Cover-data의 유사성을 고려한 StegoWaveK

김 영 실* 김 영 미** 김 륜 옥* 최 진 용**
*대림대학 컴퓨터정보과 **(주) Cest

요 약

상용화된 Wave Steganography가 가지고 있는 Mask가 쉽게 필터링 될 수 있다는 문제점을 개선하고 Capacity를 높여 Cover-data 선택의 폭을 확장시킬 수 있는 δSAcc234bit Mask Data Insertion 알고리즘을 설계 구현하였다. 또한 Mask 의 보안수준을 향상시키기 위해 3-Tier 파일 암호화 알고리즘도 제안하고 적용하였다. 3-Tier 파일 암호화 알고리즘은 파일에 따라 Ciphertext에 특정한 패턴이 나타난다는 문제점을 개선한 알고리즘이다. 본 논문에서 제안한 δSAcc234bit Mask Data Insertion 알고리즘과 3-Tier 파일 암호화 알고리즘을 이용하여 Wave Steganography를 수행해주는 StegoWaveK 모델을 설계 구현하였다.

StegoWaveK based on the Correlation Relation

Young-Shil Kim* Young-Mi Kim** Ryun-Ok Kim* Jin-Yong Choi**

ABSTRACT

A design implemented the δ SAcc234bit Mask Data Insertion algorithm that can let the Mask which commercialized Wave Steganography had improved the problem that a filter ring was able to easily become and raised Capacity and extend the width that Cover-data was alternative. Also, it applied 3-Tier file encryption algorithm with a proposal in order to improve a security level of Mask. 3-Tier file encryption algorithm is the algorithm that a specific pattern improved the problem that appeared in Ciphertext according to a file. A design implemented the StegoWaveK model carried out Wave Steganography, using δ SAcc234bit Mask Data Insertion algorithm and the 3-Tier file encryption algorithm that proposed in this paper.

1. 서 론

컴퓨터와 통신시스템의 비약적인 발전으로 인하여 데이터를 안전하게 보호하기 위한 방안들이 개발되고 있다. 가장 근원이 되는 기법이 암호화이며 이와 더불어 다양한 형태의 정보 은닉 기술이 연구되고 있다[1,9]. 이중 대표적인 응용기술이 Steganography이다. Steganography는 데이터를 다양한 형태의 자료(텍스트, 이미지, 동영상, 오디오)에 은닉함으로써 일반 사용자는 숨겨진 데이터를 찾아 내지 못하도록 지원하는 기술이다. 일반적으로 비밀 메시지를 암호화하여 은닉하게 되면 은닉된 비밀 데이터가 Attacker에 의해 발견된다 하여도 Attacker는 그 암호화된 메시지를 복호화하기 위해 노력을 해야 하기 때문에[2] 그 비밀 메시지에 대한 보안 수준이 한층 더 안전하게 높아지게 된다. 또한 은닉하려고 하는 비밀 데이터의 크기에 따라 Cover-data의 크기가 결정되므로 비밀 데이터를 압축하여 Cover-data에 은닉하고 있다. 현재 가장 일반적으로 발전된 Steganography 분야가 이미지를 이용한 Steganography이며 가장 이슈가 되는 분야는 오디오를 이용한 Steganography이다. 일반적으로 Audio Steganography에서는 음악을 듣는 청취자가 실제로 청취하지 못하는 부분에 데이터를 숨기기 때문에 실제 데이터가 숨겨져 있다는 것을 일반 청취자들은 알지 못한다. 일반적으로 Audio Steganography에서 응용되고 있는 Lowbit Encoding은 일률적으로 마지막 비트에 Mask 비트 스트림의 한 비트씩을 순차적으로 삽입하기 때문에 비록 청취자들이 Stego-data 파일에 어떤 데이터가 숨겨져 있다는 것을 알 수 없다 하여도 쉽게 은닉된 Mask를 필터링할 수 있다.

