SECRET ERROR-CORRECTING CODES (SECC)

Tzonelih Hwang

National Cheng Kung University
Institute of Information Engineering
Tainan, Taiwan, R.O.C.

T.R.N. Rao

University of Southwestern Louisiana
The Center for Advanced Computer Studies
Lafayette, Louisiana 70504

Abstract. A *secret error-correcting coding* (SECC) scheme is one that provides both data secrecy and data reliability in one process to combat with problems in an insecure and unreliable channel. In an SECC scheme, only the authorized user havingsecretly held information can correct channel errors systematically. Two SECC schemes are proposed in this paper. The first is a block encryption using Preparata based nonlinear codes; the second one is based on block chaining technique. Along with each schemes can be secure.

Key words. Algebraic-Code Cryptosystem, Block Chaining, Ciphertext-Only Attach, Chosen-Plaintext Attach, Cryptanalysis, Cryptographic Parameter, Cryptography, Cryptology, Cryptosystems, Data Authenticity, Data Integrity, Data Reliability, Data Secrecy, Data Security, JEEC, Known-Plaintext Attach, SECC, Work Factor.

S. Goldwasser (Ed.): Advances in Cryptology - CRYPTO '88, LNCS 403, pp. 540-563, 1990.

1. Introduction

The demand for **reliable, secure** and **efficient** digital data transmission and storage system has been accelerated by the emergence of large-scale and high speed communication networks. In 1948, Shannon demonstrated that errors induced by a noisy channel or storage medium can be reduced to any desirable level by proper encoding of the information [Shannon 48]. Since Shannon's work, a great deal of developments have contributed toward achieving *data reliability* and the use of coding for error control has become an integral part in the design of modern communication systems and digital computers.

Information transmitted through communication channel or stored in storage system is particularly vulnerable to eavesdropping and tampering. Although information can be protected by several ways (e.g., physical control -- data are stored in physically secure place; or computer system control -- the operating system provides access control mechanisms to check user's authentication), data encryption is the most cost-effective way to provide *data secrecy* [Diffie 76, Wood 81, Denning 82].

As computer communications are expanding to many applications, assurance of both data reliability and data secrecy becomes an important issue. To achieve this purpose, conventionally the first step is to encipher a plaintext (M) into a ciphertext and the second step is to encode the ciphertext into a codeword (C). To recover the plaintext (M), the receiver decodes the received word (C' = C + noise) first and then deciphers the ciphertext (see Figure 1.) Combining these two steps into one may obtain faster and more efficient implementations.

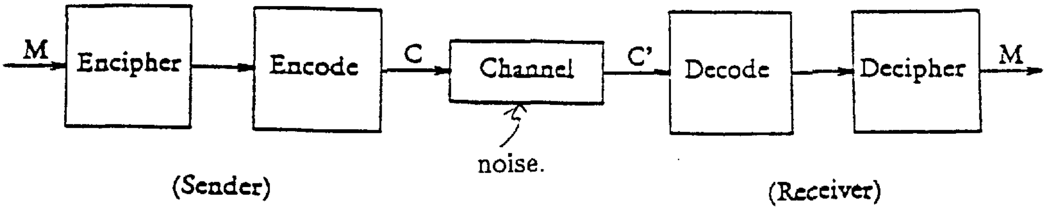

Fig. 1 Conventional approach for data reliability and data secrecy.

1.1 Joint Encryption and Error Correction (JEEC) Scheme

In his public-key cryptosystem, McEliece applied error-correcting capability of Goppa codes to provide data secrecy [McEliece 78]. His idea is to introduce a random error vector to each encoded plaintext before transmission. The Hamming weight (t') of the error vector is equal to the number (t) of errors the code càn correct. Therefore, the receiver can remove the error vector and recover the plaintext by applying the decoding of the code.

If $t' < t$, then up to $t-t'$ errors may occur in the channel and these errors can be corrected by the receiver. Thus, the system provides both data secrecy and data reliability simultaneously. Since the system becomes less secure if t' is small but provides less error correcting capability if t' is large, there is a trade-off between data secrecy and data reliability. This approach, to obtain both data secrecy and data reliability while providing a trade-off between them, is called the *Joint Encryption and Error Correction* (JEEC) scheme [Rao 85].

Definition 1. The JEEC Scheme

A scheme that combines data encryption with data encoding into one process while providing a trade-off between data secrecy and data reliability is called a JEEC scheme.

1.2 Secret Error-Correcting Codes (SECC)

Conventional approach to obtain both data reliability and data secrecy has the disadvantage of inefficiency in the implementation because data encoding and data enciphering are implemented as two different steps. JEEC scheme combines both transformations into one process while providing only a trade-off between data reliability and data secrecy. Large distance and also large block length codes are required in JEEC to combat with problems in an insecure and unreliable channel. However, such codes have low information rates and a relatively high amount of decryption overhead. Therefore, they may not be cost-effective. This leads us to introduce the SECC scheme which may use simple algebraic codes (e.g., $d_{\min} \leq 6$) and also provides both data reliability and data security in one process. The SECC scheme can be defined as follows (see Figure 2).

Definition 2. The SECC Scheme

A scheme that combines data encryption with data encoding into one process to obtain both data secrecy and data reliability, while retaining the **full** error correction capability of the introduced code for possible channel errors, is called an SECC scheme. Also in an SECC scheme, the cryptanalyst is unable to correct channel errors *systematically*. By that we mean it is computationally infeasible for the cryptanalyst to correct channel errors without decoding keys.

