http://dx.doi.org/10.14400/JDC.2016.14.2.175

프라이버시 보존형 소스기반 중복제거 방법

남승수, 서창호
공주대학교

Privacy Preserving Source Based Deduplicaton Method

Seung-Soo Nam, Chang-Ho Seo
Dept. of conversions Science, Konj-Ju National University

요 약 클라우드 스토리지에 민감한 데이터를 저장할 경우 평문상태의 데이터는 기밀성 문제가 발생하기 때문에 중복처리를 통한 스토리지 효율성 제공뿐만 아니라 데이터 암호화를 통한 기밀성 보장이 필요하다. 최근 네트워크 대역폭의 효율적인 사용을 위해 클라이언트 측 중복제거 기술이 주목을 받으면서 다양한 클라이언트 측 중복제거 기술들이 제안되었지만 아직까지 안전성에 대한 문제가 남아있다. 본 논문에서는 암호화를 통해 데이터의 기밀성을 보장하고 소유권 증명을 이용해 데이터 접근제어를 제공하여 신뢰할 수 없는 서버와 악의적인 사용자로부터 프라이버시를 보존할 수 있는 안전한 클라이언트 측 소스기반 중복제거 기술을 제안한다.

주제어 : 중복제거 기술, 개인 보호, 데이터 암호화, 대칭키 알고리즘, 공개키 알고리즘

Abstract Cloud storage servers do not detect duplication of conventionally encrypted data. To solve this problem, convergent encryption has been proposed. Recently, various client-side deduplication technology has been proposed. However, this propositions still cannot solve the security problem. In this paper, we suggest a secure source-based deduplication technology, which encrypt data to ensure the confidentiality of sensitive data and apply proofs of ownership protocol to control access to the data, from curious cloud server and malicious user.

Key Words : Deduplication, Privacy, Data encryption, Symmetric-key algorithm, Public-key algorithm

1. 서론

서로 다른 사용자가 같은 데이터를 아웃소싱 할 경우 스토리지 서버는 중복된 데이터를 단 한번만 저장함으로써 스토리지 효율성을 증대시킨다. 스토리지의 효율적인 사용을 위한 중복제거 기술은 수행 주체에 따라 서버측 타겟기반 중복제거 기술과 클라이언트측 소스기반 중복제거 기술로 구분된다.

◉ 서버측 타겟기반 중복제거: 서버는 클라이언트로부터 전체 데이터를 수신한 후 서버측에서 중복을 제거하는 방식이다. 프라이버시 침해가 적지만 데이터의 송·수신과 중복검사를 함께 수행하므로 네트워크 혼잡이 발생할 수 있다.

* 본 연구는 한국연구재단의 기초연구사업의 일환으로 수행하였음. [NRF-2013-R1A1A2010382]
Received 31 December 2015, Revised 15 January 2016
Accepted 20 February 2016, Published 28 February 2016
Corresponding Author: Changho Seo
(Konj-Ju National University)
Email: chseo@kongju.ac.kr

ISSN: 1738-1916

© The Society of Digital Policy & Management. All rights reserved. This is an open-access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (http://creativecommons.org/licenses/by-nc/3.0), which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

◉ 클라이언트측 소스기반 중복제거: 클라이언트가 직접 중복을 제거하는 방식으로써 네트워크 혼잡이 적고 대역폭의 효율적인 사용이 가능하다. 하지만 상대적으로 크기가 작은 Tag 값만으로 전체 데이터를 대체하는 것은 데이터 소유권에 대한 문제가 발생할 수 있다.