이러한 문제들을 해결하기 위해 본 논문에서 제안한 StegoWaveK는 Attacker에 의해 Mask가 필터링 되었다 하여도 복호화를 어렵게 하기 위해 3-Tier 파일 암호화 알고리즘을 이용하여 암호화를 수행하였다. 또한, Cover-data에 은닉할 수 있는 Mask 의 크기가 제한적이라는 문제점을 개선하기 위해 삽입되는 비트 수를 Cover-data와 유사한 특성을 가지는 범위인 2비트, 3비트, 4비트로 지정하며, Lowbit를 기준으로 Mask를 삽입하기 때문에 쉽게 Attack 될 수 있다는 문제점을 개선하기 위해 Mask가 삽입되는 위치의 기준을 Lowbit 가 아닌 Wave 파일의 16비트 단위를 10진수로 변환한 값 중 Cover-data의 특성이 유지될 수 있는 임계 치를 찾아 해당 비트에 Mask를 삽입한다.

2. StegoWaveK 모델

StegoWaveK는 Cover-data를 Wave 파일로 지정하여 다양한 유형의 Mask를 은닉할 수 있는 오디오 Steganography 시스템이다. 은닉하려고 하는 데이터의 보안 수준을 높이기 위해 3-Tier 파일 암호화 알고리즘을 이용한다. 그리고 Cover-data에 삽입되는 Mask의 Capacity를 높이고 Attacker로부터의 Attack에 대처하기 위해 Mask가 삽입되는 위치를 동적으로 지정하여 삽입하는 δSAcc234bit Mask Data Insertion Algorithm을 제안하고 적용하였다.

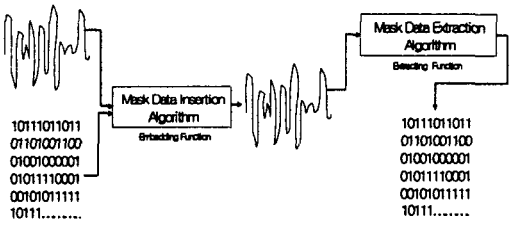


그림 1 StegoWaveK 모델

2.1 3-Tier 파일 암호화 알고리즘

3-Tier 파일 암호화 알고리즘은 보안의 수준을 높이기 위한 파일의 암호화뿐만 아니라 기존의 파일 암호화 알고리즘이 특정한 패턴을 나타낸다는 문제점을 개선할 수 있는 파일 암호화 알고리즘이다.

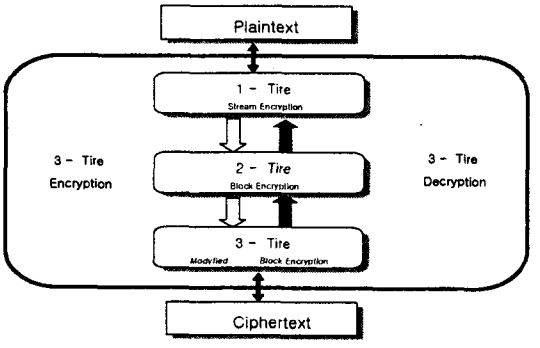


그림 2 3-Tier 파일 암호화 알고리즘

1-Tier는 파일 암호화를 위해 1차적으로 암호화를 수행하는 단계이다. 최초의 Plaintext 암호화 시 발생할 수 있는 오류의 단위를 최소화하기 위해 스트림 암호화 알고리즘 중 RC4 알고리즘을 적용하였다. RC4는 256바이트 이하의 가변 키를 사용하는 알고리즘이다[3,4,12,13].

2-Tier 는 암호화 수준을 한 단계 더 높이기 위하여 대체(substitution)와 치환(permutation)이라는 기본적인 암호화를 16번 반복하여 적용하는 TDES 블록 암호화 알고리즘을 적용하는 단계이다. TDES 블록 암호화 알고리즘은 DES를 2개의 서로 다른 키로 3번 암호화하는 알고리즘이다[3,4,14].

