REQUEST FOR B:	originator			→ destination
is comprised of these messages:	originator	→ recipient		
		originator		→ recipient
and these transmissions:	sender	→ receiver		
		sender	→ receiver	
			sender	→ receiver
THE TRANSACTION RESPONSE:	destination			← originator
is comprised of these messages:		recipient		← originator
	recipient	← originator		
and these transmissions:			receiver	← sender
		receiver	← sender	
	receiver	← sender		

Figure 1.1

1.2 SCOPE

Techniques are presented for authenticating messages or selected message elements. A brief tutorial on the needs for authenticating messages is presented in Appendix A. The financial institution's authentication process is not to be implemented or controlled in a manner that has less security, protection, or control than described herein.

This standard specifically does not cover:

- o communications related characteristics, e.g. protocol, character code, header or trailer information
- o formats, meanings or contents for specific data elements, messages or transactions
- o transaction integrity
- o privacy
- o key management. However, it is stressed that this is of the utmost importance in the security procedures outlined, since keys must be protected from disclosure or compromise.

Assurance of adequate implementation and operation of authentication systems is beyond the scope of an ANSI standard.

1.3 REFERENCES

This standard shall be used in conjunction with the following publications.

1. ANSI X3.92-1981 Data Encryption Algorithm (DEA)
2. ANSI X3.106-1983 Modes of DEA Operation

The following publications are applicable and may be referenced in this standard.

3. ANSI X9.2-1980 Interchange Message Specification for Debit and Credit Card Message Exchange Among Financial Institutions
4. ANSI X9.8-1982 Personal Identification Number (PIN) Management and Security

1.4 APPLICATION

This standard is applicable for institutions responsible for implementing techniques to safeguard EFT messages from computer fraud. Specifically, it applies in all situations where a financial message is transmitted between an originator and a recipient.

Design standards for message authentication are specified. Performance standards are designated as mandatory when they are necessary to ensure that the minimum acceptable performance within any of the involved networks does not deteriorate the security.

For private networks, message authentication is not as critical as for interchange networks. This is because there is no equivalent to the fundamental commitment and responsibility of each interchange institution to the security of all interchange partners' sensitive data and keys in connection with interchange transactions. However, the standards and guidelines specified in this document are generally applicable to private networks, and their eventual implementation in such networks is recommended.

Mandatory standard techniques and procedures are indicated by the word "must". Guidelines are indicated by the word "should".

1.5 DOCUMENT ORGANIZATION

The remainder of this document is divided into two sections:

Section 2 establishes general standards applicable to financial messages, and

Section 3 further establishes detailed standards specific to retail financial messages.

Thus, the corresponding subsections of Sections 2 and 3 must be "combined" to complete the standard for retail financial messages. For example,

Section 2.4.3 addresses message elements in general, and

Section 3.4.3 addresses message elements in more detail, specifically for retail financial messages.

2. GENERAL MESSAGE AUTHENTICATION

2.1 PURPOSE

Section 2 establishes a universally applicable method to authenticate financial messages. Application-specific requirements must be incorporated with this method to complete the standard.

2.2 SCOPE

Section 2 outlines methods that are common to all applications. Subsequent sections define the methods for specific EFT applications in greater detail - to encompass specific requirements, message types, data types, situations, etc.

2.3 DEFINITIONS

algorithm	a clearly specified mathematical process for computation; a set of rules which if followed will give a prescribed result.
alteration	the process of modifying one or more message elements of a message as a method of perpetrating a fraud.
authentication	the act of determining that a message comes from a source authorized to originate messages of that type and that the message is as authorized.
authentication algorithm	an application of an encryption process in which the results of cryptographically processed text depend upon all participating authentication elements.
authentication element	a contiguous group of bits or characters which are to be protected by being processed by the authentication algorithm.
biased	with respect to generation of random or pseudo-random numbers, a process is biased if the occurrence of some numbers and/or patterns is more likely than others.
block encryption	under DEA (See Reference 1), 64 bits of cleartext are encrypted to yield 64 bits of encrypted text.
CBC	Cipher Block Chaining (See Reference 2).

CFB	Cipher Feedback (See Reference 2).
ciphertext	encrypted output of a cryptographic algorithm.
cleartext	data in its original, unencrypted form.
closed-loop response integrity	the verification by the originator of the overall transaction integrity, i.e. of both the transaction request and its transaction response.
customer	the individual initiating the transaction.
Data Encryption Algorithm (DEA) Standard	the cryptographic algorithm adopted by ANSI (See Reference 1).
deletion	the process of preventing a message from being delivered to the intended recipient as a method of perpetrating a fraud.
design standard	specific design criteria defining both results and method of performance per a standard.
duplication	same as replay.
encryption	a process of transforming cleartext into ciphertext for security or privacy.
exclusive-or	a mathematical operation equivalent to binary addition, without carry; symbol (+) defined as: 0 (+) 0 = 0 0 (+) 1 = 1 1 (+) 0 = 1 1 (+) 1 = 0
financial institution	an establishment responsible for facilitating customer-initiated transactions or transmission of funds for the extension of credit, or the custody, loan, exchange, or issuance of money.
financial message	a message containing text or instructions which have financial implications.
insertion	the process of creating a message as a method of perpetrating a fraud.
interchange	mutual acceptance and exchange of messages between financial institutions.