클라이언트 측 소스기반 중복제거 기술은 상대적으로 크기가 큰 전체 데이터 대신 데이터에 대한 해시 값만으로 중복을 감지하기 때문에 불필요하게 낭비되는 대역폭을 절약할 수 있다[1]. 하지만 클라이언트 측 소스기반 중복제거 기술은 상대적으로 크기가 작은 해시 값이 전체 데이터를 대신하기 때문에 해당 해시 값이 위협을 받는다면 전체 데이터에 대한 위협이 발생할 수 있다. 이에 따라 클라이언트 측 중복제거 기술은 데이터 소유권에 대한 증명이 필요하며 Halevi 등은 Merkle-Tree 기반의 소유권 증명방법인 PoW(proofs of ownership)를 제안했다[2, 3]. PoW는 데이터를 encoding(또는 universal hashing, streaming)한 후, 그 값을 이용해 Merkle-Tree [4]기반의 소유권 증명을 수행한다. 그러나 Halevi 등의 방법은 평문데이터에 대한 소유권 증명만을 제공하고 있어, 데이터의 기밀성을 보장하지 않는다. 따라서 암호화된 데이터에 대한 클라이언트측 소스기반 중복제거 기술이 필요하다.

최근 Kaaniche 등은 Convergent Encryption을 통한 데이터 암호화 및 Merkle-Tree 기반의 소유권 증명방법인 PoW가 결합된 클라이언트측 중복제거 기법을 제안했다 [5]. 이 기법은 암호화된 데이터를 기반으로 소유권 증명을 수행하기 때문에 데이터의 기밀성을 유지할 수 있다. 하지만 Kaaniche 등의 방법은 기존의 Convergent Encryption을 통하여 데이터를 암호화하기 때문에 여전히 전수조사공격인 사전공격에 대해 취약한 문제점을 가지고 있다.

본 논문에서는 이와 같은 약점을 보완하기 위하여 클라우드 스토리지로부터 독립적인 키 서버의 도움을 받아 데이터 암호화를 위한 키를 발급하며, 이러한 키를 이용해 암호화를 실행하고 Merkle-Tree를 구성한 후 PoW를 수행하는 클라이언트측 소스기반 중복제거 기술을 제안한다. 이는 현재까지 알려진 모든 취약점을 보완할 수 있는 프라이버시 보존형 소스기반 데이터 중복제거 기술이다.

2. 관련 연구

2.1 Convergent Encryption

Douceur 등 [6]에 의해 제안된 Convergent Encryption은 데이터 f를 해시한 값 $H(f)$를 키로 사용하며 대칭키 암호 알고리즘 E를 이용해 데이터 f를 암호화 한다.

$$C \leftarrow E(H(f), f) \tag{1}$$

따라서 같은 평문은 동일한 암호문이 되므로 중복데이터의 처리가 가능하다. 하지만 Convergent Encryption은 $H(f)$를 키로 사용하므로 전수조사 공격인 사전공격(dictionary attack)에 매우 취약하다.

2.2 DupLESS

Convergent Encryption의 취약점인 사전공격을 보완하기 위하여 Ballere 등은 키 서버를 도입한 DupLESS를 제안했다[7]. DupLESS는 키 서버를 통하여 클라이언트에게 키를 분배하고 클라이언트는 분배받은 키를 이용해 데이터를 암호화한다. 이때 키 서버는 클라이언트와의 통신에서 RSA-OPRF [8, 9]프로토콜을 이용해 클라이언트는 키 서버의 비밀정보를, 키 서버는 클라이언트의 비밀정보를 서로 알 수 없게 메시지로부터 유도되는 키를 분배하여 데이터의 중복 처리를 수행한다. 그러나 DupLESS는 키 서버와 공모하여 키 서버의 비밀 정보를 알면서 암호화된 데이터에 접근할 수 있는 공격자에 대해 엔트로피가 매우 낮은 데이터는 안전하지 않을 수 있다. 이 경우 DupLESS의 안전성은 Convergent Encryption의 안전성과 같아진다. 따라서 키 서버는 상당히 신뢰할 수 있는 제 3자(Semi-Trusted Third Party)로 가정하고 클라이언트로부터 요구되는 키 요청을 비율적으로 제한함으로써 안전한 데이터 아웃소싱을 가능하게 한다. 하지만 DupLESS는 서버측 타겟기반 중복제거 기술이므로 중복된 데이터 저장을 요청할 경우 대역폭의 낭비가 발생한다.

