Journal of Korea Multimedia Society Vol. 16, No. 5, May 2013(pp. 577-590)
http://dx.doi.org/10.9717/kmms.2013.16.5.577

유한 체상에서의 사전 비밀이미지 공유 기법

현승일[†], 신상호[††], 유기영[†††]

요 약

Shamir의 $(k,n)-threshold$ 비밀 공유(secret sharing) 기법은 참가자(participant)의 서명(signature)과정을 생략하기 때문에 악의적인 공격자에 의해 속임(cheating) 행위가 발생할 수 있고, 이러한 문제점을 해결하기 위해 여러 가지 기법들이 제안되었다. 대표적인 기법으로는 사전 비밀 공유(proactive secret sharing)가 존재한다. 이 기법은 불규칙적인 주기로 참가자들에게 배포된 공유값(shadow value)을 새롭게 변경해준다. 본 논문에서는 사전 비밀 공유 기법을 기존의 비밀이미지 공유(secret image sharing)에 처음으로 적용시킨다. 제안하는 기법은 유한 체($GF(2^8)$)상에서 비밀이미지의 공유가 수행된다. 유한 체 연산은 효율적이고, 안전한 암호 연산을 수행하기 위해 지난 30년간 널리 사용되어 왔고, 제안하는 기법에서는 사전 비밀이미지 공유 과정 내의 비밀 이미지의 손실(lossy)을 방지하기 위해 사용한다. 공유된 이미지(shadow image)를 생성하는 과정 내에서, 비밀이미지 공유 다항식(polynomial)을 이용하여 생성된 값은 삽입 용량(embedding capacity)과 $PSNR$의 상관관계(correlation)를 고려하여 LSB-2 방법을 이용해 커버 이미지(cover image)에 삽입된다. 실험에서는 비밀이미지의 삽입 용량과 공유된 이미지와 커버 이미지(cover image)간의 왜곡(distortion)의 비율(ratio)을 측정한다. 실험 결과에서는 기존의 제안되었던 기법들과의 비교·분석을 통해 제안하는 기법의 우수성을 검증한다.

A Proactive Secret Image Sharing Scheme over $GF(2^8)$

Suhng-Ill Hyun[†], Sang-Ho Shin[††], Kee-Young Yoo[†††]

ABSTRACT

Shamir's (k,n)-threshold secret sharing scheme is not secure against cheating by attacker because the signature of participants is omitted. To prevent cheating, many schemes have been proposed, and a proactive secret sharing is one of those. The proactive secret sharing is a method to update shares in the secret sharing scheme at irregular intervals. In this paper, a proactive image secret sharing scheme over $GF(2^8)$ is proposed for the first time. For the past 30 years, Galois field operation is widely used in order to perform the efficient and secure bit operation in cryptography, and the proposed scheme with update phase of shadow image over $GF(2^8)$ at irregular intervals provides the lossless and non-compromising of secret image. To evaluate security and efficiency of images (i.e. cover and shadow images) distortion between the proposed scheme and the previous schemes, embedding capacity and $PSNR$ are compared in experiments. The experimental results show that the performances of the embedding capacity and image distortion ratio of the proposed scheme are superior to the previous schemes.

Key words: Secret Image Sharing(비밀이미지 공유), Proactive Secret Sharing(사전 비밀 공유), Information Hiding(정보은닉), Galois Field(유한 체)

※ 교신저자(Corresponding Author) : 유기영, 주소 : 대구광역시 북구 산격 3동 1370 경북대학교 IT대학 컴퓨터학부 (702-701), 전화 : 053) 950-5553, FAX : 053) 957-4846, E-mail : yook@knu.ac.kr
접수일 : 2013년 1월 8일, 수정일 : 2013년 2월 18일
완료일 : 2013년 2월 23일

[†] 정회원, 영진사이버대학 정보통신공학계열
(E-mail : mipal@yjc.ac.kr)
[††] 정회원, 경북대학교 IT대학 컴퓨터학부
(E-mail : shshin80@infosec.knu.ac.kr)
[†††] 경북대학교 IT대학 컴퓨터학부
※ 본 논문은 2010학년도 경북대학교 연구년 교수 연구비에 의하여 연구되었음.

1. 서 론

일반적으로 널리 알려진 암/복호(encryption/decryption) 알고리듬의 경우 송·수신자 간의 비밀키(secret key)의 개수는 1개 혹은 2개인 것이 일반적이다. 예를 들어, 대칭키(symmetric) 암호 알고리듬으로 널리 알려진 DES(Data Encryption Standard)와 AES(Advanced Encryption Standard)는 비밀 통신을 위해 송·수신자 간의 비밀키가 1개 존재한다. 반면에 비대칭키(asymmetric) 암호 알고리듬인 RSA와 ECC(Elliptic Curve Cryptography)의 경우 비밀키의 개수는 2개(즉, 공개키(public key)와 비밀키(secret key))이다. 이와 같은 암호 알고리듬들은 개방된 네트워크 공간(open network channel)에서 비교적 안전한 비밀 통신을 제공해주지만 키(key)의 소유주가 갑자기 사망하거나 부재중인 상황이 발생할 경우 암호화된 비밀정보를 알 수 없는 경우가 발생하게 된다. 이러한 상황을 사전에 방지하기 위해 제안된 방법이 비밀 공유이다[1,2].

비밀 공유(secret sharing)는 비밀키의 소유자가 1명이 아닌 다수의 인증된(authorized) 참가자(participant)들이 되고, 이 때 임의의 참가자는 비밀키의 일부 조각을 소유하게 되는 것이다. 예를 들어, 임의의 그룹 내에 n명의 인증된 참가자가 존재할 경우 비밀키를 n개의 조각으로 분리하여 참가자들에게 각각 비밀 조각 1개씩을 나누어주게 되고, 이 중 적어도 k명 이상의 인증된 참가자들 모이면 이들의 비밀 조각들을 이용해 본래의 비밀키(또는 비밀데이터)를 알게 된다. 임의의 참가자는 자신이 소유한 비밀 조각을 통해 원래의 비밀키를 유추할 수 없고, k명 보다 적은 인원이 참가할 경우 원래의 비밀키를 복원할 수 없기 때문에 다수 인원 중 어느 한 참가자가 이탈되어도 큰 문제없이 비밀키를 복원할 수 있다. 이러한 비밀 공유는 주로 회사 내의 중요 시스템에 접근하기 위해 고위직의 임원 다수가 참가하여 접근하는 방식이나 핵미사일과 같은 중요 군사시설의 접근 허가 권한 등에 주로 사용되고 있다.

비밀 공유 기법은 1979년 Adi Shamir[3]와 George Blakley[4]에 의해 각각 처음으로 제안되었다. 두 연구자가 제안한 기법의 세부적인 방법은 상이하지만 비밀 공유라는 큰 맥락에서는 유사하기 때문에 최초의 비밀 공유 기법의 제안에 대해선 두 연구자를 언급한다. 이 후 네트워크의 발전과 함께 개방된(opened) 인터넷에서의 보안에 대한 중요성으로 인해 여러 가지를 고려한 비밀 공유 기법이 제안되었고[5-10], 비밀 공유를 이미지(image)에 적용한 기법들도 많이 연구되었다. 기존의 비밀 공유기법과 이미지 기반의 비밀 공유기법(일반적으로 '비밀이미지 공유기법(secret image sharing)'이라고 불려진다.)의 차이점은 스테가노그래피(steganography)의 적용에 대한 차이이다. 스테가노그래피는 커버 이미지(cover image)내에 비밀데이터를 삽입하여 생성된 스테고 이미지(stego image)를 인터넷 상에 배포하고, 사람들로 하여금 스테고 이미지 내에 비밀데이터가 삽입되어 있는 사실을 모르도록 하는 것으로, 커버와 스테고 이미지 간의 차이 정도를 나타내는 왜곡 비율(distortion ratio)이 작을수록 사람들이 인지할 수 없게 된다. 즉, 기존의 비밀 공유기법은 비밀키 (또는 비밀데이터)를 참가자들에게 나눠줄 공유값(shadow value)을 생성하는 다항식 $f(x)$의 상수항으로 두어 공유값을 생성한 후 참가자들에게 이 값을 나눠준다. 이에 반해 비밀이미지 공유 기법은 비밀이미지 (또는 비밀데이터)내의 하나의 픽셀을 다항식 $f(x)$의 상수항으로 두어 공유값을 생성하는 과정은 동일하나 이 값을 다시 커버 이미지(cover image)내에 숨김으로서 공유된 이미지(shadow image)가 생성되고, 이를 참가자들에게 나눠줌으로서 참가자들은 공유된 이미지 내에 공유값이 숨겨졌는지에 대한 의문점을 원천적으로 봉쇄할 수 있는 장점이 존재한다. 기존의 비밀 공유기법에서는 참가자가 배포 받은 공유값을 분실할 경우 악의적인 공격자에 의해 여러 가지 공격들을 수행할 수 있었다. 그러나 비밀이미지 공유기법의 경우 공유값이 삽입된 친숙한 이미지를 배포함으로서 이를 분실하더라도 악의적인 공격자가 실제로 공유값이 삽입된 이미지인지를 확인할 수 없게 된다. 이러한 기법은 개방된 인터넷상에서 더욱 안전한 공유 기법으로 사용될 수 있다.

