http://dx.doi.org/10.14400/JDC.2014.12.11.233

비밀분산기법을 이용한 분산 트랜스코딩 시스템 연구

송유진*, 구석모**, 김의창***
동국대학교 경영학부*, 동국대학교 테크노경영협동과정**, 동국대학교 경영학부***

A Study on the Distributed Transcoding System using Secret Sharing Techniques

You-Jin Song*, Seokmo Gu, Yei-Chang Kim*****
Dept. of Business Administration, Dongguk University*
Dept. of Technology Management Cooperative Program, Dongguk University**
Dept. of Business Administration, Dongguk University***

요 약 초고해상도 콘텐츠는 파일 크기가 매우 크기 때문에 기존의 부호화 기술로는 네트워크를 통해 전송하는 것이 불가능하다. 고효율의 부호화 기법인 HEVC를 이용하면 네트워크 전송이 가능하나 압축시간이 많이 필요하기 때문에 분산 트랜스코딩 시스템이 필요하다. 분산 트랜스코딩 시스템은 데이터를 분산하여 저장한 뒤 다수의 노드를 이용하여 부호화한다. 그러나 분산 트랜스코딩 시스템은 분산된 정보가 노출되거나 내부관리자의 공격에 취약하다는 문제점이 있다.
본 논문에서는 초고해상도 콘텐츠를 트랜스코딩 할 때, 분산 트랜스코딩 시스템의 기밀성이 보장되지 않는다는 문제점을 해결하고자 한다. 우리는 SNA를 이용하여 데이터 노드에서 HEVC로 부호화된 콘텐츠 데이터를 비밀분산기법을 통해 암호화하여 저장했다. 결과적으로 안전한 분산 트랜스코딩이 가능하고, 내부관리자의 공격을 방지할 수 있다.

주제어 : 비밀분산, 분산 트랜스코딩, SNA, HEVC, Hadoop

Abstract Ultra high-resolution content, the file size is very large, therefore existing encoding techniques, it is not possible to transmit via the network. Efficient use of the network encoder HEVC corporation can be transferred. Compression requires a lot of time because it requires a distributed transcoding system.
Distributed transcoding system is a distributed data store, and then encoded using a large number of nodes. The disadvantage of distributed transcoding system for distributed information is exposed or vulnerable to attack by internal managers.
In this paper, when the super high definition content transcoding, distributed transcoding system does not guarantee the confidentiality of the problem to solve. We are using SNA, HEVC encoded content data encrypted using the secret distributing scheme was. Consequently, secure shared transcoding is possible, the internal administrator could prevent the attack.

Key Words : Secure Sharing, Distribution Transcoding, SNA, HEVC, Hadoop

Received 22 July 2014, Revised 27 October 2014
Accepted 20 November 2014
Corresponding Author: Yei-Chang Kim(Dongguk University)
Email: kimyc@dongguk.ac.kr

ISSN: 1738-1916

1. 서론

네트워크 기술의 발달로 기존 콘텐츠보다 우수한 고화질 콘텐츠를 사용자에게 제공하는 것이 가능해졌다. 최근 제작되고 있는 초고해상도 콘텐츠는 부호화 과정 없이 현재의 네트워크 대역폭으로 전송하는 것은 불가능하다. 따라서 압축률이 매우 높은 HEVC(High Efficiency Video Coding) 부호화 기법을 이용하여 초고해상도 콘텐츠를 전송할 수 있도록 압축해야 한다. 그러나 초고해상도 콘텐츠를 부호화하는 트랜스코딩 시스템은 현재의 단일 시스템으로 부호화 처리하는 것이 어렵기 때문에 다수의 노드를 이용하여 부호화 처리하는 분산 트랜스코딩 시스템을 구성하여 전송 가능한 상태로 트랜스코딩을 수행한 후, 실시간 스트리밍 프로토콜을 이용하여 사용자가 소유하고 있는 다양한 형태의 디바이스로 전송하고 있다. 하지만 현재의 스트리밍 서비스는 사용권한이 없는 사용자의 접근을 차단하기 어렵고, 서비스를 제공하는 서버로 부터 사용자에게 전송하는 과정에서 악의적인 공격자가 데이터 스트림을 분석하여 원본 영상을 쉽게 취득할 수 있다는 문제점이 있다.