2.3 소유권 증명(Proofs of Ownership)

Halevi 에 의해 제안된 PoW(proofs of ownership)는 Merkle-Tree 기반의 소유권 증명 프로토콜이며 클라이

언트는 서버로부터 요구되는 무작위 색인에 대하여 올바른 sibling path를 제시해야 한다 [2, 10]. Halevi 등은 최소 엔트로피가 낮은 데이터에 대한 안전성 문제를 보완하기 위하여 서로 다른 3가지 버전의 PoW를 제안했다. 클라이언트가 데이터의 소유를 서버에게 증명할 수 있는 이 방식은, Merkle-Tree를 구성한 후 Tree의 최하위 node인 leaf node의 개수와 최상위 node인 root 값을 서버에게 전송한다. 서버는 무작위 leaf node의 색인을 클라이언트에게 전송하고 클라이언트는 해당 색인의 leaf node와 올바른 sibling path를 응답한다. 이때 서버는 클라이언트로부터 전송받은 값을 이용해 자체적인 root′ 값을 계산하고 만일 root′=root이면 해당 클라이언트에게 데이터 소유를 인정한다. 하지만 Halevi 등의 방식은 데이터를 암호화하지 않기 때문에 평문 데이터에 대한 소유권 증명을 수행한다.

2.4 클라이언트 측 소스기반 중복제거 기술

클라이언트측에서 직접 중복을 제거하는 방식으로써 데이터의 Tag 값을 계산하고 이를 외부 스토리지 서버로 전송한다. 이때 Tag 값이 이전에 스토리지에 저장된 데이터의 Tag 값이라면 클라이언트는 데이터 전체를 전송하지 않고 이미 저장되어있는 데이터에 대하여 해당 클라이언트의 정보인 메타데이터만을 전송하는 방식이다. 따라서 클라이언트측 소스기반 중복제거 기술은 외부 스토리지의 저장 공간뿐만 아니라 네트워크의 대역폭을 절약할 수 있다. 그러나 실제 데이터를 소유하지 않은 악의적인 사용자는 Tag 값만을 이용해 데이터의 소유를 주장하고 외부 스토리지에 저장되어있는 데이터를 다운로드 할 수 있다. 또한 클라이언트측에서 직접 중복제거를 수행하므로 클라이언트는 데이터 중복제거의 발생여부를 판단할 수 있다. 이는 외부 스토리지에 특정 데이터의 존재유무를 확인하여 온라인상으로 전수조사 공격을 가능하게 한다[11].

민감한 데이터에 대하여 위와 같은 문제를 보완하기 위해 Kaaniche 등 [3]은 최근 Convergent Encryption을 이용해 데이터를 암호화하고 암호화된 데이터를 이용해 Merkle-Tree를 구성한 후 소유권 증명을 수행하는 클라이언트측 소스기반 중복제거 기술을 제안하였다. 이 기법은 Convergent Encryption을 통해 비인가 사용자와 신뢰할 수 없는 스토리지로부터 데이터를 보호한다. 그러나 Kaaniche 등의 제안 기법은 기존의 Convergent Encryption을 통한 암호화 방식을 사용하므로 여전히 사전공격에 매우 취약할 수 있다. 따라서 본 논문에서는 이러한 취약점을 보완하기 위하여 DupLESS의 키 서버를 이용한 암호화 방식과 Merkle-Tree 기반의 소유권 증명을 결합한 안전한 클라이언트측 소스기반 중복제거 기술을 제안한다.

3. 프라이버시 보존형 소스기반 중복제거 기술

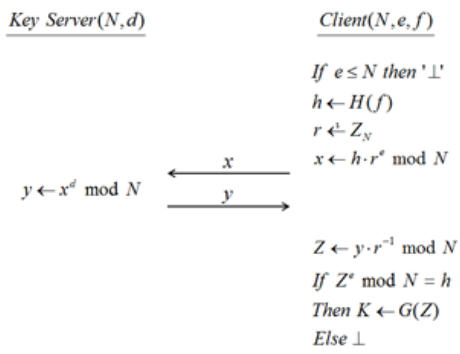
본 논문에서 제안하는 암호화된 데이터 중복제거 기술은 KS-클라이언트, CSS-클라이언트의 상호작용으로 이루어진다. 이때 해시함수 H는 충돌 저항성을 만족하고 대칭키 암호화 알고리즘 E는 안전한 의사난수 함수이다.