irreversible encryption	DEA transformation of cleartext in such a way that the encrypted text cannot be decrypted back to the original cleartext.
loss	same as deletion.
message	a set of message elements used to support the interchange of information; a communication containing text or instructions.
message authentication code (MAC)	a cryptographically computed number which is the result of passing a message through the authentication algorithm using a specific key. Lengths of from 8 to 16 hexadecimal characters can be used.
message element	a predefined meaning or representation of data within a message.
originator	the person, institution or other entity responsible for and authorized to initiate a protected message. Here the concept of responsibility, e.g. financial responsibility, is paramount.
parity	a measure of the number of "1" bits in a group of "0" and "1" bits; either odd or even.
parity bit	a bit added to a group of "0" and "1" bits to make the parity of the group odd or even.
performance standard	general design criteria defining the desired result without specifying the method of achieving that result.
privacy	the confidential nature of data which requires protection against unauthorized disclosure.
random	a value in a set that has equal probability of being selected from the total population of possibilities, hence unpredictable.
receiver	the person, institution, or other entity receiving a transmitted message.
recipient	the person, institution or other entity responsible for verifying that selected contents of the message have not been altered in transit, as well as validating the authority of the message originator.

replay	the process of sending a previously sent message as a method of perpetrating a fraud.
reversible encryption	DEA transformation of cleartext in such a way that the encrypted text can be decrypted back to the original cleartext.
sender	the person, institution, or other entity transmitting a message.
time-variant value	a value which changes with each message or transaction.
transaction	a series of messages to perform a predefined function.
transaction integrity	the soundness of a transaction flowing through a network to its intended destination without impairing its function, meaning or content.
variant of a key	a new key formed by a process (which need not be secret) with the original key, such that one or more of the non-parity bits of the new key differ from the corresponding bits of the original key.
verification	the process of associating and/or checking a unique characteristic.

2.4 APPLICATION

This standard is applicable for institutions that send or receive electronic financial messages and have established the policy of participating in message security assurance on a standardized interoperability basis in electronic-based interchange.

Mandatory standard techniques and procedures are indicated by the word "must". Guidelines are indicated by the word "should".

2.4.1 Protection Provided

The identity of the originating party is implicitly validated by proper use of the standard. Further, the standard provides a method for protection against accidental and deliberate alteration of messages between originators and recipients.

The standard neither provides for nor precludes the use of encryption for the protection of messages against unauthorized disclosure.

The authentication method, as defined in this standard, must be used to validate selected elements contained in financial messages. Note that the authentication process may include authentication elements which are not contained in the message as message elements. Various applications must further identify the minimum message elements to be included as authentication elements in the authentication process whenever these message elements are present in a message being authenticated.

The standard recommends the authentication of the entire message, but can be applied to a set of authentication elements that do not constitute the entire message. In this case, the protection provided applies only to the selected authentication elements. Other parts of the message are subject to undetected alterations. Since the user's application processing normally operates on message elements and not authentication elements, it is the user's responsibility to ensure that the protection provided by this authentication process extends to the subsequent processing by the application.

Furthermore, this standard does not cover protection against fraudulent replay or deletion of messages. Such protection should be provided by the incorporation of one or more of the following:

- time-variant authentication element(s)
- time-variant key

Such techniques should also be used in a security process that provides closed-loop response integrity.

2.4.2 Message Types

Various applications must further identify which message types -- if not all -- are to be included in the authentication procedures.

Note that the protection provided applies only to those selected message types. Other message types are subject to undetected alterations, with possible consequences to future selected or other message types as well. However, the standard also allows for the authentication of all message types within a given application.

2.4.3 Message Elements

The authentication method, as defined in this standard, must be used to validate selected elements contained in financial messages. Various applications must identify the minimum message elements to be included as authentication elements in the authentication process.

2.4.4 Operations

The MAC is computed from the authentication elements by applying the authentication algorithm with a secret cryptographic key. The authentication algorithm is based on the Data Encryption Algorithm (DEA).

The message authentication process must be performed in an environment which is designed to preclude unauthorized disclosure of the cryptographic key or subversion of the authentication process.

2.4.4.1 Cryptographic Key

A DEA cryptographic key consists of 64 bits: 56 random and independent bits must be used by the authentication algorithm (forming the active key) and 8 bits should be used to detect errors within the key (as in Figure 2.1, bits 8, 16, 24, ..., 64 are for parity checking). The parity (P) of every eight-bit byte should be odd (i.e. the number of 1-bits, including the parity bit, should be odd). Written keys should be represented in 64-bit form as sixteen hexadecimal characters (0-9, A-F).

64-BIT CRYPTOGRAPHIC KEY

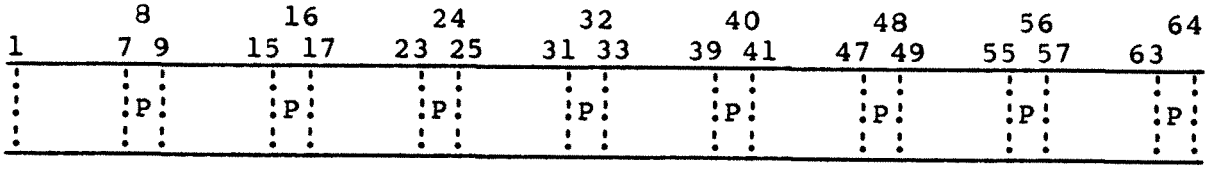


Figure 2.1

2.4.4.2 The Authentication Algorithm

The MAC is computed as illustrated in Figure 2.2 or Figure 2.3. The authentication algorithm may be implemented using either the Cipher Block Chaining (CBC) or 64 bit Cipher Feedback (CFB-64) modes of operation (See Reference 2). The procedures for computation of the MAC via these modes must be executed as outlined in the following sections. These two procedures yield identical results.

(Note: Although these procedures look similar, the initialization vector (IV) in the CFB-64 mode is set to the first 8 bytes of the data to be authenticated. The IV in the CBC mode is set to all zeroes; this effectively is a null operation and has no impact on the final MAC value. Therefore the two modes, for MAC computation purposes, are equivalent.)

An optional procedure is provided (see Section 2.4.4.5) for increased protection against exhaustive key determination. Examples of the MAC computation, both with and without this optional procedure, are given in Appendix C.