분산 트랜스코딩 시스템은 콘텐츠를 블록단위로 분산처리하기 위해 다수의 노드에 분산 저장되어 있어 콘텐츠 소유자는 어떤 노드에 자신의 데이터가 저장되어 있는지 확인할 수 없다. 그러나 내부관리자는 특정 데이터를 선별적으로 검색하여 정보를 확인하는 것이 가능하며 분산되어 있는 블록의 위치정보까지 확인할 수 있다. 기밀성이 보장되지 않은 상태로 분산 트랜스코딩 시스템을 사용할 경우, 악의적인 내부관리자에 의해 중요한 콘텐츠가 유출될 수 있다. 이러한 문제점을 해결하기 위해 HEVC로 부호화된 콘텐츠 데이터를 비밀분산기법을 이용하여 기밀성이 보장된 형태로 저장하여 콘텐츠 소유자는 자신의 원본데이터를 악의적인 내부관리자에게 노출되는 것을 방지해야 한다[1].

본 논문에서는 SNA(Secure Node Agent)를 이용하여 데이터 노드에서 HEVC로 부호화된 콘텐츠 데이터를 비밀분산기법을 통해 암호화하여 저장하도록 했다. 실험결과, 안전한 분산 트랜스코딩이 가능하고 내부관리자의 공격을 방지할 수 있다.

2. 비밀분산기법

비밀분산기법(Secret Sharing Scheme)은 데이터의 기밀성이 유지된 정보를 사전에 다양한 형태로 분할, 보관하여 사용자에 요청에 따라 분산 정보를 수집하여 암호화된 데이터를 복호화 하는 기법이다[2]. 비밀분산 방식의 대표적 방법으로 Shamir의 임계치법이 있다. 암호화된 정보를 정수 a로 가정한다[3].

정보 p를 a, n이상의 소수라 하고 무작위로 선택된 $k-1$개의 p미만의 정수를 $a_1, a_2, ..., a_{k-1}$로 한다. 이 때, 다항식 $f(x)$를$f(x) = a + a_{1x} + a_2 x^2 + ... + a_{k-1} x^{k-1} (\mathrm{mod} p)$로 정의한다.

사용자 j는 $f(ID_j)$을 준다. 여기서, $f(ID_j)$을 분산정보라 한다. ID_j는 사용자 j의 식별정보이며, $i \neq j$라면 $ID_i \neq ID_j$로 한다. 정보 a를 복원하려면 k개 이상의 분산정보가 필요하다. 모아진 k개의 분산정보를 $f(ID_{j1}), f(ID_{j2}), ..., f(ID_{jk})$로 한다.

$$\begin{pmatrix} f(ID_{j1}) \\ f(ID_{j2}) \\ \vdots \\ f(ID_{jk}) \end{pmatrix} = \begin{pmatrix} 1 & ID_{j1} & \cdots & ID_{j1}^{k-1} \\ 1 & ID_{j2} & \cdots & ID_{j2}^{k-1} \\ & & \vdots & \\ 1 & ID_{jk} & \cdots & ID_{jk}^{k-1} \end{pmatrix} \begin{pmatrix} a \\ a_1 \\ \vdots \\ a_{k-1} \end{pmatrix} (mod p)$$

이 때, 연립 1차 방정식을 풀면 $a, a_1, a_2, ..., a_{k-1}$가 결정된다. 또한 구체적인 보간 공식은 다음과 같다.

$$a = \sum_{i=1}^{k} f(ID_{ji}) \prod_{\substack{1 \le r \le k \\ r \neq i}} \frac{ID_{jr}}{ID_{jr} - ID_{ji}}$$

$k-1$개 이하의 분산정보 밖에 없을 때는, 상기 연립 1차 방정식에 대해 행수가 $k-1$이하가 되므로 해는 부정이 된다. 따라서 a는 0으로부터 $p-1$을 모두 취할 수 있으므로, a의 정보는 전혀 얻을 수 없다.