1994년과 1996년 Naor & Shamir[11,12]가 제안한 기법은 비밀 이진이미지(binary image)를 패턴을 이용해 기존의 비밀 공유 기법과 같은 방식으로 n명의 참가자들에게 나누어 주고, 이 중 적어도 k명 이상이 모일 경우 이들이 소유하고 있는 이미지 조각들을 겹치(stacking)면 원래의 비밀 이진이미지가 나타나는 것이다. 이 후 2002년에는 Thein & Lin[13]에 의

해 처음으로 그레이스케일(greyscale) 이미지에 대한 비밀이미지 공유 기법이 제안되었다. 그러나 비밀이미지에 대한 손실(loss)이 발생하는 단점이 존재하였고, 이를 보안하기 위해 여러 기법들이 제안되었다[14-17]. 한편 2010년에는 Lin & Chan[18]에 의해 가역이 가능한 비밀이미지 공유 기법이 제안되었고, $threshold$값인 k에 따라 비밀이미지의 삽입 용량(embedding capacity)이 비례적으로 증가하는 장점이 존재했다. 그러나 k명 이하가 모이더라도 비밀이미지를 복원할 수 있거나 삽입 용량이 증가할수록 공유된 이미지(shadow image)의 왜곡 정도($PSNR$)가 심해진다는 단점도 존재했다. 또한 기존에 제안된 비밀이미지 공유 기법들은 딜러(dealer: D)에 의해 공유된 이미지가 생성된 후 참가자들에게 분배한 이후로는 다시 새로운 공유된 이미지가 분배되지 않아 인증된 참가자 중 한 명이 자신이 소유한 공유된 이미지를 분실할 경우 이를 습득한 악의적인 공격자가 비밀이미지의 복원과정에 참가하여 비밀이미지의 정보를 획득할 수 있거나 복원된 비밀이미지의 정보가 올바른지에 대한 정당성을 판단할 수 없는 문제점이 발생했다[19]. 본 논문에서는 이러한 문제점을 해결하기 위해 처음으로 그레이스케일 이미지에 대한 사전 비밀이미지 공유 기법을 제안한다.

본 논문에서 제안하는 사전 비밀이미지 공유(proactive secret image sharing) 기법은 유한 체(Galois Field, $GF(2^8)$)상에서 Shamir의 $(k,n)-threshold$을 기반으로 수행된다. 기존에 제안되었던 비밀이미지 공유 기법들[14-17]과 유사하게 비밀이미지를 공유하고, 공유된 이미지들로부터 비밀이미지를 확인한다. 한편, 비밀이미지로부터 생성된 n명의 참가자들에 대한 공유된 이미지(shadow image)들은 최초에 딜러에 의해 생성된 후 필요에 의해 비규칙적으로 의사난수(pseudorandom number)를 발생하여 n명의 참가자들에게 분배하고 이를 이용해 참가자들은 새로운 공유된 이미지를 생성한다. 공유된 이미지가 새롭게 변경되더라도 이것들로부터 복구되는 비밀이미지의 정보는 항상 동일하기 때문에 앞에서 언급한 악의적인 공격자로부터의 공격은 방지할 수 있다. 실험에서는 기존의 기법들[14,16,17]과 본 논문에서 제안하는 기법 간에 공유된 이미지 내에 숨겨진 비밀이미지의 삽입 용량(embedding capacity)과 커버 이미지(cover image)와 공유된 이미지 간의 왜곡 정도를 측정하는 $PSNR$에 대한 결과를 분석한다. 이를 통해 본 논문에서 제안하는 기법의 삽입 용량과 $PSNR$의 우수성을 확인한다.

본 논문은 다음과 같이 구성되어 있다. 2장에서는 유한 체와 Shamir의 $(k,n)-threshold$기법, 사전 비밀공유 기법에 대해 소개한다. 3장에서는 제안하는 기법에 대한 방법을 자세히 설명하고, 이에 대한 실험 및 결과 분석을 4장에서 다루며, 5장에서 결론을 맺는다.

2. 관련 연구

제안하는 기법은 $GF(2^8)$상에서 Shamir의 $(k,n)-threshold$ 방법을 이용하여 공유이미지를 생성하고, 참가들에게 분배한다. 이를 위해 유한 체, Shamir의 $(k,n)-threshold$와 사전 비밀 공유 기법에 대해 소개한다.

2.1 유한 체(Galois Field)

일반적으로 컴퓨터 내에서 사용되는 비트의 연산을 수학적 모델로 표현하기 위해 유한 체를 사용한다. 비트간의 덧셈(addition) 연산은 비교적 용이하지만 비트간의 곱셈(multiplication)이나 나머지(modulus) 연산은 유한 체 연산을 이용하는 것이 효율적이다. 이러한 유한 체(Galois field)는 암호학의 발전과 함께 여러 분야에서 널리 사용되어왔다. 암호 알고리듬이 개발되기 시작한 1970년부터 큰 수의 지수 연산을 효율적으로 처리하기 위한 알고리듬의 연구에 유한 체를 사용하였고, 1980년대 후반에 발표된 타원곡선암호(Elliptic Curve Cryptography: ECC) 알고리듬과 2000년대 초반에 표준으로 제정된 AES (advanced encryption standard) 내에서의 모든 연산들은 유한 체 $GF(p^m)$과 $GF(2^8)$을 각각 사용하여 효율적이고, 빠른 연산을 수행할 수 있었다.

유한 체를 정의하기 위해서는 먼저 체를 정의해야 한다. 임의의 집합 $\mathbf{F}$에 대해, $\mathbf{F}$ 내의 원소들이 덧셈(addtion)과 곱셈(multiplication)에 대한 이항연산(binary operation)의 결과가 $\mathbf{F}$ 내의 원소로 존재해야 하고, 동시에 모든 원소들에 대해 덧셈과 곱셈에 대한 항등원(identity)과 역원(inverse)에 해당 하는 결과가 $\mathbf{F}$ 내의 원소로 존재하는 경우를 일반적으로 체(field)로 정의하고, $\{\mathbf{F},+,\times\}$으로 표기한다. 만약

체 내의 모든 원소에 대한 개수가 유한(finite)일 경우 유한 체(finite field) 또는 Galois 체로 정의하고, $\mathbf{F}_{p^n}$ 또는 $GF(p^n)$으로 표기한다. 단, p와 n은 각각 소수와 비트의 수를 의미하고, $GF(p^n)$상에서 표현할 수 있는 정수의 범위는 $[0, p^n-1]$이다. 본 논문에서는 유한 체의 표기 형식을 $GF(p^n)$으로 한다.

유한 체 $GF(2^n)$상에서의 임의의 한 원소 a는 다항식으로 표현이 가능하고, 다음의 식(1)과 같다.

$$a(x)=\sum_{i=0}^{n-1} a_i x^i = a_{n-1}x^{n-1}+a_{n-2}x^{n-2}+\cdots+a_1 x+a_0 \quad (1)$$

단, a_i는 0 또는 1의 값을 의미하고, 원소 a는 $a=(a_{n-1}a_{n-2}\cdots a_1 a_0)$로 표현이 가능하다. $GF(2^n)$상에서 다항식의 총 개수는 2^n이고, 표현 가능한 범위는 $[0, 2^n-1]$이다. 예를 들어, $GF(2^3)$상에서 표현할 수 있는 다항식의 총 개수는 8이고, 다음의 식 (2)와 같다[2].

$$\begin{array}{cccc} 0 & x & x^2 & x^2+x \\ 1 & x+1 & x^2+1 & x^2+x+1 \end{array} \quad (2)$$

본 논문에서는 그레이스케일 이미지(gray scale image)를 이용하여 사전 비밀이미지 공유 기법을 제안하고, 이미지 내의 한 픽셀은 8비트로 구성되어 있기 때문에 $GF(2^8)$상에서의 유한 체 연산을 사용한다.

2.2 Shamir의 $(k,n)-threshold$ 기법

Shamir가 제안한 비밀공유기법[3]을 이해하기 위해선 먼저 $(k,n)-threshold$ 기법에 대해 살펴보고자 한다. 다음의 정의 1은 $(k,n)-threshold$에 대한 개념을 설명[1]하고 있다.

