맵리듀스를 적용한 동형 암호화 기법 연구

김영기O, 홍충선*
경희대학교 컴퓨터공학과
{qoo0144, cshong}@khu.ac.kr

A Study on Homomorphic Encryption Scheme using MapReduce

Youngki KimO, ChoongSeon Hong*
Department of Computer Science and Engineering, Kyung Hee University

요 약

클라우드 컴퓨팅 기술이 발전함에 따라 기업뿐만 아니라 기관 또는 개인 사용자들도 더욱 늘어나고 있다. 클라우드 컴퓨팅 기술은 편의성, 확장성, 접근성 등의 장점을 이유로 더욱 각광받는 것으로 여겨진다. 그러나 이런 다양한 이점들에도 불구하고, 클라우드 컴퓨팅 기술이 널리 사용되기 위해서는 보안상의 문제점을 해결해야 한다는 의견이 지배적이다. 특히 최근에는 클라우드 스토리지에 개인정보를 저장하고 이를 활용하는 사용자들이 증가하면서 클라우드 환경에서의 보안이슈는 더욱 중요시되고 있다. 이를 극복하기 위해 암호화된 데이터에 대해 복호화를 수행하지 않고도 연산을 수행할 수 있는 동형 암호화 기법이 제안되었다. 그러나 기존의 기법은 암호화에 필요한 키의 크기가 크고 연산의 복잡도가 높아 실제로 적용하기에 어려움이 있다. 따라서 본 논문에서는 AES 알고리즘에 맵리듀스를 적용하여 동형 암호화 과정에 필요한 시간을 최소화하는 기법을 제안한다.

1. 서 론

클라우드 컴퓨팅 기술이 발전함에 따라 기업뿐만 아니라 기관 또는 개인 사용자들도 더욱 늘어나고 있다. 클라우드 컴퓨팅 기술은 편의성, 확장성, 접근성 등의 장점을 이유로 더욱 각광받는 것으로 여겨진다. 실제로 여러 기업들은 그들이 제공하는 서비스와 관련된 데이터에 클라우드 환경을 활용함으로써 상당한 경제적 효과를 누리고 있다.

그러나 이런 다양한 이점들에도 불구하고, 클라우드 컴퓨팅 기술이 널리 사용되기 위해서는 보안상의 문제점을 해결해야 한다는 의견이 지배적이다. 특히 최근에는 클라우드 스토리지에 개인정보를 저장하고 이를 활용하는 사용자들이 증가하면서 클라우드 환경에서의 보안이슈는 더욱 중요시되고 있다.

이런 문제점을 극복하기 위한 연구로 2009년 Gentry는 암호화된 데이터에 대해 복호화를 수행하지 않고도 연산을 수행할 수 있는 동형 암호화(homomorphic encryption) 기법을 제안하였다[1]. 그러나 동형 암호화를 수행하기 위해 필요한 키의 크기가 크고, 연산의 복잡도가 높아 암호화 과정에 수반되는 시간이 매우 길어 현실적으로 적용하기 어렵다.

따라서 본 논문에서는 AES 알고리즘(Advanced Encryption Standard)에 Hadoop에서 제공하는 병렬 프로그래밍 모델인 맵리듀스(MapReduce)를 활용하여 동형 암호화 과정에 필요한 시간을 최소화하는 것을 목적으로 한다.

2장에서는 보안 기법으로 널리 사용되고 있는 AES 알고리즘, 동형 암호화, 맵리듀스에 대해 언급하고, 3장에서는 기존 연구의 문제점 및 제안사항에 대해 기술한다. 4장에서는 실험 결과 및 성능평가를 분석하고, 마지막으로 5장에서는 본 논문의 결론 및 향후 연구계획에 대해 언급한다.

이 논문은 2016년도 정부(미래창조과학부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임 (R0126-16-1009, ICBMS 플랫폼 간 정보 모델 연동 및 서비스 매쉬업을 위한 스마트 중재 기술 개발).
*Dr. CS Hong is the corresponding author

2. 관련 연구

2.1 AES 알고리즘

AES 알고리즘은 2001년 미국 표준 기술 연구소(NIST)에 의해 제정된 암호화 방식이다[2]. AES 알고리즘은 암호화 및 복호화에 같은 키가 사용되는 대칭 키 암호화 기법이다. 사용되는 키의 크기는 128, 192, 256 비트이며, 블록 암호 방식으로 암호화 및 복호화가 수행된다.