(k,n) 임계치법은 일반적 구조를 가지는 비밀정보 분산기법으로 다음과 같은 방법으로 확장할 수 있다. 비밀분산방식을 구성하는 2개의 함수, 정보 분산함수 $Share$와 정보 복호함수 $Merge$로부터 비밀분산방식을 정의한

다. 또한 비밀정보를 s로 하며 n개의 분산 정보로 나누어 부호화했다. 따라서 k를 통해 비밀정보 s를 복원하는데 필요한 분산 정보의 임계치인 n을 분산 정보의 총수 $(s_1, s_2, \ldots, s_n)$을 분할된 n개의 분산정보로 할 수 있다. 비밀정보 s, 임계치 k, 분산정보의 총수 n의 정보를 분산함수 $Share$로 입력하여 $Share(s,k,n)$로부터 s의 복원에 필요한 임계치를 k로 설정하고 s에 대해 n개의 분산정보 $(s_1,s_2,\ldots,s_n)$을 계산하여 출력한다. n개 중에서 선택한 k개의 분산정보 $(s_1,s_2,\ldots,s_k)$로부터 비밀정보 s를 복원하는 함수를 정보 복호함수 $Merge$로 한다. 또한 함수 $Merge$는 k개의 분산정보 $(s_1,s_2,\ldots,s_k)$, 임계치 k, 분산정보의 총수 n을 복호함수 $Merge$로 입력하여 $Merge(s_1,s_2,\ldots,s_k),k,n)$은 비밀정보 s를 출력한다.

비밀분산기법은 다음과 같은 성질을 만족해야 한다.

첫째, 임의의 k개의 분산정보가 모이면 원래의 비밀정보 s를 복원할 수 있다. 둘째, 임의의 $k-1$개 이하의 분산 정보로부터 비밀정보 s를 복원할 수 없다. 또한, 데이터베이스에 저장되는 각 데이터는 효율적으로 검색을 실시할 수 있도록 각각 다른 키 정보를 가지고 있다. 각 데이터는 이하의 <Table 1>과 같이 대응되어 있다고 한다. 데이터 d_i에는 이름, 주소, 연령, 직업 등의 정보를 포함하고 있다고 한다. 데이터 d_i를 취득하기 위해서는 ID_i를 키로써 데이터베이스를 검색해서 d_i를 얻는다.

〈Table 1〉 Key corresponding information

Key	Data
ID_1	d_1
ID_2	d_2
ID_3	d_3
:	:
ID_i	d_i
:	:
ID_l	d_l

$k-out-of-nSSS+nDB_s$ 방식은 비밀분산기법을 사용하며 각 데이터를 데이터베이스에 분산정보로 분할하여, 이들을 다른 데이터베이스에 각각 저장한다. 데이터를 복원할 경우에는 각 데이터베이스로부터 복원하고자 하는 데이터의 분산정보를 취득하여 데이터의 복원에 필요한 수(임계치)의 분산정보를 추출하여 데이터를 복원한다.

$k-out-of-nSSS+nDB_s$ 방식은 다음과 같은 조건을 가정한다. 데이터의 수는 1개이며 데이터베이스의 수는 $n(DB_1,DB_2,\ldots,DB_n)$개 이다. 각 데이터의 분산된 정보의 수는 n개이며 복원에 필요한 임계치는 k개가 필요하다. 데이터를 분할하여 관리하기 위해서 초기화단계와 데이터 취득단계로 분류되어 구성한다.

초기화 단계는 각각의 데이터를 분산정보에 분할하여 서로 다른 데이터베이스에 저장한다. 데이터 d_i에 대해, d_i를 n개의 분산정보로 분할하여 해당정보를 서로 다른 n개의 데이터베이스에 분산하여 저장할 경우 다음과 같다. 데이터 d_i, 임계치 k, 분산정보의 합인 n을 정보 분산함수 $Share$의 입력으로 하여 d_i에 대해 n개의 분산정보$(d_i^1,d_i^2,\ldots,d_i^n) = Share(d_i,k,n)$을 얻는다. 얻을 수 있는 n개의 분산정보 $(d_i^1,d_i^2,\ldots,d_i^n)$을 서로 다른 n개의 데이터베이스에 저장한다. n개의 분산 정보는 다음과 같은 형태로 데이터베이스에 저장한다. n개의 데이터베이스에는 <Table 2>와 같이 각 데이터의 분산 정보가 저장된다.