3-Tier 는 암호화된 Ciphertext에 나타날 수 있는 특정한 패턴을 제거하기 위해 2차 암호화된 Ciphertext의 구조와 형식을 은닉하는 단계이다. 블럭 암호화 알고리즘은 Key외에 운영모드에 사용되는 초기화 백터 IV(Initialization Vector)가 암호화를 위한 매개변수로 사용되기 때문에, key를 해독한다고 하더라도, IV를 알지 못하면 암호문을 해독할 수 없다는 장점을 가지고 있다[16]. 따라서 블럭 암호화 알고리즘을 변형시킨 MBE(Modyfied Block Encryption) 알고리즘을 제안하고 적용하였다. MBE 알고리즘은 Ciphertext 블록들의 난수를 생성한 후 이를 기반으로 한 키로부터 생성해낸 255개의 키를 순차적으로 순환하며 각 블록과 XOR 연산을 수행한다. XOR 연산 수행 시 앞 블록의 결과 데이터가 다음 블록의 결과에 영향을 미치도록 구현되어 있기 때문에 Ciphertext의 구조와 형식을 은닉할 뿐만 아니라 한번의 암호화가 더 수행되는 효과를 얻게 되는 단계이다. 다음은 3-Tier에 적용된 MBE 알고리즘이다.

Input : $d_{(0)},...d_{(n)}$ created block data from Plaintext
$k_{(0,...255)}$ created key from the table of secret message in numbers
Ouput : $o_{(0)}...o_{(n)}$ result calculated per block

1. $o_{(0)} \leftarrow d_{(0)} \oplus k_{(0)}$
2. For i from 1 to n -1
   $o_{(i)} \leftarrow d_{(i)} \oplus k_{(i \bmod 255)} \oplus o_{(i-1)}$

그림 3 MBE 알고리즘

특히 2-Tier와 3-Tier 단계에서 적용된 암호화 알고리즘은 블록 암호화 알고리즘이므로 Plaintext 크기가 블록의 배수가 되지 않을 경우에는 마지막 블록에 데이터를 입력해야 하는 padding을 수행해야 한다. 기존에 제안된 padding 기법을 적용할 경우 정형화된 데이터가 입력되어 padding 영역의 값이 일정한 패턴을 가지게 되므로 쉽게 padding 영역을 찾을 수 있다. 이러한 문제점을 보완하기 위해 SELI(Select and Insert) padding 기법을 제안하고 적용하였다. SELI padding 기법은 실제 데이터가 입력된 블록의 특정 위치 값을 padding 영역에 삽입하므로, padding된 데이터뿐만 아니라 padding

의 수행여부도 알지 못하도록 하는 효과를 얻는 기법이다. SELI padding 기법의 기본 알고리즘은 다음과 같다.

n : count of block
Pad_{byte} : padding byte count of last block
$NPad_{byte}$: real data byte count of last block
k : insert position
$Value(i)$: real value of i byte

1. $value((n-1)*8+ NPad_{byte}) \leftarrow value((n-1)*8+ NPad_{byte}-1))$,
2. $value((n-1)*8+ NPad_{byte}-1)) \leftarrow Pad_{byte}$
3. For i from 0 to $Pad_{byte}-1$
   $value((n-1)*8+NPad_{byte}+1+i) \leftarrow value(i*8+k)$

그림 4 SELI padding 알고리즘

2.2 δSAcc234bit Mask Data Insertion Algorithm

가장 일반적으로 구현된 Audio Steganography 기법이 바로 Lowbit Encoding 즉 마지막 비트에 Mask 비트를 1 비트씩 삽입하는 방법이다 [9,10]. 다음 [그림 5]는 오디오 샘플안에 "A"를 Lowbit Encoding 경우의 예이다.