정의 1: 양의 정수 k와 n(단, $k \le n$)에 대하여, $(k,n)-threshold$ 기법은 n명의 참가자들의 집합 내에서 비밀키(secret key: K)를 공유하는 방법이다. 공유하는 방법은 임의의 참가자(participants)들 중 적어도 k명 이상의 인원이 존재할 경우 비밀키(K)를 계산할 수 있다. 만약 $k-1$명 또는 그 이하의 인원이 있을 경우는 비밀키(K)를 계산할 수 없다.

1979년 Shamir는 $(k,n)-threshold$ 개념과 Lagrange의 보간법(interpolation)을 이용하여 비밀공유기법[3]을 제안하였고, 이와 같은 해에 Blakley는 선형 사영기하학(linear projective geometry)기법을 이용하여 비밀공유기법[4]을 제안하였다. 두 연구자가 제안한 비밀공유기법들은 유사하고 비밀을 복원하는 과정에서 사용되는 기법이 다르지만 같은 해에 제안이 되었기 때문에 Shamir와 Blakley의 비밀 공유 기법으로 통용된다. 이후 Shamir가 제안한 기법이 암호학의 많은 분야에 응용되어 일반적인 비밀공유기법은 Shamir가 제안한 기법을 지칭한다.

Shamir가 제안한 기법은 크게 세 부분으로 나누어져 있고, 이는 초기화 과정, 공유값 분배과정, 그리고 비밀키 복원과정으로 이루어져 있다. 이 모든 과정에서 하나의 비밀키로부터 공유값을 생성, 분배, 그리고 복원하는 역할은 딜러(dealer)가 수행한다. 딜러는 합법적으로 인증된 자로 가정하고, 각각의 과정은 단계별로 자세히 설명한다[1,3].

2.2.1 초기화 과정

딜러는 Z_p상의 0이 아닌 서로 다른 원소 n개를 선택(항상 다음의 조건 $p \ge n+1$을 만족해야한다.)하고, 이를 x_i로 표기한다. 단, i는 참가자들의 순서(index)를 의미하고, $1 \le i \le n$을 만족한다. 임의의 i에 대해서, 딜러는 x_i를 i번째 참가자 P_i에 대응시켜 준다.

2.2.2 공유값 분배과정

***step* 1 :** 딜러가 공유하려는 비밀키(K)는 Z_p상의 임의의 원소(즉, $K \in Z_p$)로 가정한다. 딜러는 Z_p상에서 $k-1$개의 원소들을 비밀스럽게 선택(이때, 선택된 원소들은 모두 독립적으로 랜덤(random)하다.)하고, 이를 $a_1, a_2, \ldots, a_{k-1}$로 각각 표기한다.

***step* 2 :** 임의의 i $(1 \le i \le n)$에 대해, 딜러는 다음의 식 (3)을 이용해 공유값 $y_i = a(x_i)$값을 계산한다.

$$a(x)=K+\sum_{j=1}^{k-1} a_j x^j \pmod p \quad (3)$$

***step* 3 :** 임의의 i $(1 \le i \le n)$에 대해, 딜러는 i번째 참가자 P_i에 대응하는 공유값(y_i)을 분배한다.

2.2.3 비밀 복원과정

***step* 1 :** 딜러는 n명의 참가자 중 k명 또는 그 이상의 인원을 모집한 후 그 인원으로부터임의의 참가자 P_i가 소유한 공유 값 y_i를 수집한다.

step 2 : 수집한 k 또는 그 이상(단, $k \le n$)의 (i, y_i) 쌍들과 Lagrange 보간법(interpolation)을 이용하여 공유값 분배과정에서 사용했던 식 (3)을 복원한다. Lagrange 보간법에서 사용되는 식 (4)은 다음과 같다.

$$a(x) = \sum_{j=1}^{k} \left(y_j \prod_{1 \le o \le k, o \ne j} \frac{x - x_o}{x_j - x_o} \right) \bmod p \qquad (4)$$

단, x_j와 x_o는 순번이 j번과 o번째인 참가자의 고유값을 각각 의미하고, y_j는 $a(x_j)$에 대응되는 값이며, p는 소수이다.

step 3 : 딜러에 의해 복원된 식 (4)를 통해 비밀키(K)를 계산하고, 비밀키의 복원이 완료된다.

2.3 사전 비밀 공유 기법

1995년 Herzberg et al.[20]이 제안한 사전 비밀 공유(proactive secret sharing)기법은 참가자(participant)의 공유값(shared value)이 악의적인 공격자(attacker)에게 절도(steal) 또는 복사(copy)당하는 경우를 방지하기 위해 고안되었다. 즉, 일반적인 비밀 공유 기법은 딜러에 의해 한번 공유된 값이 지속적으로 사용되기 때문에 공격자가 합법적인 참가자의 공유값을 알게 될 경우 합법적인 참가자로 위장(forgery)하여 비밀을 복원하는 과정에 참여해 비밀의 내용을 알게 되는 상황까지 발생할 수 있다. 이러한 취약점을 보완하기 위해 제안된 기법이 사전 비밀 공유이다. 사전 비밀 공유 기법은 참가자들에게 분배한 공유값의 주기적인 변경(update) 및 재분배를 통해 위장과 같은 공격으로부터 안전하게 비밀 공유를 수행한다. Herzberg et al.이 제안한 사전비밀공유기법의 과정은 다음과 같다. 최초 비밀키(K)로부터 n개의 공유값을 생성, 분배, 복원하는 과정은 Shamir가 제안한 $(k,n)-threshold$ 기법[3]과 동일하므로 과정의 설명은 생략하고, 공유값을 새로운 값으로 업데이트(update)하는 과정에 대해 설명한다.

2.3.1 공유값의 업데이트 과정

step 1 : 딜러는 모든 참가자 n명에 대한 랜덤(random)한 식 (5)와 같은 다항식을 n개를 각각 생성한다.

$$\delta_i^t(z) = \delta_{i,1}^t z^1 + \delta_{i,2}^t z^2 + \cdots + \delta_{i,k-1}^t z^{k-1} \ (\bmod p) \qquad (5)$$

단, $i(i=1,2,...,n)$는 i번째 참가자를 의미하고, t는 공유값의 t번째 업데이트를 의미한다.

step 2 : 딜러는 생성된 n개의 다항식(식 (5))을 통해 $u_{i,j}^t = \delta_i^t(j)$ 값을 계산한다. 단, j는$[0, k-1]$ 범위 내에서 순차적으로 계산한다.

step 3 : 딜러는 생성된 $u_{i,j}^t = \delta_i^t(j)$ 값들을 참가자들에게 브로드캐스트(broadcast)통해 전송하고, 참가자 i는 딜러가 보낸 값들 중 $u_{i,i}^t$ 혹은 인덱스 j가 $j=i$인 경우에 해당하는 $u_{i,j}^t$ 값을 선택하여 받는다.

step 4 : 임의의 참가자 i는 기존의 공유값 x_i^t와 딜러로부터 전송받은 $u_{i,i}^t$와 $u_{m,i}^t$(단, m은$[0,i),(i,k-1]$)들을 이용하여 식 (6)과 같이 공유값을 업데이트한다.

$$x_i^{t+1} = x_i^t + \sum_{m=1}^{k-1} u_{m,i}^t \ (\bmod p) \qquad (6)$$

3. 유한 체($CF(2^8)$)상에서의 사전 비밀이미지 공유 기법

본 절에서는 제안하는 유한 체상에서의 사전 비밀이미지 공유 기법에 대해 설명하기 위해 제안하는 기법에서 사용되는 용어에 대해 정의하고, 각 과정에서 사용되는 알고리듬에 대해 설명한다.

3.1 용어 정의와 주요 개념

제안하는 기법의 알고리듬 내 사용되는 용어는 다음의 표 1과 같다.

표 1 내에서 커버이미지, 비밀이미지, 임의의 참가자 $P_{(i)}$의 공유된 이미지를 각각 CI, SI, $SHI_{P_{(i)}}$로 표기한다. M은 비밀이미지 SI의 가로(width) (또는 세로(height)) 크기를 각각 지칭하고, 각 이미지 내에 존재하는 C_l, S_l, SH_l은 8-비트 단위의 픽셀 값으로 구성되어 있으며, 다항식 $f^{(t)}(x)$와 $\delta_{(i)}^{(t)}(x)$의 최고차항의 차수(degree)는 비밀 공유의 기준 값(threshold)인 k에 의해 결정된다.