〈Table 2〉 Distributed information stored in each database

DB	Distributed information
DB_1	$d_1^1, d_2^1, \ldots, d_l^1$
DB_2	$d_1^2, d_2^2, \ldots, d_l^2$
...	...
DB_n	$d_1^n, d_2^n, \ldots, d_l^n$

데이터 취득 단계는 사용자가 데이터 d_i를 획득하여 사용하고자 하는 단계이다. 사용자는 n개 데이터베이스로부터 k개의 데이터베이스를 선택한다. 선택된 k개의 데이터베이스로부터 데이터 d_i의 분산정보를 취득하여 정보 복호함수 $Merge$를 이용하여 데이터 d_i를 복원한다.

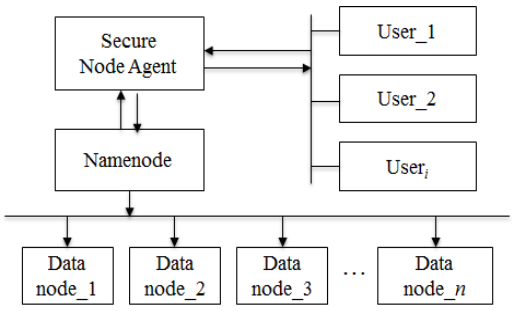
3. SNA기반의 분산 트랜스코딩 시스템

기존연구의 트랜스코딩 시스템은 트랜스코딩이 가능한 데이터를 FullHD급의 해상도 이하로 제한하여 트랜스코딩을 수행한다. FullHD급 이상의 해상도인 데이터를 트랜스코딩 하는데 있어서 단일 노드를 이용하는 트랜스코딩 시스템은 많은 한계점을 가지고 있다. 이러한 문제점을 해결하기 위해 분산 트랜스코딩 시스템을 이용하여 FullHD급 보다 높은 해상도인 4K급 해상도의 데이터를 트랜스코딩 하는 것이 가능하다[4].

분산 트랜스코딩 시스템은 Hadoop의 분산 파일시스템인 HDFS를 기반으로 분산 트랜스코딩을 수행한다. 따라서 콘텐츠 데이터를 저장할 때, 다수의 데이터 노드로 분할하여 분산 저장한다[5]. 따라서 데이터 자체가 분산 저장되기 때문에 특정 데이터 노드가 악의적인 공격자에 의해 공격당해 노출될 경우 부분적인 형태의 정보만이 유출된다. 하지만 악의적인 내부관리자의 경우 분산 저장된 블록의 위치를 모두 알 수 있기 때문에 저장된 콘텐츠를 취득할 수 있다는 문제점이 발생한다. 따라서 기밀성이 보장되지 않은 평문형태로 저장된 콘텐츠의 경우 내부 관리자에게 모든 내용이 노출된다.

기존기법의 분산 트랜스코딩 시스템은 데이터의 기밀성을 유지하기 위해 트랜스코딩 과정이 모두 완료 된 후 기밀성을 유지하기 위한 암호화를 수행하거나 트랜스코딩을 수행하기 이전의 데이터를 사전에 암호화하는 방법을 사용하였다. 하지만 SNA를 이용한 트랜스코딩 시스템은 트랜스코딩을 수행하는 과정에서 분산되는 데이터를 암호화하여 분산하기 때문에 기밀성을 유지한다. 또한 사용자의 요구에 따라 트랜스코딩을 즉시 실시하지 않는 경우에도 데이터 노드로 분산하는 과정에서 암호화를 수행하여 악의적 사용자에 의한 공격을 방지한다.

본 논문에서는 트랜스코딩 대상 데이터가 트랜스코딩 수행하기 위해 분산되는 과정에서 발생하는 보안문제를 해결하기 위해 [Fig. 1]과 같이 분산파일시스템인 HDFS 구조를 개선하여 SNA를 적용한 저장된 블록이 기밀성을 유지하고 악의적인 내부관리자의 공격을 방지하는 분산 트랜스코딩 시스템을 구성한다[6,7].