3.1 키 분배 프로토콜

KS는 [Fig. 1]과 같이 KS가 클라이언트에게 키를 분배해 주는 과정은 Bellare 등 에 의해 제안된 RSA-OPRF 프로토콜을 이용한다. 이 과정은 RSA-blind-signature의 성질을 이용해 클라이언트에게 데이터를 암호화할 키 K를 발급한다.

KS는 RSA 기반 공개키와 개인키 쌍을 생성한다. 즉, 큰 두 소수의 곱 $N=p \cdot q$ 와 $e \cdot d \equiv 1 \pmod{\varphi(N)}$ 을 계산하고, N과 e를 공개키로 사용하며, p, q 그리고 d를 개인키로 사용한다.

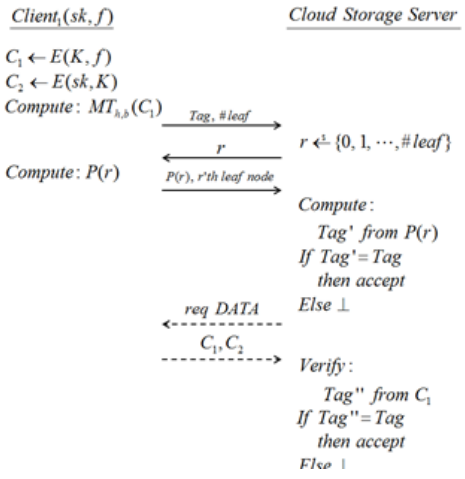
클라이언트는 Z_N에서 일정한 확률로 선택한 무작위 값 r에 KS의 공개키 e를 이용하여 r^e를 계산하고 자신의 데이터 f을 해시한 값 $h=H(f)$와 r^e을 곱한 값 $x \leftarrow h \cdot r^e \bmod N$을 KS에게 전송한다. 이때 만일 $e \leq N$이면 '$\perp$'한다. KS는 전송받은 x에 자신의 개인키 d를 이용하여 $y \leftarrow x^d \bmod N$을 계산한 후 클라이언트에게 전송한다. 클라이언트는 전송받은 y에 r^{-1}을 곱한 값 $z \leftarrow y \cdot r^{-1} \bmod N$을 계산한 후, 만일 $z^e \bmod N = h$ 이면 $K \leftarrow G(z)$를 데이터 f를 암호화할 키로 사용하고, $z^e \bmod N \neq h$이면 '$\perp$'한다. 여기서 함수 G는 충돌 저항성을 갖는 해시함수이다.



[Fig. 1] Key distribution protocol

3.2 업로드 프로토콜

새로운 데이터의 저장을 요청할 경우, 클라이언트는 먼저 KS에게 데이터 f를 암호화 할 키 K를 발급받은 후 [Fig. 2]와 같이 데이터를 암호화 한 값 $C_1 \leftarrow E(K,f)$과 자신의 개인키 sk로 키 K를 암호화 한 값 $C_2 \leftarrow E(sk,K)$를 계산한다. 이때 C_1을 이용하여 Merkle-Tree를 구성 한 후 CSS에게 leaf node의 개수와 Merkle-Tree의 root 값인 Tag 값($Tag = l_R$)를 전송한다. CSS는 클라이언트에게 leaf node의 무작위 색인을 보내고 클라이언트는 해당 색인의 leaf node와 sibling path $P(r)$로 응답한다. 이때 서버는 클라이언트에게 전송받은 leaf node와 $P(r)$을 이용해 자체적인 Tag' 값을 계산한 후 만일 $Tag' = Tag$이면($Tag' \neq Tag$이면 '⊥'한다.) 클라이언트에게 데이터에 대한 전송을 요청하고 클라이언트는 계산된 C_1과 C_2를 CSS에 전송한다. 이때 서버는 C_1에 대한 데이터 무결성을 검증하기 위하여, 클라이언트로부터 전송받은 C_1을 이용해 자체적으로 Merkle-Tree의 root 값인 Tag''을 계산한다. 만일 $Tag'' = Tag$이면($Tag'' \neq Tag$이면 '⊥'한다.) CSS는 최종적으로 클라이언트에게 C_1에 대한 소유권을 인정하고 C_1을 저장 공간에 저장하며 C_2와 Tag 값을 메타데이터로 저장한다.