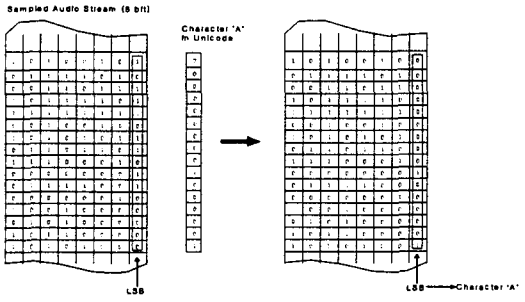


그림 5 오디오 샘플에 "A"를 Lowbit Encoding한 예

Lowbit Encoding의 경우 Attacker가 쉽게 은닉된 Mask를 Attack 할 수 있다는 단점을 가지고 있다. 그럼에도 사용되는 가장 큰 이유는 Cover-data와 Lowbit Encoding 후에 생성된 Stego-data 사이에 차이가 거의 없기 때문이다. 즉 Stego-data가 Cover-data와 같은 특성을 가지고 있기 때문이다. 본 논문에서 제안하는 δSAcc234bit Mask Data Insertion 알고리즘은 Stego-data가 Cover-data와 유사한 특성을 나타내는 범위 내에서 Capacity를 향상시키고 Attacker가 Attack을 쉽게 할 수 없도록 개선하였다.

첫째 Audio Steganography가 가지는 크기에 대한 제한점을 개선한 것이다. 제한된 size의 Cover-data에 좀 더 큰 size의 Mask를 삽입하기 위해선 Stego-data가 Cover-data와 유사한 특성을 가지는 범위 내에서 Cover-data에 은닉되는 Mask의 비트수를 1비트 이상으로 지정하면 된다. 다음 [그림 6]은 Cover-data에 Mask를 1비트, 2비트, 4비트, 6비트로 삽입한 Stego-data 와 Cover-data와의 유사성을 분석하기 위해 상관분석을 수행한 결과이다.

Variable	N	Average	Standard deviation	Sum
Origin	3000	94.03567	1637	282107
1bit	3000	94.04667	1637	282140
2bit	3000	94.08533	1637	282256
4bit	3000	94.00633	1637	282019
6bit	3000	93.69833	1638	281095

Varibale	Min	Max
Origin	-9382	11835
1bit	-9381	11834
2bit	-9382	11835
4bit	-9390	11826
6bit	-9391	11791

N=3000

	1bit	2bit	4bit	6bit
Origin	1.0000	1.0000	0.99999	0.99988
	<0.0001	<0.0001	<0.0001	<0.0001

그림 6 Correlation Analysis of Cover-data and various Stego-data I

분석한 결과를 살펴보면 Cover-data와 각각의 Stego-data의 상관계수가 1에 가깝기 때문에 Cover-data의 특성을 거의 유지하고 있음을 알 수 있다. 그러나 6비트를 삽입한 Stego-data

만이 상관계수가 0.99988로서 Cover-data의 특성 이 어느 정도 누락되는 부분이 있음을 알 수 있다. 이로부터 Capacity를 높일 수 있는 이상적인 삽입 비트 수가 **2비트부터 4비트 사이라는 정보를 얻게된다.**

둘째 Wave 파일의 마지막 비트에 2비트, 3비트, 4비트를 일률적으로 은닉하게 되면 Attacker에 의해 쉽게 필터링 될 수 있다. 이러한 문제점을 개선하기 위해 Cover-data에 Mask 삽입 시 고정된 위치에 Mask 비트를 삽입하는 것이 아니라 삽입되는 비트의 위치가 사인곡선의 형태가 되도록 Mask를 삽입한다.

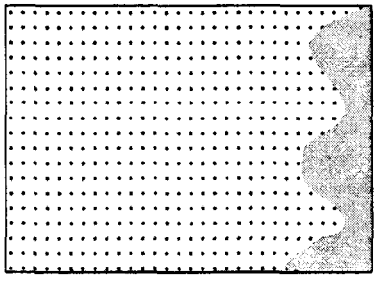
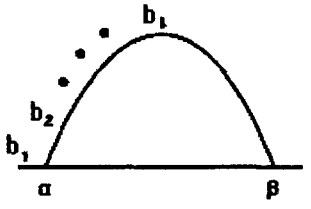


그림 7 Stego-data

실제 구현된 Audio Steganography의 경우 i 비트마다 1 비트씩 Mask를 삽입하기 때문에 Mask가 삽입된 위치가 필터링 될 확률이 다음과 같다.