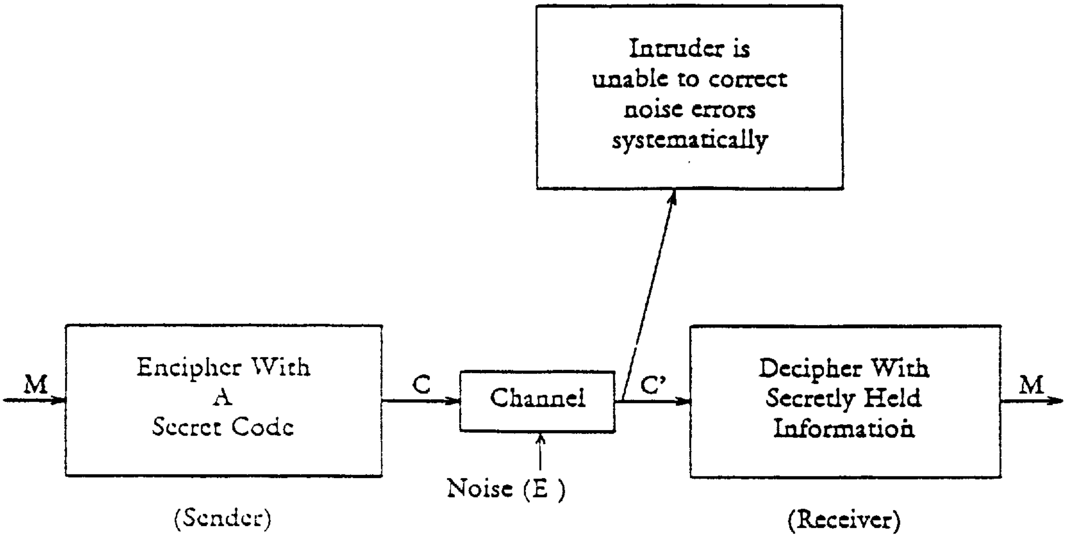
Intruder is unable to correct noise errors systematically
M
Encipher With A Secret Code
C
Channel
C'
Decipher With Secretly Held Information
M
Noise (E)
(Sender)
(Receiver)

Fig. 2 The SECC scheme

Note that JEEC scheme preserves only partial error-correcting ability, whereas SECC scheme preserves full error-correcting ability of the code. Therefore SECC scheme can provide better error correcting capabilities than JEEC does under the use of the same algebraic codes. In a noisy channel, before the plaintext can be recovered the cryptanalyst has to correct channel errors (if any) first. If he cannot correct these errors, then he cannot also recover the plaintext. This is because any uncorrected error in the received ciphertext C' will only generate an M' totally different from the plaintext M (due to the so-called "Avalanche effect" in any good cryptosystem.) Therefore, the presence of noise errors would only increase the security of the system. However, the strength of an SECC system should not depend on the presence of channel errors because they are random in nature. On the other hand, in a conventional system since the coding scheme is public, the cryptanalyst is able to correct channel errors. Therefore, the presence of channel errors doesn't increase the security of the system. We will study an SECC scheme using nonlinear codes in Sec. 2 and an SECC scheme using block chaining technique in Sec. 3. Along with each scheme discussed, we investigate various cryptanalytic attacks to show how the scheme can be secure.

2. SECC Scheme Using Nonlinear Codes

Nonlinear codes with high degree of nonlinearity and whose decoding highly depends on the structure of the codeword, are particularly promising to construct SECC systems. In this section, we investigate Preparata-based nonlinear codes to construct SECC systems. Nonlinear codes, such as Vasil'yev nonlinear codes [Vasil'yev 62] which have only one nonlinear bit in each codeword, are not very useful in this application. We begin with a brief introduction to Preparata codes. Then, we review a code construction technique to construct nonlinear codes with large minimum distances from old codes. Finally, we propose an SECC scheme using nonlinear codes and investigate its security level.

2.1 Preparata Nonlinear Codes [Preparata 68]

Preparata has constructed a class of nonlinear double error-correcting $(2^m-1, 2^m-2m)$ codes, for each even $m \geq 4$, with some interesting features. They contain twice as many codewords as the double error-correcting BCH codes of the same length and they are optimal. Moreover, their decoding can be based on the calculation of syndrome-like quantities and thus the complexity is comparable to the corresponding BCH codes. The encoding and decoding are given here without proof. However, they can be found in [Preparata 68].

Assume that all polynomials discussed here belong to the algebra of polynomials modulo $(x^{2^{m-1}-1}+1)$ over GF(2). Let $\mathrm{B} = \{m(x)\}$ be a single-error-correcting BCH code generated by a primitive polynomial $g_1(x)$ of degree $m-1$ that has a primitive element α as its root. Let $\mathrm{C} = \{s(x)\}$ be the BCH code whose generator polynomial has roots α, α^3, and 1. The polynomial $u(x)$ will denote $(x^{2^{m-1}-1}+1)/(x+1)$. Consider a linear code $\mathbf{C}_n$ given by the vectors of the form

$$\mathbf{v} = [m(x),\ i,\ m(x)+(m(1)+i)u(x)+s(x)], \text{ where } i \in GF(2).$$

$\mathbf{C}_n$ can be shown to be a $(2^m-1, 2^m-3m+1)$ linear code of minimum distance 6.

Let $\phi(x) = (x^{2^{m-1}-1}+1)/g_1(x)$. Then, there exists an s $(0 \leq s \leq 2^{m-1}-2)$ such that $x^s \phi(x) = (x^s \phi(x))^2$. Let $f(x) = x^s \phi(x)$ and $q(x)$ be a monomial of degree less than or equal to $2^{m-1}-2$. $m(x)$, $s(x)$, i, and $q(x)$ are independently chosen. Then, the code $\mathbf{K}_n$ of the form

$$\mathbf{w} = [m(x)+q(x),\ i,\ m(x)+q(x)f(x)+(m(1)+i)u(x)+s(x)]$$

is an $(n, k) = (2^m-1, 2^m-2m)$ Preparata nonlinear code of distance 5. To encode a (2^m-2m)-bit information, the first $(2^{m-1}-m)$ bits are encoded into $m(x)$; the next $(2^{m-1}-2m)$ bits are encoded into $s(x)$; the following one bit is interpreted as i and the last $(m-1)$ bit are encoded into $q(x)$.