제안하는 사전 비밀이미지 공유기법은 Shamir가 제안한 $(k,n)-threshold$ 기법의 기본 원리를 따른다. 그러나 Shamir가 제안한 기법은 식 (3)의 나머지 연산에 사용되는 소수인 p의 값이 255를 초과하는 경우에 대해 제안하는 기법의 그레이스케일 이미지의 적용에 대해 여러 가지 제약사항이 발생한다. 그러나 Shamir가 제안한 기법은 식 (3)의 나머지 연산에 사용되는 소수인 p의 값이 255를 초과하는 경우 제안하는 기법의 그레이스케일 이미지의 적용 시 여러 가지

표 1. 용어의 정의

용 어	설 명
$CI=\{C_0, C_1, ..., C_{4M^2-1}\}$	크기가 $2M\times 2M$인 커버 이미지(cover image:CI)와 커버 이미지 내의 픽셀 값들의 집합
$SI=\{S_0, S_1, ..., S_{M^2-1}\}$	크기가 $M\times M$인 비밀 이미지(secret image:SI)와 비밀 이미지 내의 픽셀 값들의 집합
D, $P_{(i)}$	D: 딜러(dealer), $P_{(i)}$: i번째 참가자(participant)
$SHI_{P_{(i)}}=\{SH_0, SH_1, ..., SH_{4M^2-1}\}$	크기가 $2M\times 2M$인 $P_{(i)}$에 대한 공유된 이미지(shadow image:$SHI_{P_{(i)}}$)와 공유된 이미지 내의 픽셀 값들의 집합
CB_l, SHB_l	연속되는 4개의 픽셀로 이루어진 l번째 커버와 공유된 이미지 내의 블록들
$f^{(t)}(x)$	비밀 공유를 위한 t번째 $k-1$차 다항식(polynomial)
$\delta_{(i)}^{(t)}(x)$	참가자의 i번째 $k-1$차 다항식(polynomial)

제약사항이 발생한다. 기존의 제안된 비밀이미지 공유 기법들은 일반적으로 다항식 $f(x)$의 나머지 연산을 위해 소수 p를 251로 지정하였다. 이 경우 그레이스케일 이미지 픽셀 값의 범위인 0부터 255까지의 값들 중 252, 253, 254, 255의 픽셀 값들은 나머지 연산에 의해 각각 1, 2, 3, 4로 변경되어 기존의 픽셀들 값과 중복되는 결과가 발생하고 이러한 픽셀 값을 가진 비밀이미지를 사용할 경우 픽셀 값들이 손실(lossy)되는 경우가 발생하여 부득이하게 비밀이미지의 픽셀 값을 나누는 방법 등을 통해 비밀이미지 공유를 수행할 수 있었다. 그러나 $GF(2^8)$상에서의 나머지 연산은 이러한 픽셀 값들의 손실을 원천적으로 막을 수 있도록 소수 p 대신 기약다항식(irreducible polynomial)을 사용한다. $GF(2^8)$상에서의 연산은 정수가 아닌 비트별 연산이 가능하기 때문에 소프트웨적 또는 하드웨어적으로도 구현이 용이하고, 임의의 소수를 정하는 불편함이 없기 때문에 그레이스케일 더 나아가 컬러 이미지에서도 유용하게 사용할 수 있다. 그러므로 제안하는 기법 내에서 공유된 이미지를 생성하기 위해 사용하는 다항식은 $(k,n)-threshold$일 경우 다음의 식 (7)과 같이 표현할 수 있다.

$$f^{(t)}(x)=\{S_l+a_1x+a_2x^2+\cdots+a_{k-1}x^{k-1}\} \bmod p(x) \quad (7)$$

단, S_l(단, $0\le l\le M^2-1$)은 임의의 8-bit 비밀데이터 값이고, $p(x)$는 $GF(2^8)$상의 기약 다항식(irreducible polynomial)이며, 다항식의 계수들 $(a_1, a_2, ..., a_{k-1})$은 $GF(2^8)$상에서 임의로 선택한 난수(random number)이다.

한편, 공유된 이미지를 분배한 후 다시 이를 업데이트 하는 과정에서 사용하는 식 (5)의 경우 본 논문에서는 $GF(2^8)$상에서 연산을 수행하므로 다음의 식 (8)과 같이 표현할 수 있다.

$$\delta_i^t(z)=\{\delta_{i,1}^t z^1+\delta_{i,2}^t z^2+\cdots+\delta_{i,k-1}^t z^{k-1}\} \bmod p(x) \quad (8)$$

단, 계수들$(\delta_{i,1}^t, \delta_{i,2}^t, ..., \delta_{i,k-1}^t)$은 유한 체($GF(2^8)$)상에서 임의로 선택한 난수이다.

3.2 사전 비밀이미지 공유 과정

사전 비밀이미지 공유 과정에서는 비밀데이터를 다항식에 삽입한 후 참가자 별로 다항식의 결과 값을 생성하여 이를 비트 단위로 나누어 커버 이미지 내의 픽셀에 순차적으로 숨기는 과정을 통해 n명의 참가자에 대한 공유된 이미지(shadow image)를 생성한다. 본 절에서는 제안하는 기법의 사전 비밀 이미지 공유 과정을 단계별로 자세히 설명한다. 제안하는 기법의 모든 과정은 Shamir의 $(k,n)-threshold$ 원리[3]를 기본으로 한다.

입력(input) : CI, SI, k, n, $t=0$
출력(output) : n개의 $SHI_{P_{(i)}}$ $(1\le i\le n)$

step 1 : 크기가 $2M\times 2M$인 커버 이미지 CI를 2×2 크기의 블록(block)으로 나누어 각각의 블록에 대해 CB_0, CB_1, ..., CB_{M^2-1}과 같이 표기를 한다. 임의의 블록 CB_l(단, $0\le l\le M^2-1$)는 4개의 픽셀로 구성되어 있고, 이는 각각 C_m, C_{m+1}, C_{m+2}, C_{m+3} (단, $m=4l$)이다.

step 2 : l번째 블록 CB_l에 대한 임의의 최고차항

의 차수(degree)가 $k-1$인 다항식을 생성하기 위해 딜러 D는 $GF(2^8)$상에서 $k-1$개의 임의의 계수들을 선택(단, $a_{k-1} \neq 0$)한 후 공유하고자 하는 비밀 이미지의 S_l과 함께 식 (7)을 생성한다. 최초의 비밀 공유 단계에서는 타임스탬프(time-stamp) 값 t는 0이 되고, 공유된 이미지를 새롭게 업데이트 하는 경우에는 매번 1씩 증가하게 된다. 제안하는 기법에서 사용하는 $GF(2^8)$상에서의 기약 다항식 $p(x)$는 다음의 식 (9)와 같다.

$$p(x) = x^8 + x^5 + x^4 + x^3 + x + 1 \text{ over } GF(2^8) \quad (9)$$

***step* 3 :** 다항식 $f^{(t)}(x)$에 대해 각 참가자들에 대한 고유 값($x_{P_{(i)}}$)을 이용하여 다음의 값을 계산한다.

$$y_{P_{(i)}} = f^{(t)}(x_{P_{(i)}}) \quad (10)$$

단, i는 각 참가자들의 순번(index)를 가리키고, 범위는 $1 \le i \le n$와 같다.

***step* 4 :** $y_{P_{(i)}} (= b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0)$의 값을 2-bit 씩 나눈 후 이를 CB_l 내의 4개 픽셀들의 최하위비트(least significant bit: LSB) 2-비트에 각각 삽입한다. 삽입 방법은 다음의 그림 1과 같다.

그림 1은 커버 이미지내의 임의의 CB_l에 존재하는 4개의 픽셀($C_m, C_{m+1}, C_{m+2}, C_{m+3}$)에 대해 $y_{P_{(i)}} (= b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0)$를 2-비트씩 나누어 삽입하고, 이 결과 공유된 이미지 내에 존재하는 임의의 SHB_l이 생성된다. 마지막으로 l의 값을 확인하고, $l \le M^2 - 1$인 경우는 l값을 1증가시킨 후 ***step* 2**의 처음단계로 되돌아가고, $l > M^2 - 1$인 경우는 커버 이미지 내의 모든 픽셀에 대한 삽입을 마친 것이므로, 공유된 이미지 $SHI_{P_{(i)}}$ n개를 전체 참가자에게 딜러가 나누어 줄 준비를 수행한다.

3.3 비밀이미지의 복원 과정

공유된 이미지 n개로부터 k개를 선택하여 본래의 비밀이미지를 복원하는 과정은 사전 비밀이미지 공유 과정의 역순으로 진행한다. 역순으로 진행하는 과정은 딜러에 의해 수행되고, 이 과정에서 참가자의 고유 값인 $x_{P_{(i)}}$와 고유 값에 대한 다항식의 결과 값인 $y_{P_{(i)}}$의 쌍을 Lagrange의 보간법(interpolation)을 이용하여 본래의 다항식 $f^{(t)}(x)$을 찾고, 다항식 중 상수항의 값을 이용해 본래의 비밀이미지 SI를 복원한다. 본 절에서는 제안하는 기법의 비밀 이미지의 복원 과정을 단계별로 자세히 설명한다.