2.4.4.3 Cipher Block Chaining Procedure (CBC: Figure 2.2)

A. Initialization

1. Load the DEA processor with the key (K), encrypt mode.

2. Clear the Initial Vector (IV) register to zero (0).

B. Cryptoprocessing

1. With data block (D_1) exclusive-OR'd (XOR) with the all zero IV, execute DEA to obtain output (C_1).

2. With data block (D_2) XOR with ciphertext output (C_1), execute DEA to obtain output (C_2).

.
.
.

[n-1]. With data block (D_{n-1}) XOR with ciphertext output (C_{n-2}), execute DEA to obtain output (C_{n-1}).

[n]. If the final data block (D_n) is less than 64 bits in length, left-justify it and pad with zeros to a full 64 bit length. With the (padded) data block (D_n) XOR with ciphertext output (C_{n-1}), execute DEA to obtain output (C_n).

C. MAC Output

1. The MAC is the left-most m bits of the ciphertext output (C_n), where m is a specified number for the application.

2. If the MAC is written or displayed, it should be represented as hexadecimal characters.

THE AUTHENTICATION ALGORITHM
CIPHER BLOCK CHAINING (CBC)
MAC COMPUTATION PROCEDURE

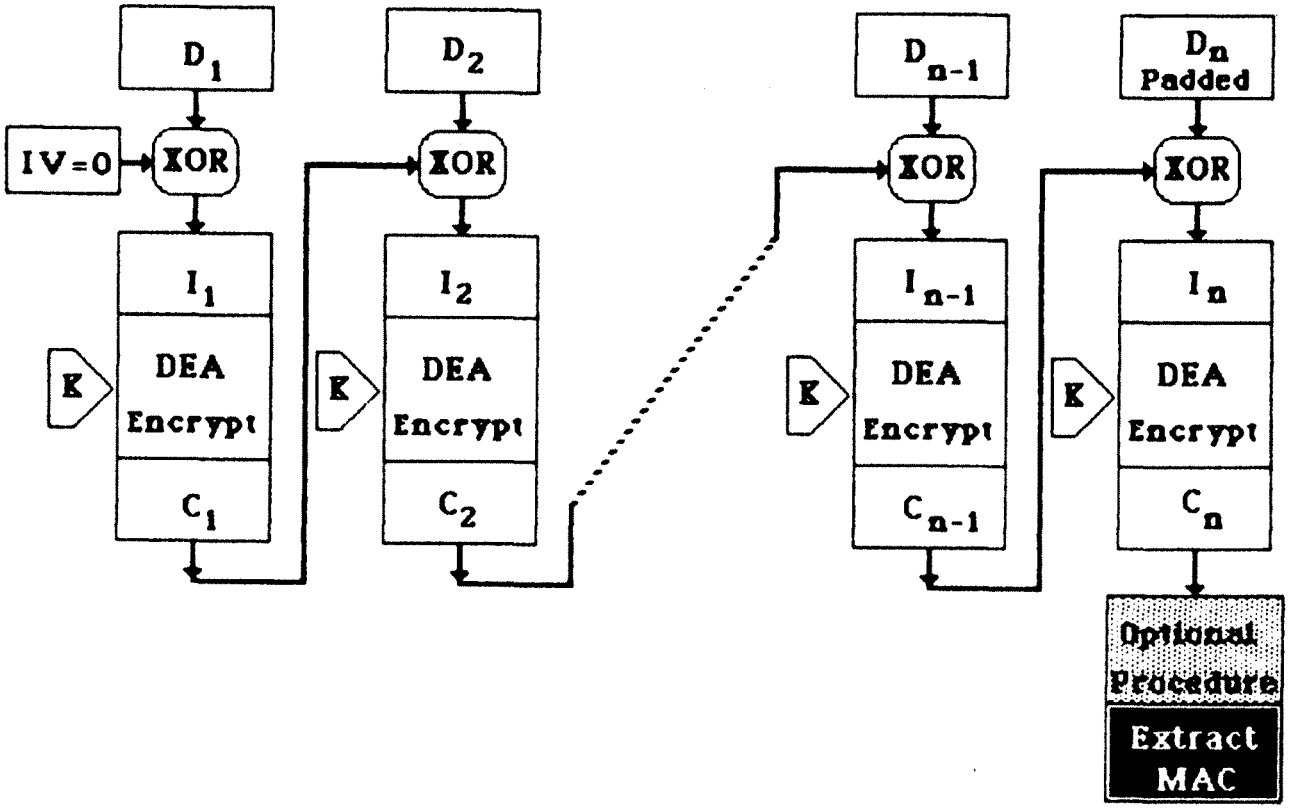


Figure 2.2

2.4.4.4 Cipher Feedback Procedure (CFB-64: Figure 2.3)

A. Initialization

1. Load the DEA processor with the key (K), encrypt mode.

2. Load the 1st data block -- the Initial Vector (IV) -- into the DEA input register (I_1).

B. Cryptoprocessing

1. Execute DEA to obtain output (C_1).

2. With data block (D_2) XOR with ciphertext output (C_1), execute DEA to obtain output (C_2).

.
.
.

[n-1]. With data block (D_{n-1}) XOR with ciphertext output (C_{n-2}), execute DEA to obtain output (C_{n-1}).

[n]. If the final data block (D_n) is less than 64 bits in length, left-justify it and pad with zeros to a full 64 bit length. With the (padded) data block (D_n) XOR with ciphertext output (C_{n-1}), execute DEA to obtain output (C_n).

C. MAC Output

1. XOR the final DEA ciphertext output (C_n) with an extra all zeros data block (D_x) for system access to the ciphertext output (C_n). (Note: For MAC processors that can pass ciphertext output (C_n) directly to external logic, this extra XOR operation is not required.)