For decoding, assume that the vector $\mathbf{w}$ was sent and that the vector

$$\mathbf{r} = [r_0(x),\ r,\ r_1(x)] = \mathbf{w} + [e_0(x),\ e,\ e_1(x)]$$

is received. Given the following definitions

$$H_1 = [\alpha^{2^{m-1}-2}, \alpha^{2^{m-1}-3}, \cdots, \alpha, 1]$$

$$H_3 = [(\alpha^3)^{2^{m-1}-2}, (\alpha^3)^{2^{m-1}-3}, \ldots,(\alpha^3), 1]$$

$$U = [1, 1, \ldots,1, 1]$$

the syndrome $S = (\sigma_0, \sigma_1, \sigma, d)$ of **r** can be computed in the following manner.

$$\sigma_0 = r_0(x)H_1^T = a\,\alpha^p + e_0(\alpha)$$

$$\sigma_1 = r_1(x)H_1^T = a\,\alpha^p + e_1(\alpha)$$

$$\sigma = (r_0(x) + r_1(x))H_3^T = a\,\alpha^3 + e_0(\alpha^3) + e_1(\alpha^3)$$

$$d = r + r_1(x)U^T = e + e_1(1)$$

where $q(x) = ax^p$ is the monomial in the codeword.

Let $\rho = \sigma + (\sigma_0 + \sigma_1)^3$. If $\rho = \sigma_j{}^3$ $(j = 0,1)$ and $d = 0$, then **r** is a member of the nonlinear code. If the above condition is not true, then let $\mathbf{c} = [c_0(x), c, c_1(x)]$ be the "correction" vector that can be added to **r** to get the codeword **w**. The vector **c** can be found by the following rules, if j is taken modulo 2.

Rule 1: If $\rho = \sigma_j{}^3$ and $\rho \neq \sigma_{j+1}$ then $c_{j+1}(x) = x^k$ where $\alpha^k = \sigma_0 + \sigma_1$ and $c = d + c_1(1)$.

If $\rho \neq \sigma_j^3$, then we have the following rules.

Rule 2: If $d = 1$ then $c = 0$ and $c_j(x) = x^{k_j}$ where $\alpha^{k_j} = \sigma_{j+1} + \sqrt[3]{\sigma + \sigma_0\sigma_1(\sigma_0 + \sigma_1)}$.

Rule 3: If $d = 0$ and $\sigma_0 + \sigma_1 \neq 0$ then set $c = 0$, $c_j(x) = 0$, and $c_{j+1}(x) = x^{k_1} + x^{k_2}$ where α^{k_1} and α^{k_2} are the solutions of

$$z^2 + (\sigma_0 + \sigma_1)z + (\rho + \sigma_j^3)/(\sigma_0 + \sigma_1) = 0$$

Rule 4: If $d = 0$ and $\sigma_0 + \sigma_1 = 0$ then **r** is at a distance ≥ 3 from any codeword.

If Preparata codes are to be used in the cryptosystems, then the only practical values of m are 6, 8 and 10. Therefore, the codes that will be considered here are (63, 52), (255, 240) and (1023, 1004) Preparata codes.

2.2 Construction of New Code From Old Codes [MacWilliams 77]

We have given a very brief introduction to Preparata nonlinear codes which can only correct double errors. In this section, we review a code construction technique to construct nonlinear codes that can correct more than two errors.

Let $\mathbf{K}_i$ be an (n_i, N_i, d_i) code where n_i is the code length, N_i is the number of codewords and d_i is the minimum distance between any two codewords of the code $(1 \leq i \leq 2)$. A new code $\mathbf{K}'$ can be constructed from both $\mathbf{K}_1$ and $\mathbf{K}_2$, called the base codes of $\mathbf{K}'$ here, as follows.

$$\mathbf{K}' = \left\{ (\mathbf{E}_1(M_1),\ \mathbf{E}_1(M_1)+\mathbf{E}_2(M_2)) \right\},$$

where $\mathbf{E}_i$ is the encoding of $\mathbf{K}_i$ and $M=(M_1, M_2)$ is a plaintext block which is divided into two subblocks M_1 and M_2 over GF(2). Then, $\mathbf{K}'$ is a $(2(max\{n_1, n_2\}), N_1 \cdot N_2, d=min\{2d_1, d_2\})$ code [MacWilliams 77]. If $n_1 \neq n_2$, then enough zeros can be added to the end of the shorter code. Note that $\mathbf{K}'$ is a linear code if and only if both $\mathbf{K}_1$ and $\mathbf{K}_2$ are linear codes, otherwise it is a nonlinear code. This procedure can be iterated to construct nonlinear codes with large minimum distances. Here, we suggest the use of Preparata codes as base codes to construct new nonlinear codes, or we assume that either $\mathbf{K}_1$ or $\mathbf{K}_2$ or both are nonlinear codes constructed from Preparata codes. The decoding of the newly developed code is rather straightforward and is omitted here.

2.3 Encryption and Decryption of SECC Scheme Using Nonlinear Codes

The SECC scheme using nonlinear codes is a block encryption and error correction combined into one that also preserves the full error correction capability of the code for possible channel errors. Each block is enciphered and deciphered independently under this scheme.