***입력(input)* :** k개의 $SHI_{P_{(i)}}$

***출력(output)* :** SI

***step* 1 :** 크기가 $2M \times 2M$인 k개의 공유된 이미지 $SHI_{P_{(i)}}$들로부터 참가자의 순번 $P_{(i)}$이가 빠른 순서대로 정렬한 다음 첫 번째 $SHI_{P_{(i)}}$의 첫 번째 SHB_l 블록으로부터 4개의 픽셀들(SH_m, SH_{m+1}, SH_{m+2}, SH_{m+3}) 내에 존재하는 LSB 2-비트를 각각 추출한 후 식 (11)과 같이 연결한다.

$$LSB2(SH_m) \| LSB2(SH_{m+1}) \| LSB2(SH_{m+2}) \| LSB2(SH_{m+3}) = y_{P_{(i)}} \quad (11)$$

***step* 2 :** 단계 1로부터 추출한 $y_{P_{(i)}}$와 참가자의 고유 값 $x_{P_{(i)}}$을 쌍으로 하여 비밀이미지의 픽셀 S_l에 대한 다항식 $f^{(t)}(x)$을 Lagrange의 보간법을 이용하여 재구성한다. 본 논문에서 사용하는 Lagrange의 보간법은 식 (12)와 같다.

$$f^{(t)}(x) = \sum_{i=1}^{k} \left(y_{P_{(i)}} \prod_{1 \le o \le k, o \neq i} \frac{x - x_{P_{(o)}}}{x_{P_{(i)}} - x_{P_{(o)}}} \right) \bmod p(x) \quad (12)$$

단, $p(x)$는 식 (9)와 동일하고, $x_{P_{(i)}}$와 $x_{P_{(o)}}$는 i번째

$CB_l\ (l = 4m)$

C_m $(w_7 w_6 w_5 w_4 w_3 w_2 w_1 w_0)$	C_{m+1} $(w_7 w_6 w_5 w_4 w_3 w_2 w_1 w_0)$
C_{m+2} $(w_7 w_6 w_5 w_4 w_3 w_2 w_1 w_0)$	C_{m+3} $(w_7 w_6 w_5 w_4 w_3 w_2 w_1 w_0)$

→

$SHB_l\ (l = 4m)$

SH_m $(w_7 w_6 w_5 w_4 w_3 w_2 b_7 b_6)$	SH_{m+1} $(w_7 w_6 w_5 w_4 w_3 w_2 b_5 b_4)$
SH_{m+2} $(w_7 w_6 w_5 w_4 w_3 w_2 b_3 b_2)$	SH_{m+3} $(w_7 w_6 w_5 w_4 w_3 w_2 b_1 b_0)$

그림 1. CB_l내의 4 픽셀에 대한 $y_{P_{(i)}}$ 삽입방법

와 o번째 순번을 가진 참가자의 고유 값을 각각 의미한다.

***step* 3 :** 단계 2에서 재구성된 $f^{(t)}(x)$로부터 비밀이미지의 픽셀 값 S_l을 추출한다. 이 후 l값을 1 증가한 후 확인하여 $l \le M^2-1$인 경우 ***step* 1**의 처음 과정으로 되돌아가고, $l > M^2-1$인 경우 단계를 종료하고, 추출된 S_0, S_1,..., S_{M^2-1}들을 이미지로 재구성하여 본래의 비밀이미지를 딜러가 확인하게 된다. 이를 통해 공유된 이미지로부터 비밀이미지를 복원하는 과정을 마치게 된다.

3.4 공유된 이미지의 업데이트 과정

공유된 이미지의 업데이트 과정은 기존에 공유된 이미지 내에 존재하는 다항식의 결과값을 새롭게 변경해주는 것으로서 숨기고자하는 비밀 이미지의 정보는 그대로 유지하되 공유된 이미지의 왜곡(distortion)이 거의 없도록 하는 것이다. 본 절에서는 제안하는 기법의 공유된 이미지의 업데이트 과정을 단계별로 자세히 설명한다.

***입력(input)* :** n개의 *SHI*, n, t
***출력(output)* :** 업데이트된 n개의 *SHI*, 업데이트된 t

***step* 1 :** 크기가 $2M \times 2M$인 공유된 이미지 *SHI*를 2×2 크기의 블록(block)으로 나누어 각각의 블록에 대해 SHB_0, SHB_1,..., SHB_{M^2-1}과 같이 표기를 한다. 임의의 블록 SHB_l(단, $0 \le l \le M^2-1$)는 4개의 픽셀로 구성되어 있고, 이는 각각 SH_m, SH_{m+1}, SH_{m+2}, SH_{m+3} (단, $m = 4l$) 이다.

***step* 2 :** 임의의 블록 SHB_l에 대한 최고차항의 차수가 $k-1$인 다항식 $\delta_i^t(z)$을 생성하기 위해 i번째 참가자 $P_{(i)}$는 $GF(2^8)$상에서 $k-1$개의 임의의 계수들을 선택(단, $\delta_{i,k-1}^t \neq 0$) 한 후 식 (8)을 생성한다. 다항식 $\delta_i^t(z)$는 최고차항의 차수가 $k-1$인 z에 대한 식으로서 공유 값을 생성하기 위해 사용되는 다항식 $f^{(t)}(x)$과 비슷한 형태이지만 상수항이 없는 것이 특징이다. 다항식 $\delta_i^t(z)$의 상수항이 없는 이유는 업데이트된 n개의 *SHI*들로부터 비밀 이미지를 완벽하게 복원하기 위한 것이다. 제안하는 기법에서 사용하는 유한 체($GF(2^8)$)상에서의 기약 다항식 $p(x)$는 다음의 식 (9)와 같다.

***step* 3 :** n명의 참가자는 생성된 다항식 $\delta_i^t(z)$를 통해 $u_{i,j}^t = \delta_i^t(j)$ 값을 계산한다. 단, j의 범위는 $[1, n]$이다. 즉, 참가자의 수만큼 j를 순차적으로 반복하여 $u_{i,j}^t = \delta_i^t(j)$ 값을 생성한다. 본 논문에서 가정하는 $(k,n)-threshold$인 경우는 전체 참가자 n명에 대한 i번째 참가자 $P_{(i)}$는 $u_{i,1}^t = \delta_i^t(1)$, $u_{i,2}^t = \delta_i^t(2)$, ..., $u_{i,n}^t = \delta_i^t(n)$를 각각 생성한 후 j값이 i와 동일한 경우 ($j=i$)를 제외한 모든 값을 전체 참가자들에게 브로드캐스트(broadcast) 통신을 수행한다. 다른 참가자들은 브로드캐스트 통신된 $u_{i,j}^t$의 j값이 자신의 인덱스(i)와 동일한 경우 받아서 저장하고, 아닌 경우는 $u_{i,j}^t$값을 버린다.

***step* 4 :** i번째 참가자 $P_{(i)}$는 다른 참가자들로부터 받은 $u_{1,i}^t$, ..., $u_{i-1,i}^t$, $u_{i+1,i}^t$, ..., $u_{n,i}^t$ 값과 자신이 생성한 $u_{i,i}^t$ 값 및 이전 단계($t-1$)의 비밀공유 값인 $f^{(t-1)}(x_{P_{(i)}})$를 이용하여 새로운 비밀 공유 값 $f^{(t)}(x_{P_{(i)}})$을 식 (13)을 이용해 생성한다.

$$f^{(t)}(x_{P_{(i)}}) = \left\{ f^{(t-1)}(x_{P_{(i)}}) + u_{1,i}^t + u_{2,i}^t + \cdots + u_{n,i}^t \right\} \bmod p(x) \quad (13)$$

생성된 $f^{(t)}(x_{P_{(i)}})$을 이용하여 임의의 SHB_l에 대해 $f^{(t)}(x_{P_{(i)}})(=b_7b_6b_5b_4b_3b_2b_1b_0)$의 값을 2-비트씩 나눈 후 이를 SHB_l 내의 4개 픽셀들의 최하위비트(least significant bit: LSB) 2-비트에 각각 삽입한다. 삽입 과정은 그림 1과 동일하게 수행된다. 이로서 업데이트 과정은 마치게 되고, 결과 값으로 업데이트된 n개의 *SHI* 들이 생성되고, 딜러 D는 t값을 1 증가한 후 이를 저장한다.