2. The MAC is the left-most m bits of the ciphertext output (C_n), where m is a specified number for the application.

3. If the MAC is written or displayed, it should be represented as hexadecimal characters.

THE AUTHENTICATION ALGORITHM
CIPHER FEEDBACK (CFB-64)
MAC COMPUTATION PROCEDURE

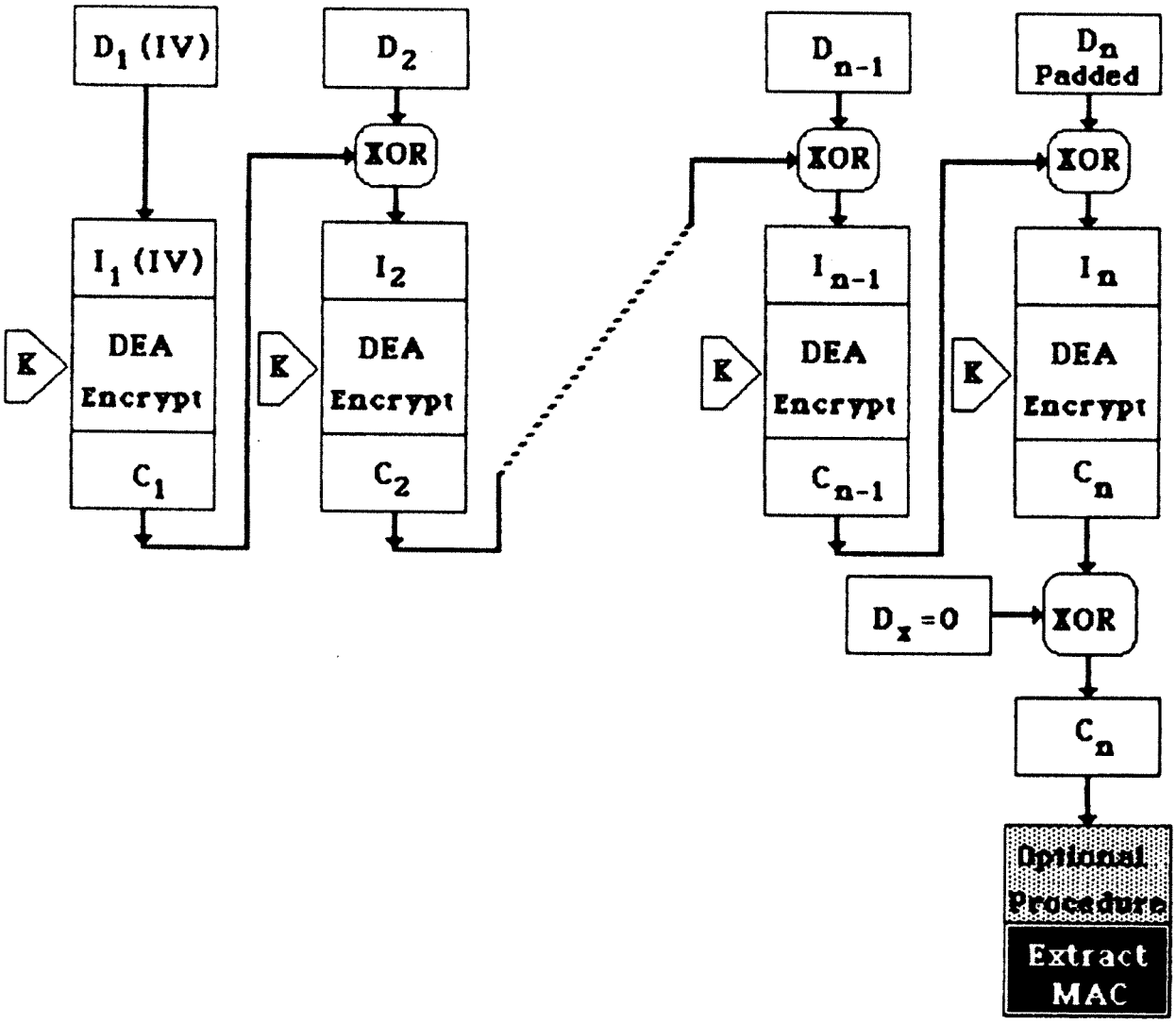


Figure 2.3

2.4.4.5 Optional Enhanced Security Procedure

Increased protection against exhaustive key determination can be provided by the optional use of two DEA keys for MAC computation according to a predefined agreement between originator and recipient. When this option is selected, the first key must be used as described for "the key (K)" in

Section 2.4.4.3 up to step C.1, or
Section 2.4.4.4 up to step C.2.

At those points, the following two additional steps must be inserted (See Figure 2.4):

a1. Decrypt the ciphertext output (C_n) using the second key.

a2. Encrypt the result of Step a1 using the first key. This result becomes the new ciphertext output (C_n'').

The two DEA cryptographic keys used in this procedure must never be used singly for any purpose.

THE AUTHENTICATION ALGORITHM
OPTIONAL ADDITIONAL PROCEDURE TO
PROTECT AGAINST EXHAUSTIVE KEY DETERMINATION

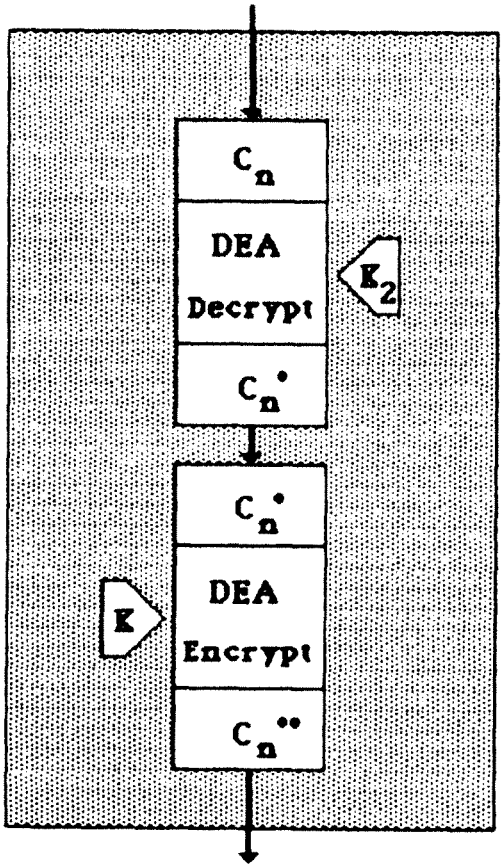


Figure 2.4

2.5 AUTHENTICATION PROCESS

The algorithm is applied either to the whole body of the message or to specific authentication elements, according to a predefined agreement between originator and recipient. The result of the algorithm in either case is a Message Authentication Code (MAC). This code is included in the message. The placement of the MAC must be predefined by agreement. The standard must be used to authenticate financial messages using any communications media.