(그림 8) 사인곡선의 형태로 삽입된 Mask의 위치

(그림 8)과 같이 일정한 간격으로 사인곡선 형태를 이루도록 Mask 비트를 삽입하는 경우 사인곡선의 한 주기에 따라 i 비트마다 삽입된

그림 8 사인곡선의 형태로 삽입된 Mask의 위치 과

식(3)과 같다. 즉 i 비트마다 Mask의 1비트를 은닉하였으므로, 확률 $P(b_j) = \frac{1}{i}$이 된다., 필터링 된 a_j 위치를 이용하여 a_{j+1}의 위치를 알아낼 수 없고 $P(b_j)$와 마찬가지로 $P(b_{j+1}) = \frac{1}{i}$이 되므로 서로 독립사상이다. 따라서 Mask가 삽입된 위치를 알아낼 확률은 식(4)와 같으며, 사인곡선의 한 주기에 은닉되는 Mask의 비트수가 필터링 될 확률은 16의 지수승 만큼 작아지게 된다.

$$P(b_j) = \frac{1}{i} \quad (i=16,\ j\in\{1,2,3\ \cdots\ 16\}) \tag{2}$$

$$P(b_1, \ldots, b_j) = \frac{1}{i^j} \tag{3}$$

그러므로 Lowbit를 기준으로 무조건 2비트, 3비트, 4비트를 삽입하는 것 보다 사인 곡선의 형태가 되도록 2비트, 3비트, 4비트를 삽입하는 것이 쉽게 Attack 되지 않는다.

다음은 Cover-data와 2비트 삽입, 4비트 삽입, 사인곡선 234비트로 삽입된 Stego-data와의 상관관계를 분석한 결과이다.

변수	N	평균	표준편차	합
Origin	3000	94.03567	1637	282107
2Bit	3000	94.08533	1637	282256
4Bit	3000	94.00633	1637	282019
SAcc234Bit	3000	94.23600	1637	282768

단순통계

변수	최소값	최대값
Origin	-9382	11835
2Bit	-9382	11835
4Bit	-9390	11826
SAcc234Bit	-9384	11828

피어슨 상관계수 N=3000

	2Bit	4Bit	SAcc234Bit
Origin	1.0000	1.0000	1.0000
	<0.0001	<0.0001	<0.0001

그림 9 Cover-data와 다양한 Stego-data의 상관분석

<표 1> 2비트 삽입 Stego-data와의 -검정

	Cover-data	2비트
평균	94.03567	94.08533
분산	2680295	2680205
관측수	3000	3000
자유도	2999	2999
F 비	1.000033	
P(F<=f) 단측 검정	0.499635	
F 기각치: 단측 검정	1.061923	

<표 2> 4비트 삽입 Stego-data와의 F-검정

	Cover-data	4비트
평균	94.03567	94.00633
분산	2680295	2680163
관측수	3000	3000
자유도	2999	2999
F 비	1.000049	
P(F<=f) 단측 검정	0.499464	
F 기각치: 단측 검정	1.061923	

<표 3> SAcc234비트 삽입 Stego-data와의 F-검정

	Cover-data	SAcc234비트
평균	94.03567	94.256
분산	2680295	2679531
관측수	3000	3000
자유도	2999	2999
F 비	1.000285	
P(F<=f) 단측 검정	0.496889	
F 기각치: 단측 검정	1.061923	

분석한 결과를 살펴보면 Lowbit를 기준으로 2비트, 4비트를 삽입한 Stego-data와 사인곡선의 형태로 Mask가 삽입된 Stego-data 모두 Cover-data와 높은 상관관계가 나타남을 알 수 있다. 따라서 Mask 삽입 시 사인곡선을 이용하여도 Cover-data의 특성을 유지할 수 있다.