Encryption

Let $\mathbf{E}_{K^*}$ denote the encoding of a nonlinear code that encodes a k-bit information into an n-bit codeword. Let Ψ be an invertible function that transforms a k-bit block into a k-bit block in either a linear or nonlinear manner. The matrix $\mathbf{P}$ is a random permutation matrix of size n. A k-bit plaintext block (M) is enciphered into an n-bit ciphertext (C) by the following equation

$$C = \mathbf{E}_{K^*}(\Psi(M))\cdot\mathbf{P}. \tag{1}$$

The cryptographic parameters that are secretly held in the system are Ψ, $\mathbf{P}$ and $\mathbf{E}_{K^*}$. Since ciphertext-only attacks are much weaker attacks than known-plaintext or chosen-plaintext attacks, constructing a cryptosystem which can withstand ciphertext-only attacks is considered to be much easier than constructing a cryptosystem which can withstand either known-plaintext or chosen-plaintext attacks. In the proposed scheme, we assume that the function Ψ can withstand ciphertext-only attacks and may be broken by a known-plaintext attack. Hence the security of the scheme should depend on the strength of the combination of functions Ψ, $\mathbf{E}_k$ and $\mathbf{P}$ and not on the strength of either Ψ or $\mathbf{E}_k$ or $\mathbf{P}$ alone. This also illustrates the difference between SECC and the conventional approach to provide both data secrecy and data reliability.

Decryption

Let $\mathbf{D}_{K^*}$ be the decoding of the nonlinear code and E_i be a correctable error vector which occurs due to channel noise when the i-th ciphertext block is transmitted. The deciphering procedure is given below.

(1) Remove the permutation matrix $\mathbf{P}$ ($\mathbf{P}^T$ is the transpose matrix of $\mathbf{P}$).

$(C+E_i)\cdot\mathbf{P}^T = \mathbf{E}_{K^*}(\Psi(M))+E_i\cdot\mathbf{P}^T$.

(2) Decoding.

$\mathbf{D}_{K^*}(\mathbf{E}_{K^*}(\Psi(M))+E_i\mathbf{P}^T) = \Psi(M)$.

(3) Recover the plaintext M.

$M = \Psi^{-1}(\Psi(M))$.

Notice that the error-correcting capability of the code is fully preserved to correct channel errors (E_i's) as a property required in an SECC scheme. Since the decoding of Preparata codes highly depends on the structure of the codeword, cryptanalyst cannot correct channel errors without knowing the matrix $\mathbf{P}$. This is another property required in an SECC scheme.

2.4 Security of SECC Scheme Using Preparata-Based Nonlinear Codes

We have discussed both the enciphering and deciphering of the SECC scheme using Preparata-based nonlinear codes. What remains to be studied is the security of the scheme. For simplicity we investigate the security of the SECC scheme using Preparata codes mainly. The security of the SECC scheme using extended nonlinear codes follows directly. Let $\mathbf{E}_p$ and $\mathbf{D}_p$ represent the encoding and decoding of a Preparata code respectively.

As we mentioned earlier that the function Ψ can either be a linear or a nonlinear transformation. If the system using a linear function Ψ could provide an acceptable level of security ($\approx 2^{60}$ operations), then the system could provide even a better security if Ψ is a nonlinear function.

First, we consider the case that both Ψ and P are removed from the original scheme. In the following lemma, we shall show that the simplified scheme can be broken by a known-plaintext attack. For this discussion, we assume that no error occurs in the channel.

Lemma 1.

The encryption scheme

$$C = \mathbf{E}_p(M)$$

can be broken by a known-plaintext attack in $O(n^2)$ *bit operations.*

<Proof>

The generator polynomial $g_1(x)$ can be derived from a pair of plaintext and ciphertext as follows. The cryptanalyst obtains $q(x)$ from the last m-1 bits of the plaintext. Hence, $m(x)$ can be computed from the first part (2^{m-1}-1 bits) of the ciphertext. Subsequently, he can derive $g_1(x)$ from $m(x)$ and the first (2^{m-1}-m) bits of the plaintext under a known-plaintext attack. Obviously, this requires only $O(n^2)$ bit operations.

Q.E.D

Let $N(g_1)$ denote the number of primitive polynomials ($g_1(x)$'s) in a class of Preparata codes of a given code length (n). Then, $N(g_1)$ can be computed by the formula $N(g_1)=[\frac{\theta(2^{m-1}-1)}{m-1}]$, where θ is the Euler totient function and m-1 is the degree of the primitive polynomial $g_1(x)$. Therefore, we have

$N(g_1)=2$ if $n=15$,

$N(g_1)=6$ if $n=63$,

$N(g_1)=18$ if $n=255$,

$N(g_1)=48$ if $n=1023$.

The number of choices of the primitive polynomials $g_1(x)$'s in Preparata codes of practical lengths is too small for the simplified scheme to be secure.

We may introduce a secret, linear function Ψ to scramble the plaintext before encoding. However, the modified system is still insecure under a chosen-plaintext attack as can be shown in the following lemma.

Lemma 2.

The encryption scheme

$$C = \mathbf{E}_p(\Psi(M))$$

can be broken by a chosen-plaintext attack in $O(n^2)$ bit operations.

<Proof>

Let M_1, M_2 and M_3 be the three plaintext blocks to be enciphered in the system where $M_1 = M_2 + M_3$. Let C_1, C_2 and C_3 be their ciphertexts respectively. Then,

$$C_1 + C_2 + C_3 = (q_1(x) + q_2(x) + q_3(x),\ 0,\ q_1(x)f(x) + q_2(x)f(x) + q_3(x)f(x)),$$

where $q_i(x)$ is a monomial whose power$\overset{i,j}{\wedge}$ is taken from the decimal equivalent of the last m-1 bits of the scrambled plaintext $\Psi(M_i)$. Let $q_i(x)=0$ if $j=2^{m-1}-1$. Consequently, $f(x)$ can be derived from the first 2^{m-1}-1 bits and the last 2^{m-1}-1 bits of the ciphertext in $O(n^2)$ bit operations. Once $f(x)$ is obtained, $g_1(x)$ can be derived easily (see Sec. 2.1). Therefore, the security of the system totally depends on the strength of the function Ψ which, unfortunately, can be broken by a known-plaintext attacks as mentioned previously.