The originator of a financial message must generate a Message Authentication Code (MAC) by applying the authentication algorithm described above to selected elements using a secret key previously exchanged with the recipient. The MAC must be included in the message forwarded to the recipient.

The recipient must compute the MAC and compare it against the originator's MAC. Equality of the computed MAC to the transmitted MAC received with the message will constitute authentication of the message or authentication elements.

Note that the process of generating the MAC is sensitive to the sequence in which the authentication elements are processed, e.g., a change in the sequence of message elements after the originator's MAC is generated may cause a change in the sequence of authentication elements which will result in a failure to authenticate.

Also note that this standard does not absolutely require that the binary bit streams or authentication elements for entry into the MAC algorithm correspond with the binary bit stream or message elements transmitted in the message text. For example:

- message text is represented in binary but authentication elements are represented in ASCII.

- message text contains three message elements, but MAC computed on five elements -- two of which weren't transmitted in the message text because the recipient already knew them.

Where message processing is automated and the precise content of the body of the message does not change between originator and recipient, the algorithm should be applied to the entire message. Header and trailer message information used for transmission purposes which may be modified or deleted before reaching the recipient must be omitted, i.e., not part of the message body nor included in the algorithm computation. The resultant MAC must be included in the message and, like header and trailer information, must not be processed in the algorithm computation by the recipient.

Where authentication of the entire message is not desired, the algorithm must be applied to the message elements specified to become authentication elements in the predefined agreement between originator and recipient. In this case:

- Authentication elements are taken in the predetermined order.
- Any message element which is used to enable the recipient to determine unambiguously the length or to identify an authentication element must itself be specified as an authentication element.
- The originator selects the unique authentication key for the recipient.
- Any appropriate editing criteria are applied as per the predefined agreement.
- The edited authentication elements are processed by the algorithm.
- The resultant MAC is added to the message before transmission.
- The recipient repeats this process. If its resulting MAC is identical to the MAC it received in the message, the recipient accepts the message elements specified.

Note: The MAC is computed from authentication elements. The user's application processing normally operates on message elements. It is the user's responsibility to ensure that the editing criteria chosen to convert message elements to authentication elements lead to no ambiguity of the values of the message elements presented to subsequent application processing.

Written agreements, controls, and security procedures must exist for secure implementation, use, and protection of the authentication process and devices. Both the originator and recipient must have provisions for:

- MAC "Passes" - resulting MAC was identical
- MAC "Fails" - resulting MAC was not identical

2.6 COMPLEMENTARY SECURITY MEASURES

Although not addressed in this standard, the originator and recipient should include in their security process such features and considerations as:

- closed-loop response integrity
- overall transaction integrity
- and, where appropriate, privacy

Assurance of adequate implementation and operation of message authentication systems is beyond the scope of an ANSI standard.

2.7 KEY MANAGEMENT

Keys must be protected. The generation, distribution, storage, destruction and protection of cryptographic keys is called "key management." Maintaining the secrecy of cryptographic keys is of the utmost importance because the compromise of any key allows the compromise of all messages ever encrypted under it.

The following should be guidance for any key management scheme which is employed in conjunction with this Message Authentication Standard:

- Key Security -- The security of the DEA authentication process is directly dependent on the security afforded to the cryptographic keys. Therefore, the utmost caution must be taken to ensure protection of the keys from compromise throughout their useful lives.

- Generation of Secret Keys -- Care must be taken to ensure that the process used and the resulting keys are random. Biased or insufficiently random techniques lower the security potential of the algorithm. Keys should be changed periodically.

- Key Exchange -- Keys exchanged between originator(s) and recipient(s) must be handled in a secure manner.

- Key Storage -- Keys stored within the authentication equipment must be protected against unauthorized disclosure. Equipment must have design features which resist and detect tampering, erase the stored keys upon tampering and require reinitialization of the equipment.

  Keys stored outside the authentication equipment must be kept in a secure manner (e.g., physical security, under dual control with split responsibility or encrypted using another key for storage), and must be available only to previously authorized personnel.

IT IS IMPERATIVE THAT USERS UNDERSTAND THAT INADEQUATE KEY MANAGEMENT JEOPARDIZES THE PROTECTION PROVIDED BY THIS STANDARD

3. MESSAGE AUTHENTICATION FOR RETAIL TRANSACTIONS

3.1 PURPOSE

Section 3 establishes specific requirements to authenticate financial messages in a retail environment. This section must be combined with Sections 1 and 2 to complete the standard for all retail applications.

3.2 SCOPE

Section 3 outlines additional or detailed methods and encompasses message types, message elements, situations, definitions, etc. The universally applicable method to authenticate financial messages outlined in Section 2 is not repeated. Any sections other than 1, 2 and 3 are not applicable for any financial retail applications.

3.3 DEFINITIONS

For definitions, see Section 2.3.

3.4 APPLICATION

Mandatory standard techniques and procedures are indicated by the word "must". Guidelines are indicated by the word "should".

3.4.1 Protection Provided

For protection provided, see Section 2.4.1.