[그림 22]은 Cover-data에 Mask를 234비트씩 은닉하는 알고리즘이다.

```
Procedure 234Mask_Data_Insertion();
begin
    Cover-data read;
    Calculate  a Insertion number of Mask data;
    for( i=0 ; i<Insertion_number ; i++)
      begin
          Read Insertion bit of Mask data;
          Loaded Mask data overwrite into
          Low bit of Cover-data;
      end;
end;
```

그림 10 234비트 Mask Data Insertion 알고리즘

셋째 Mask가 Lowbit를 기준으로 삽입되기 때문에 사인곡선의 형태로 Mask를 삽입하더라

도 마지막 비트에 숨겨진 Mask는 Attacker에 의해 필터링 될 수 있다. 이러한 문제점을 개선하기 위해 Wave 파일의 16비트 단위를 10진수로 변환한 값을 기준으로 Cover-data의 특성을 유지할 수 있는 범위 내에서, 임계 치를 지정하여 임계 치 위치를 기준으로 Mask를 삽입한다.

다음은 Lowbit를 기준으로 234비트가 삽입된 Stego-data, 16비트 단위를 10진수로 변환한 값이 임계 치 이상이 되는 지점을 기준으로 1비트가 삽입된 Stego-data 그리고 임계치를 기준으로 사인곡선 234비트로 삽입된 Stego-data가 Cover-data와 얼마나 유사한지를 분석하였다. 분석된 결과를 살펴보면 모두 Cover-data와 거의 같은 특성을 가지고 있는 것을 알 수 있다.

변수	N	평균	표준편차	합
Origin	3000	37.454333	57.77347749	112363
SAcc234Bit	3000	39.156333	60.04369327	117469
δ1Bit	3000	37.288333	57.76556679	111865
δSAcc234Bit	3000	37.502667	60.86745135	112508

단순통계

변수	최소값	최대값
Origin	-9934	9486
SAcc234Bit	-9382	9502
δ1Bit	-9930	9488
δSAcc234Bit	-9930	9494

피어슨 상관계수 N=3000 (δ = 20,000)

	SAcc234Bit	δ1Bit	δSAcc234Bit
Origin	1.0000	1.0000	1.0000
	<0.0001	<0.0001	<0.0001

그림 11 Cover-data와 다양한 Stego-data의 상관분석

<표 4> δ1비트 삽입 Stego-data와의 F-검정

	Cover-data	δ1Bit
평균	37.45433333	37.288333
분산	10009986.33	10007245
관측수	3000	3000
자유도	2999	2999
F 비	1.000273909	
P(F<=f) 단측 검정	0.49700857	
F 기각치: 단측 검정	1.061922772	

<표 5> δSAcc234비트 삽입 tego-data와의 F-검정

	Cover-data	δSAcc234Bit
평균	37.45433333	37.5026667
분산	10009986.33	11110835.1
관측수	3000	3000
자유도	2999	2999
F 비	0.900921153	
P(F<=f) 단측 검정	0.499073892	
F 기각치: 단측 검정	0.941688061	

<표 6> SAcc234비트 삽입 Stego-data와의 F-검정

	Cover-data	SAcc234Bit
평균	37.4543333	39.156333
분산	10009986.3	10812130
관측수	3000	3000
자유도	2999	2999
F 비	0.92581076	
P(F<=f) 단측 검정	0.49766103	
F 기각치: 단측 검정	0.94168806	

3. 3-TierWaveK 시스템 성능 평가

3.1 HAS(Human Auditory System) 측면

HAS(Human Auditory System) 측면에서 분석하기 위해 파일의 크기가 서로 다른 13개의 Plaintext와 4개의 Wave파일을 Cover-data로 선정하여 기존 시스템과 제안한 시스템을 비교 분석하였다. Lowbit Encoding을 사용하는 상용화된 Audio Steganography에 은닉된 Stego-data와 StegoWaveK 시스템을 통해 Mask가 은닉된 Stego-data를 100명의 학생들에게 들려주었다. 이는 다소 주관적인 판단에 의한 분석이기는 하나 대다수의 실험대상 학생들이 Cover-data와 Stego-data 음악파일의 차이를 인식하지 못했다. 다음 [그림 12]은 실험결과를 나타낸 그림이다.