[Fig. 1] Distributed transcoding system structure based on SNA

3.1 SNA의 부호화 과정

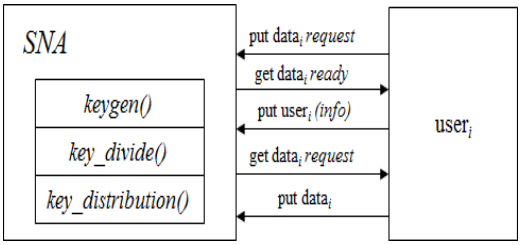
사용자는 초고해상도 콘텐츠를 트랜스코딩하기 위해 [Fig. 2]와 같이 SNA와 데이터를 전송하기 전 부호화 절차를 수행한다. SNA의 부・복호화 과정에서 수행되는 함수는 <Table 3>과 같다.

user$_i$는 data$_i$를 분산 트랜스코딩 시스템으로 전송하기 위해서 SNA에게 데이터 수신을 요청한다. SNA는 user$_i$의 데이터를 수신받기 전에 복호키를 생성하기 위해 user$_i$의 정보인 user$_i$*(info)*를 요청한다. user$_i$는 SNA로 사용자의 *ID, MAC_add, Content Context*를 전송한다.

〈Table 3〉 Function description of SNA

Function	Description
keygen( )	Generate encryption key user$_i$*(info)*
keychk( )	Generated Encryption, Decryption key to check
key_divide( )	Generated Encryption, Decryption key to divide
key_distribution( )	Encryption, Decryption key to Datanode for distribution
key_encrypt( )	Block encryption in Datanode
key_decrypt( )	Block decryption in Datanode

*Content Context*는 data$_i$의 제목, 재생시간, 사용코덱, 해상도, 비트 전송률, 프레임 속도, 오디오 채널정보, 오디오샘플 속도를 포함하고 있는 정보를 의미한다. SNA는 전송받은 user$_i$*(info)*를 *keygen*함수를 이용하여 복호키를 생성한다. 생성된 복호키는 *key_divide*함수를 이용하여 분산 트랜스코딩 시스템을 구성하고 있는 데이터 노드의 수로 분할하며 *key_distribution*함수를 통해 각 데이터 노드로 전송하기 위해 대기한다.



[Fig. 2] Encryption workflow of SNA

위와 같은 과정이 완료되면 SNA는 user$_i$에게 데이터를 전송하도록 데이터 노드의 논리주소를 네임노드로부터 전달받아 제공한다. user$_i$는 제공받은 데이터 노드의 논리주소를 통해 data$_i$를 각각의 데이터 노드로 순차적으로 전송한다. 수신이 완료된 데이터 노드는 HEVC로 data$_i$의 k_i 블록을 트랜스코딩 한 뒤 SNA로부터 전송받은 user$_i$의 data$_i$의 k_i 복호키를 수신하여 암호화한다. 따라서 data$_i$의 k_i 블록은 암호화되어 HDFS에 저장되어 있기 때문에 악의적인 내부관리자가 k_i 블록의 위치를 파악하고 있어도 블록의 정보는 알 수 없게 된다. 위의 과정은 <Table 4>와 같은 알고리즘에 따라 수행된다.

〈Table 4〉 SNA Encryption process

Algorithm SNA

```
1:  while true do
2:      get user_i(info)
3:      if(user_i(info)) == true )
4:          {keygen(user_i(info)) = usk
5:           key_divide(usk)
6:      if(key_divide( ) == true )
7:          {key_distribution( ) }
8:      else return -1
9:  }
10: else
11:     return -1

12: Datanode_n(usk, data_iki)
13: get usk, data_iki
14: if(usk == true)
15:     {key_encrypt(usk, data_iki)}
16: else return -1
```

3.2 SNA의 복호화 과정

user$_i$가 암호화되어 있는 data$_i$를 요청할 경우, [Fig. 4]와 같은 방법으로 복호화 과정을 실시한다. 데이터를 복호화하기 위해서는 data$_i$를 부호화하기 위해서 받은 정보 user$_i$*(info)*가 필요하다. 따라서 user$_i$는 부호화시 입력된 user$_i$*(info)*를 모두 입력하면 최초 저장된 형태의 데이터를 수신하는 것이 가능하다. 하지만 user$_i$*(info)*^{n}의 정보를 모두 알지 못하고 부분적으로 파악하고 있는 경우에는 data$_i$를 제한적으로 수신할 수 있으며 user$_i$*(info)*^{n}의 정보를 사용자가 사전에 설정해둔 임계치 값 t보다 낮게 입력될 경우 복호화 할 수 없다. 따라서 사용자는 얻고자하는 데이터를 복호화하기 위해서는 user$_i$*(info)*^{n} >= t가 요구된다.