3.4.2 Message Types

The types of messages to be protected must be determined by predefined agreement between originator and recipient; see Section 2.4.2. Several categories of message types should be considered for protection:

- Transaction-oriented messages
- Operational-oriented messages
- Administrative-oriented messages

3.4.3 Message Elements

The message elements, message element formats, and the order of the message elements specified to become authentication elements must be determined by predefined agreement between originator and recipient; see Section 2.4.3. This agreement must specify the character coding or binary representations as well as any editing rules. Therefore, this standard does not specify particular message elements which must be included as authentication elements in the MAC computation.

All message elements should be included in the MAC computation. The advantages are:

- o maximum protection against message element alteration
- o simplicity of software and system implementation
- o common hardware functions for "dumb" devices

The disadvantage to including all message elements is the possible degradation of total system throughput for performing the MAC computation, including the actual time for computation by encryption component as well as transmission of messages to the component.

If inclusion of all message elements is not desirable, several categories of message elements should be considered for protection:

- o For transaction-oriented messages:
  - identification data
  - verification data
  - transaction description data
  - authorization response data
  - message integrity data
  - transaction integrity data
  - key management data

- o For operational- and administrative-oriented messages:
  - status data
  - audit data
  - message integrity data
  - network control data
  - security data
  - key management data

In addition, any message element which is used to enable the recipient to determine unambiguously the length or to identify an authentication element must itself be specified as an authentication element.

The elements to be included in a system's MAC computation are determined by a trade-off analysis by the implementor as well as by the aforementioned predefined agreement. Inclusion of a particular message element category in the MAC computation diminishes the exposure to attack by substitution of that message element. Other message security functions, such as protection against message replay or message deletion, may require additional elements for inclusion. However, each additional element included in the MAC computation effectively increases the system resources required.

3.4.4 Operations

The predefined agreement between originator and recipient must specify:

- the message types to be protected
- the message elements (or non-message elements) to become authentication elements
- the edit rules to determine authentication elements formats and processing order
- the position and format of the MAC.

For each message type, the same set of authentication elements must always be accessed and concatenated for input to the MAC computation.

The MAC must be 32 bits in length. Note that for a 32 bit MAC, the probability of an adversary contriving the correct MAC for a fraudulently generated or modified message is one in approximately four billion.

For each protected interchange message, the MAC must be computed as a function of specified authentication elements by the originator, and the MAC must be included within the transmitted message for recomputation and verification by the recipient prior to acceptance of the integrity of the message; see Section 2.4.4.

3.5 AUTHENTICATION PROCESS

The authentication process must be determined by predefined agreement between originator and recipient; see Section 2.5.

3.6 COMPLEMENTARY SECURITY MEASURES

Complementary security measures should be determined by predefined agreement between originator and recipient; see Section 2.6.

3.7 KEY MANAGEMENT

Keys must be protected; see Section 2.7. In security systems there is always a concern of using security techniques for purposes which were not intended in order to subvert the system.

The keys used in the authentication process must not be used for any other purpose, e.g. encrypting Personal Identification Numbers (PIN) (see Reference 4). The authentication process for computing the MAC may use other techniques with a high security level, such as:

- a variant of a key used for another purpose
- a key which changes with every transaction.

APPENDIXES

APPENDIX A: MESSAGE AUTHENTICATION TUTORIAL

(This Appendix is not part of American National Standard X9.19-1986, but is included for information only.)

The purpose of message authentication is to ensure that transaction messages are received exactly as originated by the legitimate originator. To accomplish this, message authentication detects both the fraudulent insertion of totally spurious transaction messages, and the fraudulent modification of otherwise legitimate transaction messages.

Message authentication differs from message encryption in that the latter does not inherently protect against modified transactions, whereas the former not only provides this protection, but provides it on the cleartext message, allowing the message to be comprehended, processed and journaled while still protected.

Message authentication is needed to thwart "active wiretapping" and related fraud threats. These are relatively sophisticated threats in which transaction data is modified or inserted in real time, perhaps via a microcomputer system inserted into a communications line. For example, assume that a criminal cuts the communications line from an ATM (which does not use any form of message authentication) to its host, and inserts a microcomputer system in series with this line. To the host, this system "looks" like an idle ATM. To the ATM, this system "looks" like the host. This fraudulently inserted system is programmed to intercept and discard every request-for-cash message originated by the ATM, and in response to send the approval indication. Thus the criminal can readily "drain" the ATM of cash, yet no account will be debited in the process.

The probability of "active wiretapping" is a very subjective question, and depends upon the type of terminal. For example, an unattended currency dispenser would seem to be a highly tempting target for this type of fraud, whereas a conventional credit card authorization terminal would seem to be a less tempting target.

Message authentication thwarts "active wiretapping" fraud scenarios by appending a "message authentication code" to each of the transaction messages. This code consists of some number of check digits, which are analogous to a parity check or cyclic redundancy check except that they are generated via a cryptographic process using the Data Encryption Standard (DES) and a secret key.

The "message authentication code", or "MAC", is generated by the originator of the message, and is based on the entire message, or upon critical elements of the message, as determined by predefined agreement between originator and recipient. (Elements not included in the message but known to both originator and recipient can, by such a predefined agreement, be included in the MAC computation.) The MAC is included in the transmitted message, and then verified by the recipient, who holds the same secret key used in the generation process.

Should anyone attempt to modify the protected message elements between the time the MAC is generated and the time it is checked, his attempt would be detected. Not knowing the secret key, he would be unable to generate the correct MAC for the modified message. Similarly, no one can successfully introduce a spurious message because, not knowing the secret key, he cannot generate the proper MAC for this message.

For message authentication to be effective, the secrecy of the cryptographic key must be assured. Preferably a unique key is used by each communicating pair, so that the compromise of a key jeopardizes the transactions between only two parties, and limits accountability to these two parties.

Although message authentication can detect spurious and modified transaction messages, it cannot inherently detect the fraudulent replay of a previously valid message, nor the loss of a message. See Appendix B for a discussion of these issues.