2.2 동형 암호화

동형 암호화 기법은 앞서 언급한 클라우드 컴퓨팅 환경의 데이터를 보호하기 위한 기법으로, 암호화된 데이터에 대해 복호화하지 않고도 연산을 수행할 수 있는 암호화기법이다[3]. 그림 1은 동형 암호화기법의 과정을 설명하는 그림이다.

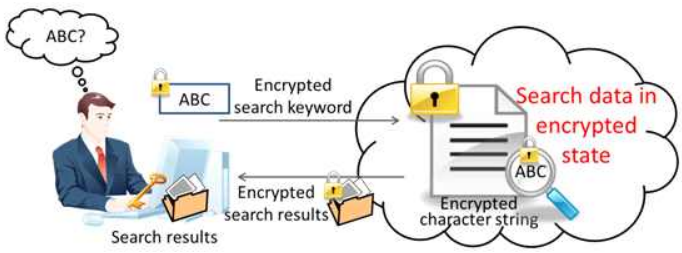


그림 1. 동형 암호화기법

2009년 Gentry에 의해서 처음으로 완전 동형 암호화기법이 제안되면서 그 후 모든 동형 암호화기법에 대한 연구는 Gentry가 제시한 기본 틀을 바탕으로 효율성을 개선하는 것으로 초점이 맞춰져 있다.

2.3 맵리듀스

맵리듀스는 구글에서 대용량 데이터 처리를 분산 병렬 컴퓨팅에서 처리하기 위한 목적으로 제작한 소프트웨어 프레임워크이며, 현재는 아파치에서 제공하는 클라우드 컴퓨팅 환경인 하둡에서 분산 처리를 위한 모델로 사용되고 있다.

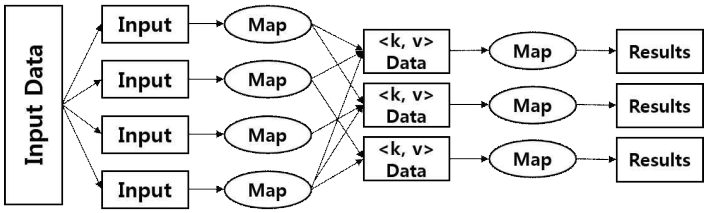


그림 2. 맵리듀스 워크플로우

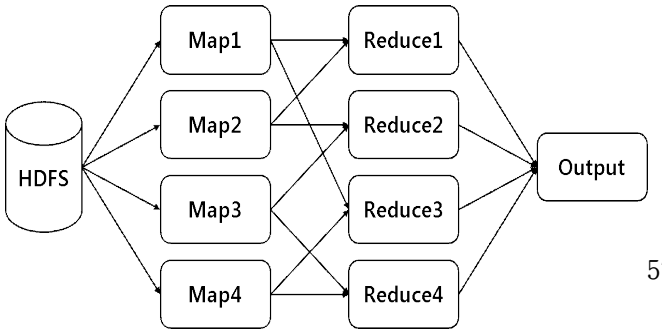
그림 2는 맵리듀스의 워크플로우를 나타낸다. 그림에서 확인할 수 있듯이, 맵리듀스 프로그래밍 모델에는 두 개의 주요 함수가 존재한다. 맵 함수는 효율적인 처리를 위해 데이터를 입력받아 키-값 형태의 쌍으로 변형하고, 연관성이 있는 데이터들을 그룹화하여 리듀스 함수에 전달한다. 리듀스 함수는 입력받은 데이터를 키 값을 기준으로 정렬하고 실질적인 처리를 수행한다.

3. 기존 연구의 문제점 및 제안사항

앞서 언급한 바와 같이 동형 암호화 기법에는 암호화 과정에 필요한 키의 크기가 크고 연산의 복잡도가 높아 시간이 오래 걸린다는 문제점이 존재한다. 이를 해결하기 위한 방법으로 [4]에서는 기존 동형 암호화 기법에 병렬처리를 적용한 바 있다. [4]에서 제안한 기법은 전처리 과정에서 동형 암호화에 필요한 연산들의 의존성을 확인하여 세분화하고 그래프 형태로 구성한다. 그리고 이를 여러 개의 프로세서를 활용하여 암호화에 필요한 실행시간을 단축시켰다.