4. 실험 및 평가 분석

본 절에서는 제안한 기법에 대한 성능과 안전성을 분석하기 위해 여러 가지 실험을 수행하고, 수행된 실험 결과를 통해 제안한 기법과 다른 기법들[14,16, 17] 간의 삽입 용량(embedding capacity)과 *PSNR*

(Peak Signal to Noise Ratio)을 비교한다. 이를 통해 본 논문에서 제안한 사전 비밀이미지 공유기법의 우수성을 검증한다.

4.1 실험 도구와 평가 기준

일반적으로 정보 은닉(information hiding) 또는 비밀이미지 공유 기법들에 대한 성능 평가를 위해 크게 두 가지 측정 기준을 사용된다. 첫 번째는 삽입 용량으로서 커버 이미지(cover image) 내에 비밀데이터(secret data)를 숨겨(또는 삽입(embedding)하여) 스테고(또는 공유된) 이미지(stego image or shadow image)를 생성하는 경우 스테고 이미지 내에 얼마나 많은 비밀 데이터가 삽입되었는지를 측정하는 것이다. 측정하는 방법은 보통 스테고 이미지 내의 한 픽셀 내에 비밀데이터가 얼마나 삽입되었는지를 나타내는 bpp(bit per pixel)나 전체 이미지 내에 삽입된 비밀데이터의 용량(단위는 bit)을 사용한다.

또 다른 측정 기준은 커버와 스테고 이미지 간의 이미지 왜곡(distortion)을 측정하는 것이다. 정보 은닉 기법의 목적은 공격자에게 커버 이미지 내에 비밀데이터를 숨겼는지(또는 삽입되었는지)를 알 수 없게 하는 것이므로, 커버와 스테고 이미지 간의 왜곡은 매우 중요하다. 이러한 왜곡을 측정하기 위해서는 *PSNR*을 사용하는데 이것은 다음의 식 (14)와 같이 표현된다.

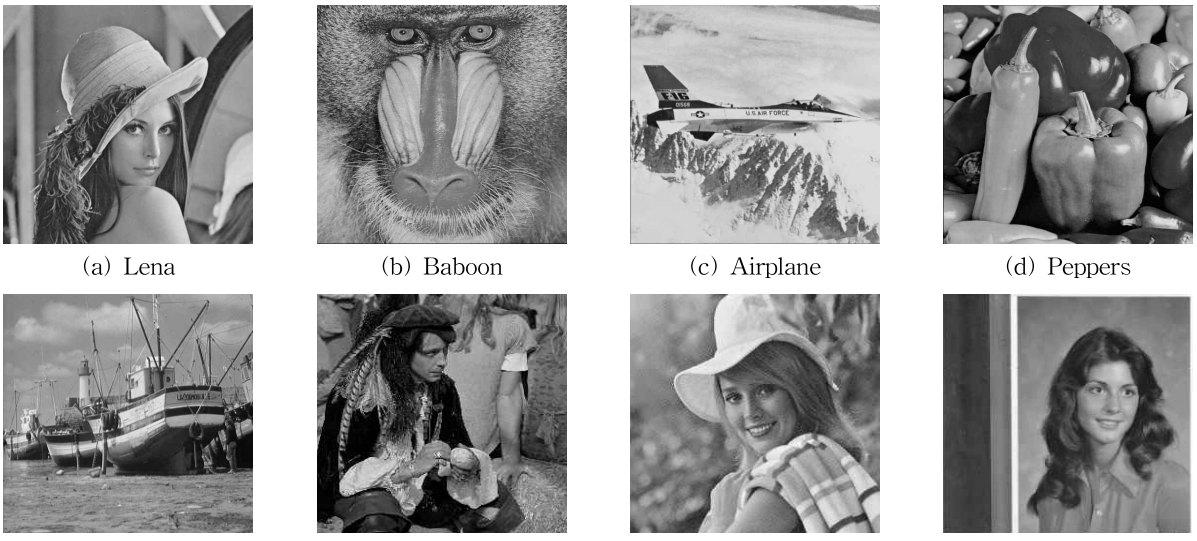
$$PSNR = 10 \times \log\left(\frac{MAX^2}{MSE}\right) \tag{14}$$

단, MSE는 에러평균의 제곱(mean squared error) 값으로 다음의 식 (15)와 같이 표현되고, MAX는 한 픽셀이 표현할 수 있는 최댓값(maximum value)을 의미하는 것으로 본 논문의 실험에서는 그레이스케일 이미지의 픽셀 값 중 최댓값인 255를 사용한다.

$$MSE = \frac{1}{4M^2}\sum_{i=0}^{4M^2-1}[C_i - SH_i]^2 \tag{15}$$

단, C_i와 SH_i는 본 논문에서 사용하는 커버 이미지와 공유된 이미지의 i번째 픽셀 값을 각각 의미한다. *PSNR* 수치는 높으면 높을수록 두 이미지 간의 왜곡이 거의 없는 것(최댓값: ∞)이고, 낮으면 낮을수록 두 이미지 간의 왜곡이 매우 많아져(최솟값: 0), 사람의 시각(human visible system: HVS)으로 인지할 수 있게 된다. 일반적으로 사람이 인지할 수 없는 왜곡의 *PSNR* 수치는 35dB로서 이 값보다 작을 경우 HVS 측면에서 왜곡을 확인할 수 있다. 또한, 공간영역(spatial domain)에서의 삽입 용량과 *PSNR* 수치의 상관관계(correlation)는 삽입 용량이 많을수록 *PSNR*의 수치는 일반적으로 낮아지고, 삽입 용량이 적을수록 *PSNR*의 수치는 일반적으로 높아진다[21-24].

실험에 사용된 커버 이미지는 정보은닉의 실험에서 일반적으로 사용하는 그레이스케일 이미지 8개를



(a) Lena (b) Baboon (c) Airplane (d) Peppers

(e) Boat (f) Man (g) Elaine (h) Woman

그림 2. 실험에서 사용한 8개의 커버영상

사용하였고, 각 이미지의 크기는 512×512로 고정하였다. 본 논문에서 비밀이미지의 크기를 $M\times M$으로 가정하면, 커버 이미지의 크기는 $2M\times2M$이 되므로, 실험에서 사용되는 비밀이미지의 크기는 256×256로 고정된다. 또한, 비밀이미지의 크기가 $M\times M$ 보다 작을 경우 커버 이미지에 대한 크기만큼 비례하게 $[0,255]$ 범위 내의 임의의 값을 딜러가 삽입한 후 비밀공유를 진행한다. 반대로 비밀이미지의 크기가 $M\times M$ 보다 클 경우는 커버 이미지의 크기를 늘려서 비밀공유를 진행한다.

본 실험에서는 $(2,3)-threshold$과 $(4,4)-threshold$ (Wang & Shyu[16] 기법에 해당)인 경우에 대해 실험을 수행했다. 그림 2는 실험에서 사용한 그레이스케일 커버 이미지 8개를 보여준다. 비밀이미지는 난수를 발생하는 C++ 라이브러리를 이용하여 비트열(bitstream)을 생성한 후 이를 8-bit 단위로 나누어 하나의 픽셀로 사용하였다.

4.2 삽입 용량(embedding capacity)의 실험 결과 및 분석

2002년 처음 비밀이미지 공유 기법이 제안된 이래로 2010년 Lin과 Chan[18]이 제안한 기법 이전까지는 삽입용량에 대해 거의 언급되지 않았다. 스테가노그래피와 비밀이미지 공유 기법은 이미지에 비밀을 숨긴다는 방법은 비슷하지만 지향하는 목적이 다르기 때문에 삽입 용량보다는 공유된 이미지의 왜곡(distortion) 정도와 공유된 이미지로부터 안전하게 비밀이미지 (또는 비밀데이터)를 찾는 것이 중요했다. 또한, 지금까지 제안된 대부분의 비밀이미지 공유 기법들은 비밀데이터가 숨겨진 다항식의 결과값을 이용하여 이미지에 삽입되는 기법이었기 때문에 커버이미지에 삽입되는 정확한 비밀이미지 (또는 비밀 데이터)의 삽입 용량을 측정할 수 없었다. 그러나 Lin & Chan이 제안한 기법에서는 공유된 이미지의 왜곡을 없애고, 가역(reversible)을 적용하기 위해 다항식의 비밀데이터 삽입 방법을 달리하여 $threshold$ 값인 k를 이용하여 비교적 정확한 삽입 용량을 측정할 수 있었다. 그러나 공유된 이미지 내에 삽입된 비밀 데이터의 용량을 측정하는 것은 불규칙적이기 때문에 기준을 어디에 두는가에 따라 다른 결과가 나타나는 문제점이 발생했다.

