클라우드 저장소에 민감 데이터 보안 강화를 위한 암호화 알고리즘 연구

주형진*, 김대훈*, 최상현*, 민연아O, 백영태**
O가천대학교 소프트웨어학과,
*가천대학교 소프트웨어학과,
**김포대학교 멀티미디과
e-mail: hjjooace24@gmail.com*, gachonsw19csh@gmail.com*, anonkroea4869@gmail.com*,
yah0612@naver.comO, hannae@kimpo.ac.kr**

A Study on Encryption Algorithm for Sensitive Data Security in Cloud Storage

Hyung-Jin Joo*, Dae-hun Kim*, Sang-hyun Choi*, Youn-A MinO, Yeong-tae Baek**
ODept. of Software, Gachon University,
*Dept. of Software, Gachon University,
**Dept. of Multimedia Kimpo University

● 요 약 ●

본 논문에서는 클라우드 저장소의 민감한 데이터를 보호하기 위해 제시된 암호화 알고리즘을 이용하여 텍스트 데이터를 암호화하고 처리 속도에 대한 성능을 측정하여 기존의 방식과 비교 ·분석하였다. 클라우드 데이터는 사용자 로그인 정보 탈취나 SSL Strip 공격에 취약하기 때문에 이러한 보안 사고의 피해를 최소화하기 위해 데이터 암호화를 통한 데이터의 보안이 요구된다. 본 논문에서는 클라우드 전송을 위해 구글 드라이브 API를 연동했으며, 암호화 알고리즘을 텍스트에 적용하기 위해 Python 언어를 이용하여 데이터를 암호화하고 구글 드라이브에 전송하는 테스트 프로그램을 제작하여 프로젝트를 진행하였다.

키워드: 클라우드 저장소(Cloud Storage), 암호화(Encryption), 보안(Security), 시뮬레이션(Simulation)

I. Introduction

클라우드는 가상 컴퓨팅, 공유 저장소, DB, 콘텐츠 플랫폼과 함께 인공지능을 통한 빅데이터 분석 등의 솔루션을 제공하면서 기업 생산성의 혁신을 이끌기 때문에 많은 기업이 클라우드 서비스를 이용하고 있다[1]. Gartner에 따르면 전 세계 퍼블릭 클라우드 시장 규모가 2018년 21% 성장했으며 2019년에는 17.3% 성장한 2,062억 달러에 이를 것으로 전망했다[2]. 이처럼 클라우드가 활성화되면서 보안의 중요성이 다시 한 번 대두된다. Cloud Security Alliance에서 2019년 발표한 보고서 “Top Threats to Cloud Computing The Egregious 11”에서 계정 탈취 위협을 5위, 악의적인 내부자의 위협을 6위로 꼽았다[3]. 실제로 2014년 Code Spaces사의 AWS 계정이 탈취되어 서비스를 계속 이어갈 수 없는 상황이 되자 Code Spaces가 사업을 접은 사례가 있었고, 2018년 Tesla에서 내부 직원이 대용량의 기밀 데이터를 유출한 사례가 있었다[4]. 사용자의 클라우드 계정이 탈취되면 클라우드 플랫폼 자체에서 데이터를 암호화하더라도 공격자에게 데이터가 그대로 노출되어 2차 피해로 이어질 수 있다.

이를 위한 해결법 중 하나는 클라우드에 데이터를 업로드 할 때부터 클라이언트에서 정보를 암호화하여 올리는 것이다. 해당 방식을 이용하면 원본 데이터가 암호화되어있기 때문에 공격자가 계정을 탈취하더라도 데이터를 열람할 수 없다. 하지만 이는 서비스 속도 저하의 원인이 될 수 있으므로 클라우드 저장소에 효율적인 암호화 알고리즘을 선택해야 한다.

II. Experiments

본 논문에서는 클라우드 저장소에 데이터를 업로드하는 과정에서 데이터를 암호화하고 속도를 비교하여 클라우드 저장소에 사용하기 효율적인 암호화 알고리즘을 제시하고자 한다.

연구에 사용된 알고리즘은 클라우드 저장소에 데이터 전송을 위해 빠른 속도를 필요로 하기 때문에 대칭키 알고리즘 중 AES, RC4, 3DES를 지정하여 연구를 진행하였다.

3DES는 DES의 보안성이 약해지면서 대안된 알고리즘으로 56bit key를 사용한다. Triple-DES라고도 하며 DES를 3번 반복하는 구조이다.