Q.E.D.

The simplified scheme in Lemma 2 is insecure because the structure of the code is revealed. Therefore, the cryptanalyst can remove the linear component of the codewords and then break the system. In order to avoid this weakness, a permutation matrix may be introduced to scramble the structure of the code. However, the following lemma shows that the modified scheme can be broken by a chosen-plaintext attack if the function Ψ is not introduced to the scheme.

Lemma 3.

The encryption scheme

$$C = \mathbf{E}_p(M)\cdot\mathbf{P}$$

can be broken by a chosen-plaintext attack in $O(n^3)$ operations.

<Proof>

Let $\mathbf{M} = \left[\begin{array}{ccc|cccc} 0 & \cdots & 0 & 0\,0 & \cdots & 0\,1 \\ 0 & \cdots & 0 & 0\,0 & \cdots & 1\,0 \\ 0 & \cdots & 0 & 0\,0 & \cdots & 1\,1 \\ & & & \cdots & & \\ 0 & \cdots & 0 & 1\,1 & \cdots & 1\,1 \end{array}\right]_{(2^{m-1}-1)\times(2^m-2m)}$ be a matrix of plaintexts.

Let M_i $(1 \leq i \leq 2^{m-1}-1)$, the i-th row in M, be the i-th plaintext to be enciphered in the system. Let $\mathbf{C}_M$ denote the matrix of ciphertexts of $\mathbf{M}$. Then, the following relation holds where $\mathbf{T}_M$ is the matrix of codewords of $\mathbf{M}$ encoded by the Preparata code.

$$\mathbf{C}_M = \mathbf{T}_M\cdot\mathbf{P} = \begin{bmatrix} 0\,0 & \cdots & 0\,1\,0 & f(x) \\ 0\,0 & \cdots & 1\,0\,0 & x^1 f(x) \\ & \cdots & & \cdots \\ 1\,0 & \cdots & 0\,0\,0 & x^{2^{m-1}-2} f(x) \end{bmatrix}_{(2^{m-1}-1)\times(2^m-1)} \cdot\ \mathbf{P}.$$

Notice that the columns in the matrix $\mathbf{T}_M$ are all distinct. Therefore, by trying all possible $f(x)$'s (i.e., $N(g_1)$ of them), the cryptanalyst can obtain both the function $f(x)$ and the permutation matrix $\mathbf{P}$ used in the system. The work factor of this attack is dominated by the overhead of enciphering $2^{m-1}-1$ chosen plaintexts, i.e. $O(n^3)$.

Q.E.D.

From these lemmas, we see if both the function Ψ and the permutation matrix $\mathbf{P}$ are introduced to the system as a portion of the key, then these attacks cannot break the resulting scheme.

Since there are only a small number of primitive polynominals for a given code length n, the cryptanalyst may try to guess the generator polynomial $g_1(x)$ used in the system. However, the work factor to check the correctness of each guess involves a very large amount of overhead to figure out both functions Ψ and $\mathbf{P}$. That is a hopeless task.

The SECC scheme using Preparata codes is a block encryption and error correction combined into one that also preserves the full error correction capability of the code for possible channel errors. This is a major distinction from McEliece's scheme, which has no error correcting capability or has only a partial error correcting capability when used as JEEC. While somewhat simpler SECC schemes given by Lemmas 1-3 are shown to be breakable under known-plaintext or chosen-plaintext attacks, the proposed scheme with both functions of Ψ and $\mathbf{P}$ appears to be secure. It would be a challenge indeed to find cryptanalytic attacks to break this scheme.

These attacks are performed under the assumption that there is no error occurs in the channel. If there exist channel errors, then it will be much more difficult to perform these attacks against the SECC system. Therefore, the presence of channel errors introduces additional level of data security to the system as required in an SECC scheme.

There are several types of cryptanalytic attack against algebraic-code cryptosystems discussed in [Rao 87, Struik 87]. These attacks are performed based on the linearity of the system. They will not be applicable for this nonlinear coding scheme.

3. SECC Scheme Using Block Chaining Technique

In this section, we proposed an SECC scheme based on block chaining technique. In this scheme because each ciphertext is a function of all previous plaintexts, decoding error of one ciphertext will propagate all the way through the last block. This "error propagation" property can be applied to detected any illegal modification to the ciphertext thus provides data integrity [Meyer 82]. Therefore, this scheme can provide not only data reliability and data secrecy but also data integrity in one enciphering. But any decoding error requires the retransmission of all blocks chained together.

3.1 Encryption and Decryption of the Proposed Scheme

Rao and Nam have suggested a private-key algebraic code cryptosystem (Rao-Nam scheme) using *simple* linear codes [Rao 87]. By *simple* codes we mean small distance codes,.i.e. $d_{\min} \leq 6$. In this scheme, a k-bit plaintext block M_i is enciphered to an n-bit ciphertext block C_i by the following equation.

$$C_i = (M_i \mathbf{SG} + Z_i)\mathbf{P},$$

where

$\mathbf{S}$: an arbitrary $(k \times k)$ nonsingular matrix,

$\mathbf{G}$: an (n, k) code generator matrix,

$\mathbf{P}$: a random $(n \times n)$ permutation matrix,

Z_i : an error vector of length n randomly selected from a predetermined syndrome-error table.