사용자는 최초에 데이터를 저장하는 과정에서 임계치 값 t를 설정할 수 있으며 user$_i$*(info)*^{n}의 정보에 따라 복호화 결과를 다르게 설정할 수 있다. 따라서 t보다 크지만 user$_i$*(info)*보다 작을 경우, 최초 저장된 콘텐츠를 전송하지 않고 원본영상의 해상도보다 작은 저화질의 영상을 전송하여 user$_i$*(info)*에 따라서 화질의 차이를 보여 데이터의 기밀성을 유지하게 된다. 또한 사용자는 다양한 디바이스의 해상도에 따라 모든 정보를 입력하지 않고 부분적으로 입력하여 모든 데이터를 수신하지 않고 부분적으로 수신함으로써 저장된 데이터를 확인하는 것이 가능하다.

〈Table 5〉 SNA Decryption process

Algorithm SNA

```
1:  while true do
2:      request user_i(info)
3:      if(user_i(info)^n) >= t )
4:          {keychk(user_i(info)^n) = usk^n
5:           key_divide(usk^n)
6:      if(key_divide( ) == true )
7:          {key_distribution( ) }
8:      else return -1
9:  }
10: else
11:     return -1

12: Datanode_n(usk^n)
13:     get usk^n
14: if(keychk(usk^n == true))
15:     {key_decrypt(usk^n, data_iki)}
16: else return -1
```

<Table 5>와 같이 SNA는 $user_i$에게 수신된 정보 *keychk*함수를 통해 복호화 가능 여부를 확인하며 데이터를 복호화하는 것이 가능하면 *key_divide* 함수를 이용하여 키를 분할하며 *key_distribution* 함수를 통해 $data_i$를 저장하고 있는 모든 데이터 노드로 전송한다. 전송받은 데이터 노드는 *keychk* 함수를 통해 보유하고 있는 $data_i$ k_i 블록의 복호키인지 확인하고 맞을 경우 *key_decrypt* 함수를 이용하여 블록을 복호화 한다. 복호화과정이 완료되면 네임노드로 $user_i$로 전송할 준비가 완료되었다는 것을 알려주고 데이터 노드는 네임노드가 지정해준 주소를 통해 SNA를 거쳐 $user_i$에게 데이터를 전송한다. 데이터를 수신 받은 $user_i$는 HEVC 디코더를 이용하여 초고해상도 콘텐츠를 확인할 수 있다.

4. 결론

본 논문에서는 비밀분산기법을 이용한 분산 트랜스코딩 시스템에 대해 연구하였다. 초고해상도 콘텐츠를 트랜스코딩하기 위한 분산 트랜스코딩 시스템의 기밀성이 보장되지 않는다는 문제점을 해결하고자 SNA를 이용하여 데이터 노드에서 블록을 암호화하여 분산 저장함으로써 악의적인 내부관리자의 공격을 방지하는 것이 가능해졌다. 분산 저장된 블록을 부분적으로 취득하여 정보 공격이 가능한 만큼 모든 블록을 암호화하여 저장함으로써 안전한 분산 트랜스코딩이 가능하도록 했다. 하지만 기밀성 보장을 위해서 기존 HDFS구조에서 수행되는 과정보다 더 많은 과정이 추가적으로 발생하여 시스템에 미치는 영향이 존재하게 된다.

향후 연구과제로서 암호화 과정에서 소비되는 시간이 분산 트랜스코딩 시스템에 미치는 영향을 측정, 분석하여 시스템의 성능을 개선할 수 있는 방안을 제시해야 한다. 그리고 초고해상도 콘텐츠 등의 빅 데이터 처리에 효율적인 클라우드 환경에서 안전한 분산 트랜스코딩 구조 설계 등에 대해 연구하고자 한다. 또한, 사용자 상황의 동적인 변화에 따라 정보 분산함수, 정보 복호함수를 이용한 분산 트랜스코딩의 Fine -grained 복호화 과정에 대한 검토도 필요할 것이다.