Message authentication cannot protect against errors in, nor subversion of, the message processing which takes place before the MAC is generated, nor after it has been verified. For example, it cannot protect against a dishonest merchant which modifies its terminal to indicate one value of the transaction to the customer, while causing the customer's account to be debited (and the merchant's account to be credited) by a higher value.

Message authentication can be effectively used by some participants in a retail EFT system even if not used by all. Should an institution decide not to implement message authentication, but later become the victim of an "active wiretapping" fraud scenario, this institution could be made liable for the fraud loss, since transaction journals, etc., would indicate where the transaction was fraudulently modified. Thus each institution participating in the retail EFT system can estimate the implementation cost for message authentication, and the fraud cost for no message authentication, and make its decision accordingly.

This standard has been prepared so that those institutions desiring to implement message authentication can do so in a secure manner, can achieve interchange interoperability with other such institutions, and can find sources of suitable equipment.

APPENDIX B: DUPLICATION AND LOSS PROTECTION EXAMPLES

(This Appendix is not part of American National Standard X9.19-1986, but is included for information only.)

Purpose

Message authentication can be enhanced, as part of an overall security process, to protect against:

- o fraudulent replay or duplication of messages, and
- o fraudulent deletion or loss of messages.

This can be accomplished, in accordance with predefined agreements, with the use of time-variant authentication elements, time-variant keys, or other methods. The following are examples of how duplication and loss of messages may be detected; they use time-variant authentication elements -- date and message sequence identifier (MID). Other methods, including variations of those described below, may also be devised.

Duplication Protection

Duplicated messages may be detected if under normal operation the MID does not repeat for a given date and a given key. The recipient must check the MID to ensure that it did not appear in a previous message. This check may be performed in one of several ways. If MIDs are sent in no predetermined order, then the recipient may compare the received MID against a list of the MIDs received for the day. If the MIDs for messages authenticated under a particular key are always sent in increasing order, and assuming they are received in the same order, the recipient need only check to ensure that the identifiers are strictly increasing.

When more than one originator/recipient pair share a common key, duplication may be detected if each pair uses a mutually exclusive portion of the possible MIDs. The recipient checks that the MID is in the proper range and has not been already received. If the identities of both the originator and recipient are included as authentication elements in each message, the receiver need check only that it is the intended recipient and that the MID has not appeared previously in a message from that originator. In this case, the entire range of MIDs may be used by each originator/recipient pair, and MIDs may repeat between different pairs.

Loss Protection

Loss of a message may be detected if both the originator and recipient keep a list of all MIDs used in a given time period. One party sends its list (via an authenticated message which has duplication protection) to the party wishing to detect any loss. A comparison of the two lists is then performed. Alternatively, if the MIDs are to be received in sequence, the recipient may detect a lost message as soon as an out-of-sequence MID is received. The last MID for the time period may be sent to the loss-detecting party by way of an authenticated message which has duplication protection.

APPENDIX C: MAC COMPUTATION EXAMPLES

(This Appendix is not part of American National Standard X9.19-1986, but is included for information only.)

Purpose

This appendix presents examples of MAC computation using the authentication procedures in this standard. These examples use a transaction-oriented message which could be generated by an ATM and include an encrypted PIN block.

Example 1 uses the entire message for MAC computation. Only the message text (whole body), not the protocol-related fields such as header, are used. Example 2 outlines a MAC computation using only selected fields of the message. Example 3 illustrates the use of two DEA keys as outlined in the optional enhanced security procedure.

Assumed Pre-defined Agreement

The authentication elements are expressed as the hexadecimal representation of ASCII characters (two hexadecimal digits per character). MAC computation uses the cipher feedback CFB-64 procedure.

The resulting hexadecimal MAC is converted into ASCII characters for transmission. Each hexadecimal digit of the MAC is transmitted as an ASCII character from the set 0-9, A-F.

(Note: In other environments, the pre-defined agreement might specify a different representation for the authentication elements, and also for the transmitted MAC. For example, in a bit-oriented protocol, binary representation might be used in both cases, reducing both MAC computation time and MAC transmission time.)

Sample Input Message

All three examples are based on the following input message text (ASCII):

Msg Type	Terminal ID	Time Variant No.	Track-2 Data
1 1 F_S	9 1 8 2 7 3 6 4 5 F_S	F_S 5 8 1 4 3 2 7 6 F_S	F_S ; 1 2 3 4 5 6 7 8

Two Data	Transaction Data	Encrypted PIN Block
9 0 1 2 3 4 5 6 = 9 9 1 2 1 0 0 0 0 ? F_S	0 0 0 1 2 5 0 0 F_S	9 7 8 6 5 3 4 1 2 4 8 7 6 9 2 3 F_S

For clarity, a brief description of the sample message content follows:

Message Type	A code by which the terminal indicates the type of message being sent.
Terminal ID	A number by which the terminal is identified to the network.
Time Variant No.	A number (or value) which changes with each message or transaction.
Transaction Data	A field in which the terminal informs the network of the type and value of transaction requested.
Encrypted PIN Block	A field in which the consumer-entered PIN is transmitted to the network in encrypted form.
Track Two Data	The information encoded on Track 2 of a consumer access card. The Track 2 card data used in the sample input message is shown (ASCII) below.

SS	Primary Account Number	Expry Date	Discretionary Data	ES
;	1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 =	9 9 1 2	1 0 0 0 0	?

MAC COMPUTATION EXAMPLE 1

Example 1 uses the entire message text (which may be regarded as a single authentication element) for MAC computation. For this example, the pre-defined agreement specified inclusion of separators and of all card encoded (Track 2) data from Start Sentinel (SS) through End Sentinel (ES).