[Fig. 2] $Client_1$ upload protocol

3.3 다운로드 프로토콜

데이터의 소유자가 클라우드 스토리지에 저장해 놓은 데이터에 대한 복구를 요청할 경우 클라이언트는 유지하고 있는 $MT_{h,b}(C_1)$에서 Tag 값인 l_R과 leaf node의 개수를 전송한다. CSS는 클라이언트에게 leaf node의 무작위 색인 r을 전송하고 클라이언트는 해당 색인의 leaf node와 sibling path $P(r)$로 응답한다. 이후 해당 클라이언트의 소유권이 인정되면 CSS는 클라이언트에게 해당 Tag에 대한 C_1과 해당 클라이언트에 대한 C_2(또는 C_2')를 전송한다. 이때 클라이언트는 자신의 개인키인 sk(또는 sk')로 C_2(또는 C_2')를 복호화하여 데이터를 암호화한 키 K를 복구하고 C_1을 복호화하여 데이터 f를 복원한다.

4. 안전성 분석

<Table 1>은 제안하는 기술과 타 소스기반 중복제거 제안 기법과의 비교표이다. 본 논문에서 제안하는 기법은 키 서버로부터 발급받은 키를 이용해 데이터를 암호화하기 때문에 키 서버가 안전성에 영향을 미치는 중요한 주체가 된다. 공격자 와 키 서버가 공모할 경우 공격자가 키 서버의 개인키인 d를 안다면 클라이언트의 암호화된 데이터는 전수조사 공격에 대하여 안전하지 않을 수 있다. 또한 키 서버는 정당한 클라이언트와 악의적인 목적을 가진 클라이언트를 구분 할 수 없기 때문에 키 서버는 악의적인 사용자의 키 발급 요청일지라도 정당하게 키를 발급한다. 이 경우, 엔트로피가 매우 낮은 데이터는 온라인 전수조사 공격에 위협을 받을 수 있다. 따라서 본

〈Table 1〉 Comparison of client-side source-based deduplication

	Convergent Encryption	Halevi etc.[6]	Kaaniche etc.[1]	Proposed method
Data encryption	○	×	○	○
Proof of ownership	×	○	○	○
Brute-force attack resilience	×	×	×	○

논문에서 제안하는 키 서버의 구성은 Bellare 등 에 의해 제안된 DupLESS의 키 서버와 동일한 조건을 가정한다.

4.1 키 서버로부터의 안전성

키 서버는 클라이언트에게 데이터를 암호화할 키를 분배한다. 키 서버가 악의적인 의도를 가지고 클라이언트의 정보를 위협할 경우라도 키 서버와 클라이언트는 RSA-OPRF 프로토콜을 이용하여 통신하기 때문에 클라이언트의 민감한 정보인 $H(f)$ 값은 무작위 값 r^e에 가려진 상태로 전송되어 키 서버는 클라이언트의 데이터 정보를 도출해 낼 수 없다. 따라서 키 서버는 자신이 클라이언트에게 분배한 키를 알 수 없다. 또한 서로 다른 클라이언트가 같은 파일에 대한 키 발급을 요청할 경우 즉, $H(f)$ 값이 같을 경우 무작위 값 r에 의해 KS는 키 발급이 요청되는 데이터의 동일함을 확인할 수 없다.