$\mathbf{S}$, $\mathbf{G}$ and $\mathbf{P}$ are private keys.

Struik and van Tilburg proposed chosen-plaintext attacks (ST-type attacks) on Rao-Nam scheme. Their attacks are based on estimating the rows of the encipher matrix $\mathbf{G'} = \mathbf{SGP}$ by constructing unit vectors from the chosen plaintext or by solving a set of linear equations [Struik 87]. They also proposed a modified scheme to withstand these attacks. In their modified scheme, the matrix **S** in Rao-Nam scheme is replaced by an invertible, nonlinear function f such that $M_i = f^{-1}(f(M_i, Z_i), Z_i)$. In the modified scheme, M_i is enciphered into C_i by the following equation.

$$C_i = f(M_i, Z_i)\mathbf{GP} + Z_i .$$

These schemes are proposed mainly for providing data secrecy. They are not designed to realize JEEC or SECC and therefore do not provide data reliability. However, by modifying the way the error vectors (Z_i's) is introduced, an SECC system can be constructed. Block chaining technique will be applied to facilitate this construction. The proposed system is described below and is shown in Figure 3.

Encryption.

The cryptographic parameters (that are secretly held) for this scheme are

f : an *invertible*, nonlinear function which transforms a k-bit block to a k-bit block,

G : an (n, k) code generator matrix,

g : a k-bit to n-bit block expanding function.

The following symbols are used for this scheme.

X_i : the i-th output of f, $(i = 1,2,...)$.

Z_i : the i-th error vector, $Z_{i+1} = g(X_i)$. Z_1 is a correctable error randomly generated by the system.

E_i : error vector due to channel noise occurs when the i-th block is transmitted.

$C_i' = C_i + E_i$ is the i-th block received at the receiver end.

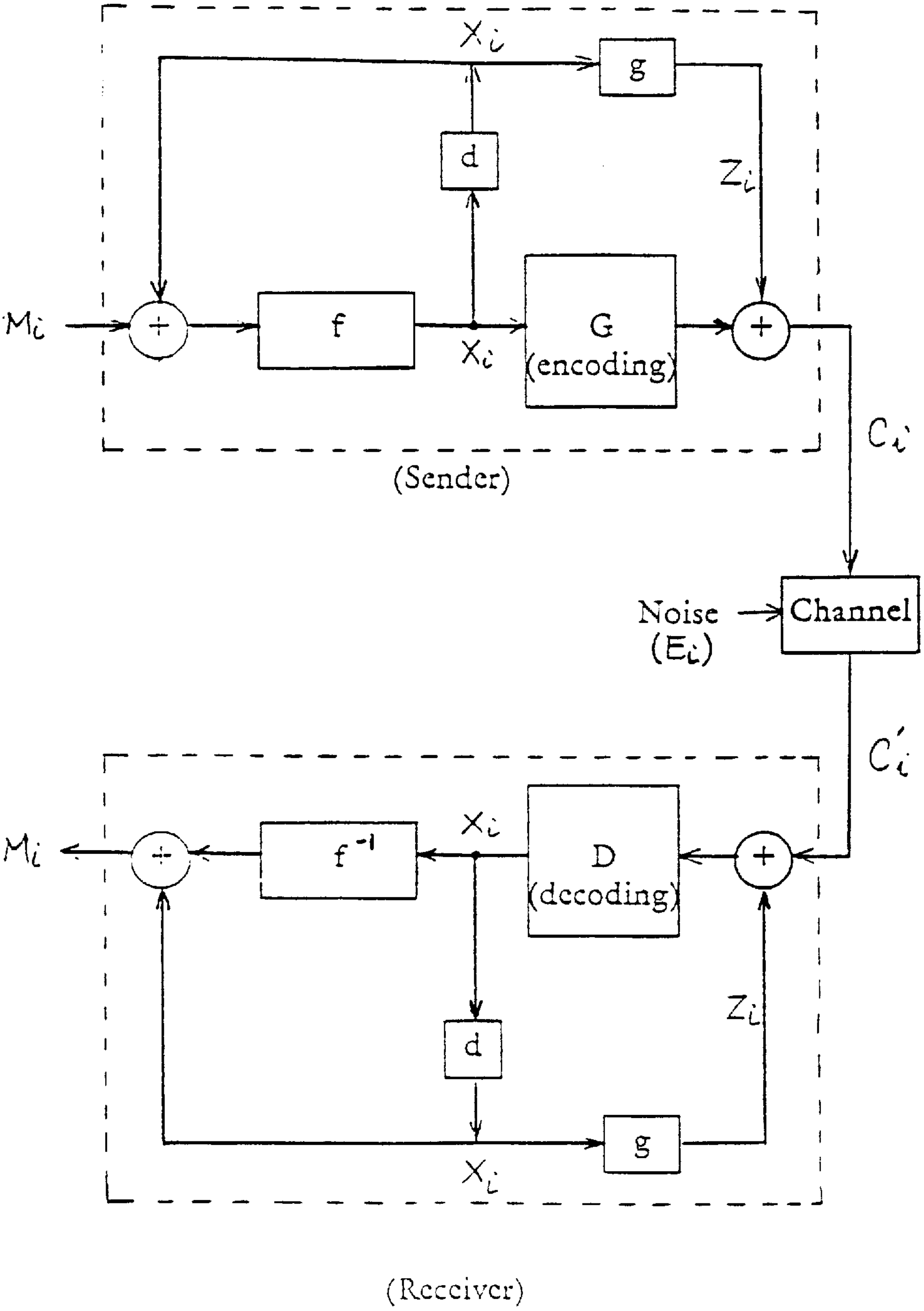

Fig.3 SECC Scheme Using Block Chaining Technique

D : the decoding of the introduced code.