AES는 보안성이 약해진 DES를 대체하기 위한 대칭 블록화 알고리즘이며 4행 4열 바이트 행렬인 128비트의 데이터 블록으로 분할하여 각 블록에 대해 반복 실행함으로써 암호화를 하는 방식이다[5].

RC4는 바이트 단위의 암·복호화를 위한 스트림 알고리즘으로 고속 소프트웨어 구현을 목적으로 개발되었다[6].

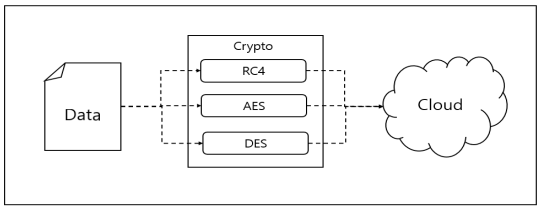

Fig. 1. The Process of Stream alogorithm

암호화 알고리즘은 모두 Python의 pycrypto 모듈을 이용하여 구현하였으며, fig. 1과 같이 AES(mode CBC), RC4, 3DES(mode CBC)를 사용하였다. 각 알고리즘은 모두 같은 key를 사용하였다. 암호화 대상 파일은 10MB~15MB이며 100개의 데이터 셋을 두었다. 파일 업로드는 Google Drive API를 사용하여 연구를 진행하였다. 연구에 사용된 컴퓨터의 성능은 table 1과 같다.

Table 1. Performance of the computer

OS	Window 10
CPU	intel(R) Core(TM) i7-8656U CPU @ 1.80GHz
GPU	NVDIA GeForce GTX 1650 with Max-Q Design
RAM	8.0GB

제시된 3개의 알고리즘으로 각각 데이터를 암호화하고 클라우드 저장소에 업로드 하는 과정을 시간 측정하여 비교 분석했다.

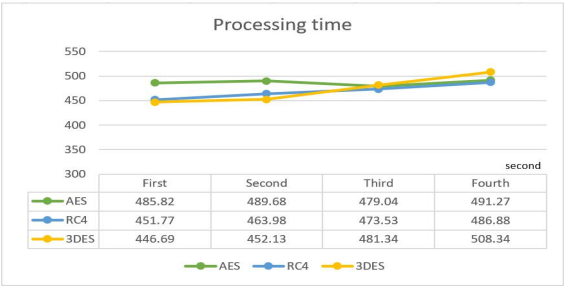

Fig. 2 Comparative analysis of measurement results

각 4번의 비교 측정 결과는 fig. 2와 같다. RC4는 평균 469.04초로 가장 빠른 처리 시간을 보여주었다. 그 다음은 3DES로 평균472.125초였으며 AES는 평균 486.453초로 측정되었다.

III. Conclusion

본 논문에서는 클라우드 저장소의 데이터 보안을 강화하기 위해 AES, RC4, 3DES의 알고리즘을 비교 분석하였다. 클라우드 저장소에 암호화되지 않은 데이터는 사용자의 클라우드 계정이 탈취되었을 경우 원본 데이터가 공격자에게 그대로 노출된다. 또한 클라우드 자체에서 데이터를 암호화하더라도 클라우드 서비스 시스템 자체의 취약점을 통한 공격이나 서비스 제공자의 내부 유출 위협 등이 존재한다. 이를 해결하기 위해 민감 데이터 자체의 암호화가 필요하다. 본 연구에서 진행한 결과는 RC4가 평균적으로 가장 빠른 처리 시간을 보여주었고 그 다음은 3DES, AES 순으로 나타났다. 네트워크 환경에 의한 속도차이가 존재할 수 있으나 평균적으로 나온 수치를 이용해 RC4로 암호화한 데이터를 클라우드 저장소에 전송하는 것이 효율적이다.

향후 민감정보의 보안 뿐 아니라, 처리속도를 고려한 보다 효과적 알고리즘에 대한 연구가 필요하다.

REFERENCES

[1] SHJo · GTHan, “A Partial Encryption Method for the Efficiency and the Security Enhancement of Massive Data Transmission in the Cloud Environment”, Korea Information Processing Society, August 7, 2017

[2] “Gartner Forecasts Worldwide Public Cloud Revenue to Grow 17.3 Percent in 2019”,

[3] “Top Threats to Cloud Computing The Egregious 11”, Cloud Security Alliance, 2019

[4] Ibid.

[5] JYOh, JHSeo, “Experimental Analysis of AES Cryptographic Algorithms”, Korea Information Electron Communication Technology, p.2, June. 2010

[6] HHKim, SBLee, SILee, CSOh, “Mobile DB Encryption Technique Using Lightweight RC4 Algorithm”, Korea Entertainment Industry Association, March. 2012