REFERENCES

[1] Y. J. Song, “A Study on the secret sharing for managing a large quantity of data including individual information,” Korea Internet & Security Agency Research report, 2009.

[2] Kemal Ugur, Kenneth Andersson, Arild Fuldseth, Gisle Bjøntegaard, Lars Petter Endresen, Jani Lainema, Antti Hallapuro, Justin Ridge, Dmytro Rusanovskyy, Cixun Zhang, Andrey Norkin, Clinton Priddle, Thomas Rusert, Jonatan Samuelsson, Rickard Sjoberg, and Zhuangfei Wu, “High Performance, Low Complexity Video Coding and the Emerging HEVC Standard,” Proceeding of Circuits and Systems for Video Technology, pp. 1688-1697, 2010.

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

[4] S. M. Gu, A. Seo, Y. C. Kim, “A Study on Distributed Transcoding Using Dynamic Allocation of Virtual Machines in Cloud Computing Environment”, The 1’st Int. Conf. on Digital Policy & Management, The Society of Digital Policy & Management, pp. 125-126, 2013.

[5] Dongmahn S., Joahyoung L., Yoon K., Changyeol C., Hwangkyu C., Inbum J., “Load Distribution Strategies in Cluster-Based Transcoding Servers for Mobile Clients”, Computational Science and Its Applications(ICCSA), pp. 1156-1165, 2006

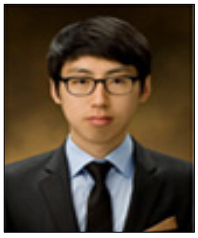
[6] Shvachko K., Kuang H., Radia S., and Chansler R., “The Hadoop Distributed File System,” in Proc of the IEEE 26th International conference on Mass Storage Systems and Technologies, pp. 1-10, 2010.

[7] S. M. Gu, Y. C. Kim, "Implementation and Performance Evaluation of Distribution Transcoding System based on Hadoop for Realistic Media Transmission," in The e-Business Studies vol 15, num 3, pp. 125-137, 2014.

송 유 진(Song, You-jin)

- 1982년 2월 : 한국항공대학교 공학사
- 1987년 8월 : 경북대학교 (공학석사)
- 1995년 8월 : 일본 Tokyo Institute of Technology (공학박사)
- 2003년 1월 ~ 2005년 1월 : Univ. of North Carolina at Charlotte 연구교수
- 1996년 3월 ~ 현재 : 동국대 경영학부 정보경영전공 교수
- 2006년 1월 ~ 현재 : 정보보호학회 부회장
- 2006년 1월 ~ 현재 : 국제e-비즈니스학회 이사
- 2006년 1월 ~ 현재 : 한국사이버테러정보전학회 이사
- 2011년 1월 ~ 현재 : 한국인터넷방송통신학회 이사
- 관심분야 : Privacy Protection, Secret Sharing, 클라우드 보안, 상황정보응용보안, 사물인터넷 UI/UX
- E-Mail : song@dongguk.ac.kr

구 석 모(Gu, Seokmo)



- 2013년 2월 : 동국대학교 컴퓨터공학과(공학석사)
- 2013년 3월 ~ 현재 : 동국대학교 테크노경영협동과정 박사과정
- 관심분야 : 클라우드 컴퓨팅, 분산컴퓨팅, 사물인터넷
- E-Mail : seokmogu@dongguk.edu

김 의 창(Kim, Yei-chang)

- 1983년 2월 : 동국대학교 수학과 (이학사)
- 1986년 8월 : 동국대학교 컴퓨터공학과(공학석사)
- 1993년 8월 : 동국대학교 컴퓨터공학과(공학박사)
- 1997년 1월 ~ 1998년 1월 : Univ. of Illinois(Post Doc.)
- 1991년 3월 ~ 현재 : 동국대 경영학부 정보경영전공 교수
- 2011년 1월 ~ 현재 : 한국인터넷전자상거래학회 상임이사
- 2013년 1월 ~ 현재 : 국제e-비즈니스학회 수석부회장
- 2014년 7월 ~ 현재 : 동국대학교 인재개발처장
- 관심분야 : 유비쿼터스응용, 사물인터넷, 모바일비즈니스
- E-Mail : kimyc@dongguk.ac.kr