δ : a one-cycle delay register used to store X_i.

The data stream consists of k-bit blocks $M_1, M_2, \cdots M_l$. At time 1, the first plaintext M_1 is transformed into X_1 by the function f. Then, X_1 will be stored at the register δ and also encoded by **G** simultaneously. Before the first codeword $X_1\mathbf{G}$ is sent to the receiver, a randomly generated errors Z_1, that can be corrected by the code, is added to the codeword $X_1\mathbf{G}$. The result, $C_1=X_1\mathbf{G}+Z_1$, is the first ciphertext transmitted to the receiver. At time i, the plaintext M_i is exclusive-oring with X_{i-1} taken from the register δ. Then, $X_i=f(M_i+X_{i-1})$ is computed and stored at δ. At that time, $Z_i=g(X_{i-1})$ is also computed. After $X_i\mathbf{G}$ is obtained, the ciphertext $C_i=X_i+Z_i$ can be constructed. In general, the encryption sequence is given as follows.

$$C_1=f(M_1)\mathbf{G}+Z_1,$$

$$C_2=f(M_2+X_1)\mathbf{G}+Z_2, \quad (X_1=f(M_1),\ Z_2=g(X_1)),$$

$$C_3=f(M_3+X_2)\mathbf{G}+Z_3, \quad (X_2=f(M_2+X_1),\ Z_3=g(X_2)),$$

$$\cdots$$

$$C_l=f(M_l+X_{l-1})\mathbf{G}+Z_l, \quad (X_{l-1}=f(M_{l-1}+X_{l-2}),\ Z_l=g(X_{l-1})).$$

Due to the block chaining feature, the same plaintext blocks will be enciphered to different ciphertexts. Therefore, the cryptanalysis would be more difficult. Since the ciphertexts are not codewords, the cryptanalyst cannot construct a combinatorially equivalent generator matrix of the code from the ciphertexts. Therefore, he cannot correct errors systematically as required for an SECC scheme.

Decryption.

Here, we assume that the receiver could synchronize with the sender on the sequence of vectors X_i and Z_i added to both plaintext and the corresponding codeword respectively. Furthermore, we assume that the decoding is correctly carried out. The decryption sequence is given below.

$$D(C_1'\ )=X_1=f\ (M_1),\quad f^{-1}(X_1)=M_1,$$

$$D(C_2'\ +g\,(X_1))=X_2=f\ (M_2+X_1),\quad f^{-1}(X_2)+X_1=M_2,$$

$$\cdots$$

$$D(C_t'\ +g\,(X_{t-1}))=X_t=f\ (M_t+X_t\ ),\quad f^{-1}(X_t\ )+X_{t-1}=M_t\,.$$

Because the errors introduced deliberately at the sender end can be removed at the receiver end by this synchronization, the error-correcting capability of the code is fully preserved for possible channel errors. By this chaining feature, errors due to intruder's tampering which cannot be corrected by the code will propagate all the way through the last block. However, this may serve as a checksum to detect illegal modification to the ciphertext by the intruder [Denning 82]. Hence the proposed scheme provides not only data reliability and data secrecy but also data integrity (data authenticity) [Meyer 82]. That is, the SECC scheme using block chaining technique can provide two levels of error control. The first level is the correction of channel errors; the second level is the detection of uncorrectable modification to the ciphertext by the intruder. But, the presence of such errors requires the retransmission (or reenciphering) of all blocks chained together.

3.2 Security of the Proposed SECC Scheme

If we define errors in one block of binary information as the bits different from the original block sent by the sender, then both channel errors and intruder's tampering are regarded as errors. However, the manner of the errors introduced by channel noise is different from that of intruder's tampering. The errors introduced by intruder's tampering are primarily multiple errors. In binary symmetric channels, the probability of multiple random errors is very small [Lin 83]. Algebraic codes are designed for correcting random errors due to channel noise. They are not designed to correct multiple errors due to intruder's tampering. In the presence of multiple errors, erroneous decoding might occur. Consequently, to combat with problems in an *insecure and unreliable* channel, a scheme which is capable of hiding information, correcting channel errors and also detecting any illegal modification to the ciphertext is desirable. The SECC scheme based on block chaining technique could provides these characteristics and hence is very useful in an insecure and unreliable environment.

The SECC scheme withstands ST-type chosen-plaintext attacks because of the plaintext is transformed by a nonlinear function f before encoding and also because of the chaining feature. This prevents the cryptanalyst from constructing unit vectors from the chosen plaintexts to derive **G**.

Simplified versions of the SECC Scheme

To show how this scheme provides a high level of security, we may consider two simplified versions of the original one. First, if X_i, the output of f, is fed forward to the function g only (i.e., X_i is not fed back to f), then the encryption sequence is given as follows.

$$C_1 = f\,(M_1)\mathbf{G} + Z_1,$$

$$C_2 = f\,(M_2)\mathbf{G} + g\,(f\,(M_1)),$$

$$\cdots$$

$$C_l = f\,(M_l)\mathbf{G} + g\,(f\,(M_{l-1})).$$

A chosen-plaintext attack can break **G** if g is a public linear function that has a left inverse. For example, let $M_i = M_{i+1}$ and $M_{i+2} = M_{i+3}$. Then

$$C_{i+1} + C_{i+2} = (f(M_{i+1}) + f(M_{i+2}))\mathbf{G}, \text{ and}$$

$$C_{i+2} + C_{i+3} = g(f(M_{i+1})) + g(f(M_{i+2})).$$

Thus $(f(M_{i+1}) + f(M_{i+2})) = g^{-1}(C_{i+2} + C_{i+3})$. If the cryptanalyst could obtain k such distinct pairs, then $\mathbf{G}$ can be derived. However, if g is a secret nonlinear function or g has no left inverse then this attack does not work.