그러나 제안된 기존의 방법에서는 이미 동형 암호화를 위한 조건을 만족하도록 암호화가 수행된 상태에서 각각의 데이터들에 대한 기본적인 덧셈 및 곱셈을 수행하고 그 결과를 확인했다. 따라서 동형 암호화를 위한 키를 생성하고 데이터들을 암호화하는 과정 자체에 대한 검증이 불가하다. 또한 저자가 제약조건들을 사전에 정의한 상태에서 연구를 수행했기 때문에 그 확장성 또한 제한적이라고 판단할 수 있다.

본 논문에서는 동형 암호화에 사용되는 키를 생성 및 암호화 과정을 포함하는 전자 코드북 모드의 AES 알고리즘을 병렬처리 프로그래밍 모델인 맵리듀스에 적용하여 동형 암호화 수행에 필요한 시간을 최소화하는 기법을 제안한다.



57

그림 3. 맵리듀스를 적용한 동형 암호화 기법

그림 3은 본 논문에서 제안하고자 하는 맵리듀스를 적용한 동형 암호화 기법이다. 하둡 분산 파일 시스템은 블록 타입 기반의 저장소로 평문을 블록 타입으로 나누어 저장하고 있다.

각각의 맵 함수는 하둡 분산 파일 시스템으로부터 (블록 번호-데이터) 형태로 평문을 입력받고 동형 암호화를 수행하여 (블록 번호-암호화된 데이터)의 형태로 변환하여 이를 리듀스 함수로 전달한다. 리듀스 함수는 (블록 번호-암호화된 데이터) 형태로 나누어진 암호문을 입력받아 블록 번호를 기준으로 정렬하여 하나의 완전한 암호문 형태로 병합한다. 또한 병합과정에 필요한 시간을 단축하기 위해 맵 함수에서 암호화를 수행하고 해당되는 데이터의 블록 번호를 고려하여 리듀스 함수에서 인접한 블록 번호를 갖도록 전달한다.

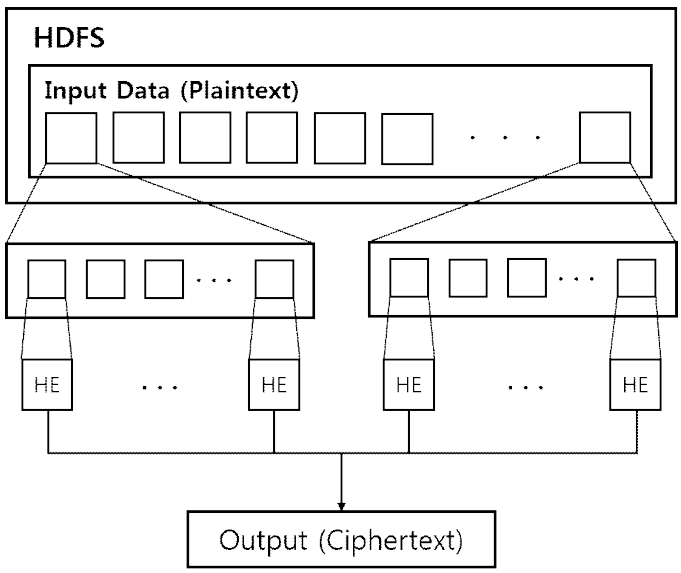


그림 4. 제안하는 암호화 기법의 데이터 처리 과정

그림 4는 제안하는 동형 암호화 기법에서 데이터가 어떤 과정으로 처리되는지를 나타낸다. 블록 단위로 저장되어있는 데이터를 더 작은 단위로 나누어 병렬적으로 암호화를 수행한다.

4. 성능 평가

표 1. 실험 환경

Operating System	Ubuntu 14.04 LTS
Num. of Processor	2
RAM	4GB
Hadoop version	Apache-Hadoop 2.7.2
Input Data	5GB
Block Size	128MB
Num. of Map functions	4
Num. of Reduce functions	4

표 1은 본 논문에서 성능 평가를 위해 구축한 실험 환경이다. 병렬처리 환경을 구축하기 위해서 2개의 프로세서를 동형 암호화 연산에 사용하였다. 평문은 128MB 단위로 나누어 블록 암호화에 사용하였으며, 맵-리듀스 함수는 각각 4개를 활용하였다. 또한 본 논문에서 제안하는 동형 암호화는 오픈소스 라이브러리인 HElib[5]를 기반으로 구현되었다.