3.2 FS(File Structure) 측면

Cover-data와 다양한 형태로 Mask가 삽입된 Stego-data를 비교하기 위해 Bitcmp 모듈과 Wavecmp 모듈을 구현하였다. Bitcmp모듈은 Cover-data와 Stego-data가 몇 비트 다른 가를 비교 하며, Wavecmp는 각 비트의 위치에 가중치를 주어 Cover-data와 Stego-data가 얼마만큼 다른가를 비교분석하는 모듈이다. [그림 13]은 Lowbit를 기준으로 2bit, 4bit, 6bit, Acc246bit, S234bit, SAcc234bit로 삽입한 Stego-data와 임계 치를 기준으로 δ1bit, δ2bit, δ3bit, δ4bit, δ5bit, δ6bit, δSAcc234bit로 Mask를 삽입한 Stego-data들이 Cover-data와 몇 비트 다른가를 보여주고 있다. 가장 적은 비트 수를 나타내는 방법이 임계 치를 기준으로 Mask가 삽입된 δSAcc234bit Stego-data라는 것을 알 수 있다. 또한 [그림 14]를 살펴보면 임계 치 이상이 되는 지점을 기준으로 1비트 삽입한 Stego-data와 사인곡선 누적 234비트를 삽입한 Stego-data 그리고 Lowbit를 기준으로 234비트 삽입한 Stego-data가 Cover-data와의 비트 값 차이가 적게 나타남을 알 수 있다. 즉 임계치 이상이 되는 지점을 기준으로 1비트 삽입한 Stego-data가 Cover-data와 가장 유사한 것으로 나타난다. 하지만 Capaciry 문제를 효율적으로 개선하기 위해서는 δSAcc234bit 적용해야 한다.

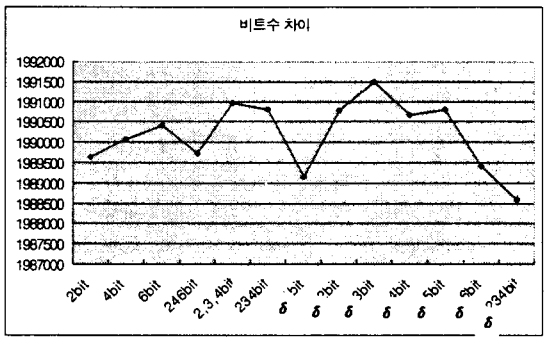


그림 13 Cover-data와 Stego-data와의 비트 수차이

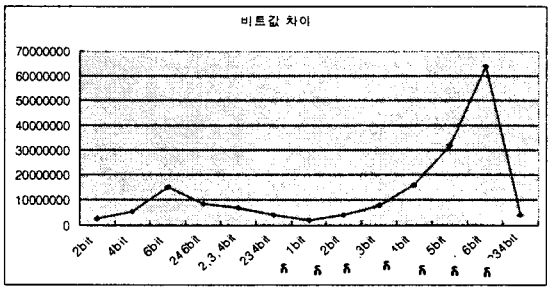


그림 14 Cover-data와 Stego-data와의 비트 값 차이

4. 결 론

정보를 은닉하기 위해 응용되고 있는 Wave 파일을 이용한 Audio Steganography는 Cover-data나 Stego-data의 파일 size가 같기

때문에 일반 청취자나 사용자는 Stego-data에 정보가 은닉되어 있다는 사실을 알 수 없다. 일반적으로 Mask를 은닉하기 위해 Lowbit Encoding을 사용하기 때문에 산술적으로 은닉하고자 하는 Mask size의 16배 이상 큰 Cover-data가 필요하게 된다. 뿐만 아니라 Attacker에 의해 Attack 되기가 쉽다. 이러한 문제점을 개선하기 위해 Cover-data와 유사한 특성을 가지면서 Capacity를 향상시키고 Attacker로부터의 Attack에 강한 Mask Data Insertion 알고리즘을 제안하고 구현하였다. 본 논문에서 제안한 StegoWaveK는 Lowbit Encoding의 문제점을 개선한 모델이다.