본 실험에서는 공유된 이미지 내에 삽입된 비밀 데이터의 용량을 측정하는 것으로 기준을 정하여 다른 기법들과 비교를 수행하였다. 표 2는 삽입 용량에 대한 결과를 보여주고 있다. 표 2에서 8개의 테스트 이미지들에 대해 각각의 기법들마다 모두 동일한 값이 들어가는 이유는 동일한 공간 영역 기법들을 이용했기 때문이다. Wang & Shyu[16]가 제안한 기법의 경우 일반적인 비밀이미지 공유 기법들과 달리 비밀이미지를 분할하여 공유된 이미지로 생성하고, n명의 참가자 중 n명 모두 참가해야 완벽하게 비밀이미지가 복원된다는 특징이 있기 때문에 본 실험에서는 그들이 제안한 기법의 모드 중 가장 우수한 결과를 보여주는 점진적(progressive) 모드와 $(4,4)-threshold$일 경우에 대해 수행하였고, 다른 기법들의 경우 LSB-2 또는 LSB-3 방법을 사용하였기 때문에 이에 대해 실험을 수행하였다. 제안한 기법의 경우 모든 커버 이미지 내의 픽셀에 대해 LSB-2 방법을 사용했기 때문에 524,288-bit를 숨길 수 있었고, Lin & Tsai[14]의 경우 제안한 기법 내에서 사용된 방법인 네 개 픽셀을 하나의 블록으로 매핑(mapping)하여 비밀이미지의 한 픽셀을 숨기는 과정은 동일하나 블록 내의 첫 번째 픽셀에는 숨기지 않고, 두 번째 픽셀부터 연속적으로 3-bit씩 총 9-bit를 숨긴다. 그러나 이중 1-bit는 오류 수정을 위한 비트(parity bit)의 역할을 수행하기 때문에 삽입 용량 측정에선 제외시켰다. 고로 Lin & Tsai[14] 또한 524,288-bit 숨길 수 있었다. 한편, Chang et al.[17]이 제안한 기법은 인증(authentication)을 가미한 기법이기 때문에 LSB-3 방법을 이용하여 커버 이미지 내의 모든 픽셀에 대해 비밀 데이터 2-bit와 인증 데이터 1-bit 씩을 숨겼다. 이로 인해 다른 기법들에 비해 삽입 용량이 커졌고,

표 2. 8개 테스트 이미지들에 대한 삽입 용량의 실험 결과
(단위: bit)

테스트 이미지	Lin & Tsai[14]	Wang & Shyu[16]	Chang et al.[17]	제안한 기법
Lena	524,288	131,072	786,432	524,288
Baboon	524,288	131,072	786,432	524,288
Airplane	524,288	131,072	786,432	524,288
Peppers	524,288	131,072	786,432	524,288
Boat	524,288	131,072	786,432	524,288
Man	524,288	131,072	786,432	524,288
Elaine	524,288	131,072	786,432	524,288
Woman	524,288	131,072	786,432	524,288

그 결과 786,432-bit를 숨길 수 있었다. Wang & Shyu[16]가 제안한 기법은 앞에서 언급한 것처럼 기존에 제안되었던 것들과 달리 비밀이미지를 분할하여 공유된 이미지를 생성하기 때문에 비교적 삽입 용량은 작지만 광범위하게 사용할 수 있는 장점이 존재한다.

제안한 기법의 경우 비밀이미지의 크기(size) 또는 딜러의 필요에 따라 LSB-1, 2, 3과 같은 다양한 방법들로 숨기는 것이 가능하다. 그러나 뒤에서 언급될 *PSNR*과 삽입 용량의 상관관계 때문에 본 논문에서는 LSB-2 방법을 이용하여 공유된 이미지의 왜곡은 줄이되 삽입 용량은 기존의 제안되었던 기법들과 유사하게 유지되었다.

4.3 *PSNR*의 실험 결과 및 분석

삽입용량에서 비교했던 것과 마찬가지로 제안한 기법과 기존에 제안되었던 3가지 기법들에 대해 *PSNR*의 실험 결과를 분석한다. 표 3은 8개의 그레이스케일 이미지들에 대한 *PSNR*의 실험 결과를 보여준다. 표 3에서 Wang & Shyu[16]이 제안한 기법은 공유된 이미지가 의미 없는(meanless) 것으로 생성되기 때문에 실험에서 제외하였다. Lin & Tsai[14]가 제안한 기법과 본 논문에서 제안한 기법의 삽입 용량은 동일했지만 실제 삽입되는 방식이 틀리기 때문에 제안하는 기법의 *PSNR*의 결과는 더 우수했다. Chang et al.[17]이 제안한 기법은 커버 이미지의 모든 픽셀에 대해 LSB-3 방법을 사용했고, Lin & Tsai[14]가 제안한 기법은 커버 이미지의 한 블록(하나의 블록은 4개의 픽셀로 구성)당 3개의 픽셀에 대해 LSB-3 방법을 사용했기 때문에 일반적으로는 Lin & Tsai[14]가 제안한 기법의 *PSNR*에 대한 결과가 더 우수해야 한다. 그러나 Chang et al.[17]은 이러한 단점을 해결하기 위해 인증으로 사용되는 1-bit를 해당 픽셀값과 유사하게 생성해 삽입하므로 Lin & Tsai[14]가 제안한 기법의 *PSNR* 결과보다 더 우수했다. 이로서 본 논문에서 제안한 기법의 *PSNR* 실험 결과는 기존의 제안되었던 기법들에 비해 우수함을 알 수 있었다. 물론 *PSNR*의 수치를 더 높이기 위해 LSB-1 기법을 사용해도 되지만 삽입 용량과의 상관관계를 고려해 본 논문에서는 LSB-2 기법을 사용하였다. 만약 제안한 기법 내에서 LSB-3 방법을 사용한다면 삽입용량은 Chang et al.[17]이 제안한 기법과 동일하고, *PSNR*의 결과는 대략적으로 40*dB*이 될 것이다. 표 4는 제안한 기법 내에서 공유된 이미지의 업데이트 과정 후 생성된 새로운 공유된 이미지에 대한 *PSNR*의 단계별 실험 결과를 보여준다. 표 4에서 단계별로 업데이트가 되더라도 새롭게 생성된 공유된 이미지들의 *PSNR* 결과는 큰 차이 없이 비슷한 수치를 보여주고 있다. 이를 통해 본 논문에서 제안한 기법은 업데이트 과정을 지속적으로 반복하더라고 새롭게 생성되는 공유된 이미지의 왜곡은 눈으로 확인할 수 없을 정도의 안전함을 보장한다는 것을 알 수 있었다.

한편, 사전 비밀이미지 공유로 인해 기존의 비밀이미지 공유 기법에 비해 통신량(traffic)의 증가가 유발된다. 본 논문에서는 실제 네트워크상에서 딜러

표 3. 8개 테스트 이미지들에 대한 *PSNR*의 실험 결과
(단위: *dB*)

테스트 이미지	Lin & Tsai[14]	Wang & Shyu[16]	Chang et al.[17]	제안한 기법
Lena	39.16	-	40.92	44.15
Baboon	39.15	-	40.92	44.14
Airplane	39.21	-	40.87	44.16
Peppers	39.20	-	40.96	44.16
Boat	39.18	-	40.93	44.04
Man	39.18	-	40.93	44.15
Elaine	39.13	-	40.89	44.13
Woman	39.18	-	40.93	44.15
평균값	39.17	-	40.92	44.14

표 4. 공유된 이미지의 업데이트 과정 후의 *PSNR* 실험 결과
(단위: *dB*)

테스트 이미지	최초 단계 (t=0)	1번째 단계 (t=1)	2번째 단계 (t=2)	…	10번째 단계 (t=10)
Lena	44.15	44.16	44.15	…	44.15
Baboon	44.14	44.13	44.14	…	44.14
Airplane	44.16	44.17	44.16	…	44.17
Peppers	44.16	44.15	44.14	…	44.17
Boat	44.04	44.02	44.03	…	44.06
Man	44.15	44.15	44.14	…	44.15
Elaine	44.13	44.13	44.13	…	44.13
Woman	44.15	44.14	44.14	…	44.13
평균값	44.14	44.13	44.13	…	44.13

와 참가자들 간의 공유된 이미지의 배포 시 발생하는 통신량에 대해서만 언급한다. 딜러가 생성하는 공유된 이미지의 크기를 $M \times M$의 그레이스케일 비트맵 이미지라고 가정한다. 이런 경우에 기존의 비밀이미지 공유 기법과 본 논문에서 제안한 사전 비밀이미지 공유 기법의 데이터 통신량은 각각 $(M \times M \times 8) \times n$ (bit)와 $(M \times M \times 8) \times n \times t$ (bit)이다. 예를 들어, $M = 512$이고 $n = 10$인 경우 기존의 비밀이미지 기법들은 약 20.98 Mbyte이고, 제안한 기법은 약 $20.98 \times t$ Mbyte이다. 즉, 기존의 기법들에 비해 t배 만큼 통신량이 증가한다. 전송속도나 효율성이 떨어지는 네트워크의 경우 제안한 기법은 비효율적이다. 그러나 보안성 관점에서는 기존의 기법들에 비해 우수하므로 보안성이 우선시 되는 경우 제안한 기법을 사용하는 것을 권장한다.