4.2 악의적인 사용자로부터의 안전성

클라이언트는 키 서버 및 클라우드 스토리지 서버와 통신을 한다. 이때 정당한 클라이언트가 악의적인 행동을 하는 경우 키 서버의 비밀 정보를 위협하고 CSS에 저장 되어있는 다른 사용자들의 데이터를 위협한다. 또한 악의적인 사용자는 파일 업로드시 정상적인 파일로부터 생성된 Merkle-Tree를 이용해 소유권 증명 프로토콜을 수행한 후, 데이터 전송 시 악의적인 파일을 업로드 하여 정상적인 데이터 저장을 위협할 수 있다.

첫 번째, 키 서버의 비밀 정보인 (p, q, d)를 위협할 경우 RSA-OPRF 성질에 의해 공격자는 인수분해 문제를 해결해야만 키 서버의 비밀 정보를 위협할 수 있다.

두 번째, CSS에 저장되어 있는 암호화된 데이터를 위협할 경우 악의적인 사용자는 데이터에 대한 소유권 증명을 수행해야 한다. 이때 CSS는 검증자, 클라이언트는 b-bit의 정보를 알고 있을 때 $t-b$가 충분히 크다면 공격자는 소유권 증명 프로토콜을 통과할 수 없다. 본 논문에서 제안하는 기법은 암호화된 데이터에 대한 Merkle-Tree를 생성하므로 공격자는 소유권 증명 프로토콜에서 무작위 색인에 대한 sibling path를 제시할 수 없다[10]. 또한 공격자는 평문데이터를 예측 하여 온라인 상으로 전수조사 공격을 시도할 수 있다. 이 경우 키 서버의 Rate Limiting에 의해 온라인 전수조사 공격에 대한 저항성을 가질 수 있다[8].

마지막으로, 악의적인 사용자는 데이터 C_1에 대한 소유권 증명을 정당하게 수행한 후, 데이터 업로드 과정에서 C_1'을 전송한다. 따라서 서버는 데이터 C_1'에 대한 메타데이터를 C_1에 대한 Tag 값으로 저장하게 된다. 이후 C_1을 업로드 하는 정당한 사용자는 데이터 다운로드 시 C_1'을 받게 된다. 이러한 공격을 보완하기 위해 서버는 전송받은 C_1에 대해 자체적인 Tag''을 계산하여 소유권 증명 시 전송된 Tag와 비교하여 전송된 데이터의 무결성을 검증한다. 따라서 악의적인 사용자는 소유권 증명을 통과하는 악의적인 데이터를 업로드 할 수 없다.

위와 같이 단일 주체가 공격자에게 공모된 경우 본 논문에서 제안하는 기법은 전수조사공격에 대한 저항성을 갖는다. 그러나 키 서버가 클라우드 스토리지 및 악의적인 사용자와 함께 공모된 경우, 즉, 키 서버의 비밀 정보인 d를 알고 암호화된 데이터에 접근할 수 있는 공격자는 전수조사 공격을 시도할 수 있다. 따라서 엔트로피가 낮은 데이터는 사전 공격에 취약할 수 있으며, 이 경우 Convergent Encryption을 통한 암호화 방식과 같은 안전성을 갖는다.

5. 결론 및 향후 연구 방향

본 논문에서는 최근 중복제거 기술 방법에 대하여 데이터 프라이버시를 보존하고 외부 스토리지의 저장 공간과 네트워크 대역폭을 절약할 수 있는 방법을 제안하였

다. 신뢰할 수 없는 외부 스토리지 서버로부터 안전하게 데이터의 프라이버시를 보존하고 비인가 사용자로부터 데이터의 접근을 차단할 수 있는 프라이버시 보존형 소스기반 중복제거의 구현을 가능하게 한다.

ACKNOWLEDGMENTS

This research was supported by the Basic Sciences Programs through the National Research Foundation of Korea (NRF), Republic of Korea (NRF-2013-R1A1A2010382).

REFERENCES

[1] R. Elbaz, D. Champagne, C. Gebotys, R. B. Lee, N. Potlapally, and L. Torres. "Hardware mechanisms for memory authentication: A survey of existing techniques and engines", *Transactions on Computational Science IV, LNCS*, pp. 1-22, 2009.