On the other hand, if X_i is fed back to f only (i.e., X_i is not fed forward to g), then the encryption sequence is given as follows.

$$C_1 = f(M_1)\mathbf{G},$$

$$C_2 = f(M_2 + X_1)\mathbf{G}, \quad (X_1 = f(M_1)),$$

$$\cdots$$

$$C_l = f(M_l + X_{l-1})\mathbf{G}, \quad (X_{l-1} = f(M_{l-1} + X_{l-2})).$$

To attack the scheme, the cryptanalyst may find the equivalent ciphertexts. For example, if $C_i = C_j$, then $f(M_i + X_{i-1}) = f(M_j + X_{j-1})$ i.e., $X_i = X_j$. If f is a linear transformation, then $C_{i+1} + C_{j+1} = f(M_{i+1})\mathbf{G} + f(M_{j+1})\mathbf{G}$. Thus, $f \cdot \mathbf{G}$ can be figured out by a known plaintext attack.

If f is a nonlinear transformation, then this line of attack may not work. However, the cryptanalyst could collect k linearly independent codewords to construct a generator matrix ($\hat{\mathbf{G}}$) which is combinatorially equivalent to $\mathbf{G}$. Let $\hat{\mathbf{G}} = \mathbf{S}^{-1}\mathbf{G}$ for any nonsingular matrix $\mathbf{S}$ of rank k. Since the number of nonsingular matrices of rank k is about $0.3 \times 2^{k^2}$, it is computationally infeasible to estimate the matrix $\mathbf{G}$ used if k is large enough. Thus, the scheme appears secure. But, the cryptanalyst may be able to correct channel errors if t is small (e.g. $t \leq 3$). Thus, it is important to feed X_i forward to g in order to construct an SECC system. As a result, the SECC scheme can be very secure if f is an invertible, nonlinear function and g is a nonlinear, one-way function. It will be a challenge to design other lines of attack to break this scheme.

4. Conclusion

For the very first time, we introduce the concept of secret error-correcting codes in this paper. An SECC scheme combines data encoding with data encryption into one process and enables the system to correct channel errors as well as conceal information from unauthorized user simultaneously. The main purpose of this research is to construct SECC schemes to facilitate a *reliable, secure* and *efficient* digital transmission.

We have proposed two SECC schemes to realize this new concept. The first one is a block encryption using Preparata-based nonlinear codes. In this scheme, each block can be enciphered and deciphered independently.

The other SECC scheme is based on block chaining technique. This scheme provides not only data secrecy and data reliability but also data integrity due to the chaining feature. However, the decryption of each ciphertext cannot be carried out independently. The decoding error in one block requires retransmission of all blocks chained together.

Although we have investigate various cryptanalytical attacks against these schemes, they are still not fully proven systems. Several problems relating to the proposed schemes, such as the key generation and key management problems, still remain unsolved. Furthermore, there may exist other good techniques to realize the SECC concept. These indeed require further research.

References

[Denning 82] Dorothy E. Denning, *Cryptography and Data Security* Addison Wesley, 1982.

[Diffie 76] Whitfield Diffie and Martin E. Hellman, "New Directions in Cryptography," *IEEE Trans. on Information Theory,* Vol. IT-22, No. 6, pp. 644-654, Nov. 1976

[Lin 83] Shu Lin and Daniel J. Costello, Jr., *Error Control Coding: Fundamentals and Applications*, Prentice-Hall, 1983.

[MacWilliams 77] F.J. MacWilliams and J.J.A. Sloane, *The Theory of Error-Correcting Codes*, North-Holland, Amsterdam, 1977.

[McEliece 78] R.J. McEliece, "A Public-Key Cryptosystem Based on Algebraic Coding Theory", *DSN Progress Report,* Jet Propulsion Laboratory, CA., Jan. & Feb. 1978, pp. 42-44.

[Meyer 82] Meyer C.H. and Matyas S.M., *Cryptography: A New Dimension in Computer Data Security,* John Wiley & Sons, Inc., 1982.

[Peterson 72] W. Wesley Peterson and E.J. Weldon, Jr., *Error-Correcting Codes,* Second Edition, The MIT Press, 1972.

[Preparata 68] F.P. Preparata, "A Class of Optimum Nonlinear Double-Error-Correcting Codes," Inform. and Control, 13, pp. 378-400, 1968.

[Rao 85] T.R.N. Rao, "Cryptosystems Using Algebraic Codes," *IEEE International Symp. on Info. Theory,*, Brighton, England, June, 1985.

[Rao 87] T.R.N. Rao and K.H. Nam "A Private-Key Algebraic-Coded Cryptosystem", Advances in Cryptology CRYPTO '86, editor A.M. Odlyzko, New York, Springler Verlag, pp. 35-48, 1987.

[Shannon 48] C.E. Shannon. "A Mathematical Theory of Communication," *Bell Syst. Tech. J.*, 27, pp. 379-423 (Part I), 623-656 (Part II), July 1948.

[Struik 87] R. Struik and van Tilburg J., "The Rao-Nam Scheme is Insecure Against a Chosen-plaintext Attack," Advances in Cryptology CRYPTO '87, pp. 445-457, 1987.

[Wood 81] Charles C. Wood, "Future Application of Cryptography," *Proc. of the 1981 Symposium on Security and Privacy,* pp. 70-74, Apr. 1981.

[Vasil'yev 62] Vasil'yev, Jr. L. "Nongroup Close-Packed Codes", *Probl. Cybernet.* (USSR) 8, pp. 337-339, 1962.