5. 결 론

본 논문에서는 $GF(2^8)$상에서의 사전 비밀이미지 공유 기법을 처음으로 제안하였다. 제안한 사전 비밀이미지 공유 기법은 비 규칙적으로 비밀이미지 공유를 업데이트함으로서 최초 비밀이미지 공유 이후 악의적인 공격자에 의해 비밀이미지의 도청 및 복원과정의 참가가 이루어지는 있는 문제점을 해결하였고, 나머지 연산 수행에서 소수 p를 사용함으로서 발생할 수 있는 이미지 픽셀 값의 손실(lossy)에 대한 문제도 $GF(2^8)$상에서의 기약 다항식을 이용한 나머지 연산 수행을 통해 해결하였다. 또, 삽입 용량과 *PSNR*의 상관관계를 고려하여 LSB-2 기법을 사용하였다. 실험 결과에서는 Chang et al.이 제안한 기법이 삽입 용량 측면에서 우수했지만 실제 비밀데이터로 삽입되는 용량은 제안한 기법과 동일하고, 인증데이터가 1-bit 추가되어 실제 삽입용량이 증가되었다. 그러나 사전 비밀이미지 공유기법을 통해 지속적인 공유된 이미지의 업데이트가 인증역할을 대신 수행할 수 있기 때문에 *PSNR*의 실험 결과가 우수한 수치를 보여주었던 제안한 기법이 더 좋은 안전성을 제공해준다고 사료된다.

향후 연구는 제안한 기법이 가역(reversible)이 가능하도록 하는 것이다. 가역이 가능하기 위해선 비밀공유 시 사용하는 다항식(polynomial)의 형태를 변경하는 것이 중요하므로 이에 대한 선행 연구가 필요할 것으로 사료된다. 또한, 기존의 LSB-2 삽입 방법이 아닌 *PSNR*의 실험 결과가 우수하면서 삽입 용량이 많아지는 연구도 수행되어야 할 것이다.

참 고 문 헌

[1] W. Stallings, *Cryptography and Network Security*, Prentice Hall, 5rd Edtion, 2010, Upper Saddle River, NJ, USA.

[2] A. Menezes, P.C. van Oorschot, and S.A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1996, Boca Raton, FL, USA.

[3] A. Shamir, "How to Share a Secret," *Communications of the ACM*, Vol. 22, No. 11, pp. 612-613, 1979.

[4] G.R. Blakley, "Safeguarding Cryptographic Keys," *Managing Requirements Knowledge, Proc. International Workshop on the National Computer Conference*, pp. 313- 317, 1979.

[5] I. Mitsuru, S. Akira, and N. Takao, "Secret Sharing Scheme Realizing General Access Structure," *Electronics and Communications in Japan (Part III: Fundamental Electronic Science)*, Vol. 72, Issue 9, pp. 1520-6440, 1989.

[6] S. Berry, "A Simple Publicly Verifiable Secret Sharing Scheme and Its Application to Electronic Voting," *Advances in Cryptology-CRYPTO' 99, Lecture Notes in Computer Science 1666*, pp. 148-164, 1999

[7] F. Paul, "A Practical Scheme for Non-Interactive Verifiable Secret Sharing," *Proc. 28th Annual Symposium on Foundations of Computer Science*, pp. 427-438, 1987.

[8] B. Josh and L. Jerry, "Generalized Secret Sharing and Monotone Functions," *Advances in Cryptology-CRYPTO' 88, Lecture Notes in Computer Science 403*, pp. 27-35, 1990.

[9] C-C Chang and R-J Hwang, "Efficient Cheater Identification Method for Threshold Schemes," *IEE Proceedings of Computers & Digital Techcuques*, Vol. 144, No. 1, pp. 23-27, 1997.

[10] A. Beimel and B. Chor, "Secret Sharing with Public Reconstruction," *IEEE Transactions on Information Theory*, Vol. 44, No. 5, pp. 1887-1896, 1998.

[11] M. Naor and A. Shamir, "Visual Cryptography," *EUROCRYPT' 94, Lecture Notes in Computer Science 950*, pp. 1-12, 1995.

[12] M. Naor and A. Shamir, "Visual Cryptography II: Improving the Contrast Via the Cover Base," *EUROCRYPT' 96, Lecture Notes in Computer Science 1189*, pp. 197-202, 1997.

[13] C-C Thien and J-C Lin, "Secret Image Sharing," *Journal of Computers & Graphics*, Vol. 26, No. 5, pp. 765-770, 2002.

[14] C-C Lin and W-H Tsai, "Secret Image Sharing with Steganography and Authentication," *Journal of Systems and Software*, Vol. 73, No. 3, pp. 405 - 414, 2004.

[15] Y-S Wu, C-C Thien and J-C Lin, "Sharing and Hiding Secret Images with Size Constraint," *Journal of Pattern Recognition*, Vol. 37, No. 7, pp. 1377-1385, 2004.

[16] R-Z Wang and S-J Shyu, "Scalable Secret Image Sharing," *Journal of Signal Processing: Image Communication*, Vol. 22, No. 4, pp. 363-373, 2007.

[17] C-C Chang, Y-P Hsieh and C-H Lin, "Sharing Secrets in Stego Images with Authentication," *Journal of Pattern Recognition*, Vol. 41, No. 10, pp. 3130-3137, 2008.

[18] P-Y Lin and C-S Chan, "Invertible Secret Image Sharing with Steganography," *Journal of Pattern Recognition Letters*, Vol. 31, No. 13, pp. 1887-1893, 2010.

[19] M. Tompa and H. Woll, "How To Share a Secret with Chearters," *Journal of Cryptology*, Vol. 1, No. 2, pp. 133-138, 1988.

[20] A. Herzberg, S. Jarecki, H. Krawczyk and M. Yung, "Proactive Secret Sharing Or: How to Cope With Perpetual Leakage," *Proc. the 15th Annual International Cryptology Conference on Advances in Cryptology (CRYPTO' 95)*, pp. 339 - 352, 1995.

[21] A. Cheddad, J. Condell, K. Curran and P. McKevitt, "Digital Image Steganography: Survey and Analysis of Current Methods," *Journal of Signal Processing*, Vol. 90, No. 3, pp. 727-752, 2010.

[22] B. Li, J. He, J. Huang and Y.Q. Shi, "A Survey on Image Steganography and Steganalysis," *Journal of Information Hiding and Multimedia Signal Processing*, Vol. 2, No. 2, pp. 142-172, 2011.

[23] 김기종, 신상호, 유기영, "비밀데이터의 패턴정보에 기반한 새로운 정보은익 기법," 멀티미디어학회논문지, 제15권, 제4호, pp. 526-539, 2012.

[24] 장봉주, 이석환, 권기룡, "전염성 정보은닉 시스템을 위한 능동형 비디오 워터마킹 기법," 멀티미디어학회논문지, 제15권, 제8호, pp. 1017-1030, 2012.

현 승 일

1990년 2월 계명대학교 전자계산학과 공학사
1999년 2월 성균관대학교 전기전자컴퓨터공학과 공학석사
2013년 2월 현재 경북대학교 컴퓨터공학과 공학박사 수료
1999년 9월~2002년 2월 영진전문대학 전자정보계열 교수
2002년 2월~현재 영진사이버대학 정보통신공학계열 교수
관심분야 : 멀티미디어시스템, 암호학, 분산시스템

신 상 호

2006년 8월 금오공과대학교 응용수학/컴퓨터공학 학사
2008년 8월 경북대학교 전자전기컴퓨터학부 공학석사
2009년 3월~현재 경북대학교 전자전기컴퓨터학부 박사과정
관심분야 : 고속암호 알고리듬, 양자암호, 클라우드 컴퓨팅 보안

유 기 영

1976년 2월 경북대학교 수학교육과 이학사
1978년 2월 한국과학기술원 전산학과 공학석사
1992년 3월 미국 Rensselaer Polytechnic Institute 전산학과 공학박사
1978년 3월~현재 경북대학교 IT대학 컴퓨터학부 교수
관심분야 : 암호학, 정보보호, 유비쿼터스보안, 네트워크보안, 데이터베이스보안, 스테가노그라피, 인증프로토콜