첫째 Cover-data의 Capacity를 향상시키며, Attacker에 의해 쉽게 Attack 될 수 있다는 문제점을 해결하기 위해 10진수로 변환한 값이 임계치 이상이 되는 지점을 기준으로 사인곡선의 형태가 되도록 234비트를 삽입하였다. 이는 제안한 모델이 Private-Key를 사용하는 Steganograpy임에도 불구하고 Public-Key Steganography와 같이 은닉된 정보의 유무를 구별하기가 힘든 효과를 갖게 한다.

둘째 Mask를 삽입하기 전에 일반적으로 Mask의 특성을 가시적으로 확인할 수 없고 Stego-data와 Cover-data로부터 Mask을 얻었다고 해도 그 내용을 파악할 수 없도록 하기 위해 암호화를 수행한다. 하지만 현재 파일 암호화를 위해 사용되고 있는 대부분의 비밀키 암호화 알고리즘을 적용한 Ciphertext들을 살펴보면 도면이나 실행파일과 같은 특정한 데이터에 특정한 패턴이 나타나는 특징을 가지고 있다. 이는 곧 Attacker에게 파일의 특성을 알려주는 결과를 초래하게 된다. 이러한 문제점을 해결하기 위해 파일 암호화를 3단계로 처리해주는 3-Tier 파일 암호화 알고리즘을 제안하고 적용하였다.

향후에는 Mask가 삽입될 위치 즉 δ 를 특정한 값으로 고정하는 것이 아니라 Cover-data와 Mask의size 특성에 따라 동적으로 지정할 수 있는 모델 연구가 필요하다. 더 나아가 Mask와 Cover-data의 크기 비율에 따라 시스템에서 Mask를 가능한 한 Filtering이 되지 않으면서 Capacity를 높여주는 방법의 알고리즘들을 동적으로 선택하는 Audio Steganograpny를 구현하는 모델 연구가 필요하다.

참고문헌

[1] J.Zollner, H.Federrath, H.Klimant, A.Pfitzmann, R.Piotraschke, A.Westfeld, G.Wicke, G.Wolf, "Modeling the security of steganographoc systems", 2nd Workshop on Information Hiding, Portland, LNCS 1525, Springer-Cerlag, pp.345-355, April 1998.

[2] http://www.cbcis.wustl.edu/~adpol /courses/cs502/project/report/node1htm

[3] 박창섭, "암호이론과 보안", 대영사, pp. 13-38, 1999.

[4] Raymond G. Kammer "DATA ENCRYPTION STANDARD", Federal Information Processing Standards Publication 1999.

[5] http://www.securitytechnet.com .

[6] 이동훈, 임채훈, "국제/업계 표준 암호 알고리즘 및 프로토콜의 이해", (주) 퓨처 시스템 암호치계 센터, pp.2-10, 2000.

[7] 이종근, 최돈승, 이경석, "정보보호를 위한 암호운영방식 고찰", pp 13-18 ,창원대학교 정보통신연구소 논문집 제 2집, 1998.

[8] http://www.softforum.co.kr/learningcenter/l earningcenter_04.html

[10] 김현곤, 원동호 외 12, "지적 재산권 보호를 위한 정보 은닉 기술 및 표준화 연구", 한국 전산원, pp.19-41, 2000.

[9] Ross J. Anderson, Fabieb A.P. Petitcolas, "On The Limits of Steganography", IEEE Journal of Selected Areas in Communication, 16(4):474-481, May 1998.

[10] S.K. Pal, P.K. Saxena, S.K. Muttoo, " The Future of Audio Steganography".

[11] http://www-ccrma.stanford.edu/CCRMA/Courses/422/projects/WaveFormat

[12] http://www.wisdom.weizmann.ac.il/~itsik/RC4/rc4.html

[13] http://burtleburtle.net/bob/rand/isaac.html

[14] http://www.itl.nist.gov/fipspubs/fip81.htm

[15] RG van Schyndel, AZ Trikel, CF Osborne, "Digital Watermak", International Covference on Image Processing, vol 2 pp.86-90

[16] http://crypto.jmi.co.kr/product/package/ceal/block.html