Cryptographic Key (Hex):

0 1 2 3 4 5 6 7 8 9 A B C D E F

First Data Block (Hex):

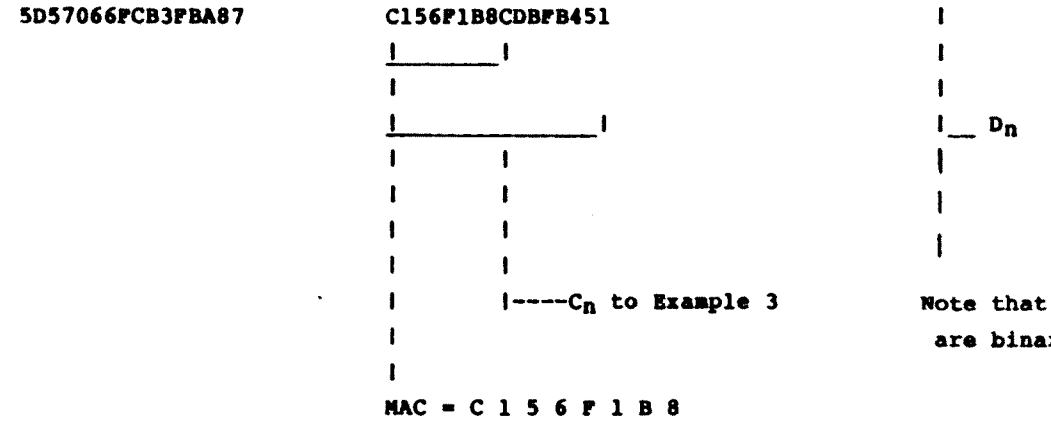
3 1 3 1 1 C 3 9 3 1 3 8 3 2 3 7

(this is the hex representation of ASCII 1 1 F_S 9 1 8 2 7)

EXAMPLE 1

ALL DATA REPRESENTED AS HEXADECIMAL

DEA INPUT BLOCK ($D1/IV$ thru I_n)	DEA OUTPUT BLOCK (C_1 thru C_n)	DATA BLOCKS (D_2 thru D_n)	DATA (+) OUTPUT = FEEDBACK
31311C3931383237	356C20A9E60304D9	333634351C1C3538	065A149CFA1F31E1
065A149CFA1F31E1	BE3EDA28E5A358EA	3134333237361C1C	8F0AE91AD29544F6
8F0AE91AD29544F6	D451B35100C56A84	3B31323334353637	EF60816234F05CB3
EF60816234F05CB3	BCF794DAA6BB0FFE	3839303132333435	84CEA4EB94883BCB
84CEA4EB94883BCB	3622C2A8A5F73F94	363D393931323130	001FFB9194C50EA4
001FFB9194C50EA4	EA776E4F7064C650	3030303F1C303030	DA475E706C54F660
DA475E706C54F660	2ABFE53C0CA6C57D	31323530301C3937	1B8DD00C3CBAFC4A
1B8DD00C3CBAFC4A	OEBF212FA1E0EBB2	3836353334313234	3689141C95D1D986
3689141C95D1D986	65603056F90CA687	3837363932331C00	5D57066FCB3FBA87
5D57066FCB3FBA87	C156F1B8CDBFB451		



Note that last two digits are binary zero-padded.

MAC = C 1 5 6 F 1 B 8

MAC COMPUTATION EXAMPLE 2

Example 2 outlines a MAC computation using only the following selected message elements:

- Time Variant No. (or value)
- Account Number (PAN) from Track 2 of the consumer card
- Transaction Data
- Encrypted PIN Block

For this example, the pre-defined agreement specified inclusion of separators and start sentinels in the authentication elements. The values for Example 2 are (ASCII):

Time Variant No.	Account Number
5 8 1 4 3 2 7 6 F_S ;	1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 =

Transaction Data	Encrypted PIN Block
F_S 0 0 0 1 2 5 0 0 F_S	9 7 8 6 5 3 4 1 2 4 8 7 6 9 2 3 F_S

Cryptographic Key (Hex):

0 1 2 3 4 5 6 7 8 9 A B C D E F

First Data Block (Hex):

3 5 3 8 3 1 3 4 3 3 3 2 3 7 3 6

(this is the hex representation of ASCII 5 8 1 4 3 2 7 6)

EXAMPLE 2

DEA INPUT BLOCK	DEA OUTPUT BLOCK	DATA BLOCKS	DATA (+) OUTPUT = FEEDBACK
3538313433323736	4B89BB374B4A3301	1C3B313233343536	57B28A05787E0637
57B28A05787E0637	93DE1FBEE3254E7B	3738393031323334	A4E6268ED2177D4F
A4E6268ED2177D4F	9C728CDBD801E4B9	35363D1C30303031	A944B1C7E831D488
A944B1C7E831D488	21272D8B48DF5EE2	323530301C393738	13121DBB54E669DA
13121DBB54E669DA	260B3443565D8890	3635333431323438	103E0777676FBCA8
103E0777676FBCA8	0E210B4CDBDCBB15	37363932331C0000	3917327EE8C0BB15
3917327EE8C0BB15	AB4884061A159618		

Note that last four digits are binary zero-padded.

MAC: A B 4 8 8 4 0 6

MAC COMPUTATION EXAMPLE 3

Example 3 is an extension of Example 1, illustrating the use of two DEA keys as outlined in the optional enhanced security procedure. This example utilizes the cipher text output (C_n) from Example 1.

Cryptographic Key 1 (Hex) (same as Example 1):

0 1 2 3 4 5 6 7 8 9 A B C D E F

Cryptographic Key 2 (Hex):

F E D C B A 9 8 7 6 5 4 3 2 1 0

Cipher Text Output (C_n) from Example 1 (Hex):

C 1 5 6 F 1 B 8 C D B F B 4 5 1

Cipher Text Output (C_n') (Hex):

C C C D 3 C 0 8 4 1 F 6 C 7 A B

Cipher Text Output (C_n'') (Hex):

C 2 0 9 C C B 7 8 E E 1 B 6 0 6

MAC: C 2 0 9 C C B 7