표 2. 키 생성 매개변수

Parameter	Value	Description
λ	72	Security parameter
μ	140.034	Hardness parameter
s	15	Sparse sub-set size
S	512	Big set size
p	4	Number of bits of precision
t	384	Bit size of coefficients for γ
n	25	Lattice dimension
R	22	Ratio of elements in the big set

표 2는 동형 암호화의 키 생성 과정에 필요한 매개변수이며, 각각의 값과 설명들을 언급하였다. 또한 정확한 성능 분석을 위해 기존의 동형 암호화 기법과 제안하는 기법에 동일한 매개변수를 적용한다.

그림 5는 전자 코드북 모드의 AES 알고리즘을 적용한 동형 암호화 기법과 제안하는 기법의 키 생성과 암호화 과정에 소요되는 시간을 비교한 것이다. 키 생성에 필요한 매개변수 값들은 표 2에서 언급한 바 있으며 기존의 기법과 제안하는 기법에 동일하게 적용하였다. 전자 코드북 모드의 AES 알고리즘은 128비트를 사용하였다. 5GB의 평문을 입력 데이터로 사용하여 키 생성과 동형 암호화 과정에 소요되는 시간을 측정하여 비교해 보았을 때, 제안하는 기법이 기존의 기법보다 시간이 덜 걸리는 것을 확인할 수 있었다.

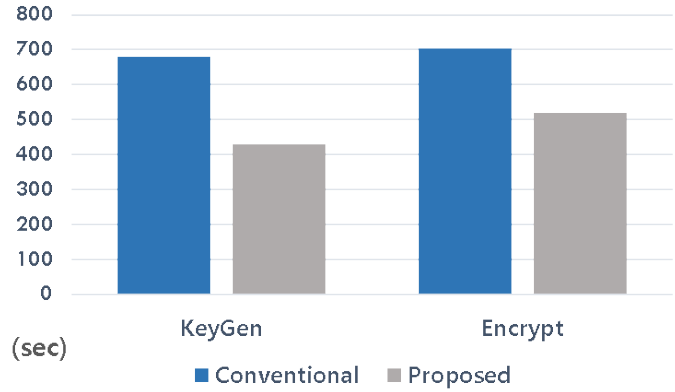


그림 5. 키 생성 및 암호화 수행에 필요한 시간 비교

5. 결론 및 향후 연구계획

본 논문에서는 기존 동형 암호화 기법의 암호화에 필요한 키의 크기가 크며 연산의 복잡도가 높아 실행 시간이 오래 걸린다는 기존 연구의 문제점을 해결하기 위해서, 하둡에서 제공하는 병렬처리 프로그래밍 모델인 맵리듀스를 전자 코드북 모드의 AES 알고리즘에 적용하였다. 4장의 성능 평가에서 살펴보았듯이, 제안하는 동형 암호화 기법은 기존의 암호화 기법보다 소요되는 시간의 측면에서 더욱 우수한 결과를 얻을 수 있었다. 향후에는 완전 동형 암호화를 위해 제안하는 구조에 부트스트랩핑(bootstrapping)과 스쿼싱(squashing)방식을 추가하는 연구를 수행할 것으로 예상된다.

참 고 문 헌

[1] Craig Gentry, "Fully homomorphic encryption using ideal lattices," ACM Symposium on Theory of Computing, pp.169-178, May 2009.

[2] Daemen, Joan, Vincent, Rijmen, "The design of Rijndael: AES - The Advanced Encryption Standard," Springer Science & Business Media, 2013.

[3] Craig Gentry, Shai Halevi, "Implementing Gentry's Fully-Homomorphic Encryption Sheme," Annual International Conference on the Theory and Applications of Cryptographic Techniques, pp.129-148, May 2011.

[4] Hayward, Ryan, Chia-Chu Chiang, "Parallelizing fully homomorphic encryption for a cloud environment," Journal of applied research and technology, Vol. 13, no. 2, pp.245-252, April 2015.

[5] HElib, [Online], Available: https://github.com/shaih/HElib/