[2] S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg, "Proofs of ownership in remote storage systems", *in ACM conference on Computer and communications security*, pp. 491-500, Chicago, USA, OCT 2011.

[3] D. B. Kevin, J. Ari, O. Alina, "HAIL: a high-availability and integrity layer for cloud storage", *Proceedings of the 16th ACM Congerence on CCS'09*, pp. 187-198, 2009.

[4] S. Michael, "Merkle Tree Traversal in Log Space and Time", *EUROCRYPT 2004. LNCS*, Vol. 3027, pp. 541-554, 2004.

[5] N. Kaaniche, M. Laurent, "A Secure Client Side Deduplication Scheme in Cloud Storage Environments", *New Technologies, Mobility and Security (NTMS)*, 2014.

[6] J. R. Douceur, A. Adya, W. J. Bolosky, P. Simon, "Reclaiming space from duplicate files in a serverless distributed file system", *IEEE Distributed Computing Systems*, pp. 617-624, 2002.

[7] M. Bellare, S. Keelveedhi, T. Ristenpart, "DupLESS: server-aided encryption for deduplicated storage", *22nd USENIX conference on Security*, pp. 179-194, 2013.

[8] J. Camenisch, G. Neven, A. Shelat, "Simulatabel Adaptive Oblivious Transfer", *EUPROCRYPT 2007 LNCS*, Vol. 4515, pp. 573-590, 2007.

[9] N. Moni, R. Omer, "Number-theoretic constructions of efficient pseudo-random functions", *Journal of the ACM(JACM)*, Vol. 51, pp. 231-262, 2004.

[10] Ralph. C. Merkle, "A Certified Digital Signature", *CRYPTO'89 Proceedings LNCS*, Vol. 435, pp. 218-238. 1990.

[11] H. Danny, P. Benny, S. P. Alexandra, "Side channels in clude services, the case of deduplication in clude storage", *Security & Privacy, IEEE*, pp. 40-47, 2010.

[12] NIST, "NIST SP 800-17 : MOVS", 1998.

[13] NIST, "Derived Test Requirements for FIPS PUB 140-1, Security Requirements for Cryptographic Modules", 2001.

[14] NIST, "NIST SP 800-20 : TMOVS", 2000.

[15] NIST, "NIST SP 800-21 : AES", 1999. 11.

[16] NIST, "Advanced Eneryption Standard(AES)", FIPS PUB 197, 2001. 11.

[17] R.L. Rivest, A. Shamir, and L.M. Adleman, A method for obtaining digital ignatures and public-key cryptosystems, Communications of the ACM, volume 21, pp.120-126, February 1978.

[18] Seong-Hoon Lee, "Actual Cases and Analysis of IT Convergence for Green IT", Journal of the Korea Convergence Society, Vol. 6, No. 6, pp. 147-152, 2015.

[19] Jin-Keun Hong, "Threat Issues of Intelligent Transport System in the V2X Convergence Service Envrionment", Journal of the Korea Convergence Society, Vol. 6, No. 5, pp. 33-38, 2015.

남 승 수(Nam, Seung Soo)

- 2012년 2월 : 공주대학교 응용수학과(이학사)
- 2014년 8월 : 공주대학교 융합과학과(공학석사)
- 2014년 8월 ~ 현재 : 공주대학교 융합과학과 박사과정중
- 관심분야 : 영상처리, 정보보호, 네트워크 보안
- E-Mail : tnfok815@naver.com

서 창 호(Seo, Chang Ho)

- 1990년 2월 : 고려대학교 수학과 졸업(학사)
- 1992년 2월 : 고려대학교 수학과(석사)
- 1996년 8월 : 고려대학교 수학과(박사)
- 1996년 8월 ~ 2000년 2월 : 한국전자통신연구원 선임연구원, 팀장
- 2000년 3월 ~ 현재 : 공주대학교 응용수학과 교수
- 관심분야 : 암호알고리즘, PKI, 무선 인터넷 보안 등
- E-Mail : chseo@kongju.ac.kr