KIPS Trans. Comp. and Comm. Sys.
Vol.9, No.12 pp.307~314
ISSN: 2287-5891 (Print), ISSN 2734-049X (Online)

https://doi.org/10.3745/KTCCS.2020.9.12.307

A Video Encryption Based Approach for Privacy Protection of Video Surveillance Service

Jeongseok Kim† · Jaeho Lee††

ABSTRACT

The video surveillance service is being widely deployed around our lives and the service stores sensitive data such as video streams in the cloud over the Internet or the centralized data store in an on-premise environment. The main concerning of these services is that the user should trust the service provider how secure the video or data is stored and handled without any concrete evidence. In this paper, we proposed the approach to protecting video by PKI (public key infrastructure) with a blockchain network. The video is encrypted by a symmetric key, then the key is shared through a blockchain network with taking advantage of the PKI mechanism. Therefore, the user can ensure the sensitive data is always kept secure and traceable in its lifecycle.

Keywords : Blockchain, Privacy Protection, Video Surveillance

개인정보보호를 위한 영상 암호화 아키텍처 연구

김 정 석† · 이 재 호††

요 약

영상 감시 시스템은 광범위한 영역에서 쉽게 설치되고 녹화 장치 혹은 인터넷을 통한 클라우드 저장소에 영상 정보를 관리하는 중앙 관리 방식을 사용하고 있다. 이러한 시스템의 주요한 문제점은 저장 영상의 전송 과정과 저장 대해서 객관적으로 신뢰할 수 있는 방법이 제공되지 않고 있으며, 개인정보보호를 위한 장치 유무와 별개로 모든 권한을 서비스 제공자에게 위임한 상태에서 운영하고 있다는 점이다. 본 연구에서는 공개키 기반 암호화와 블록체인 기반의 키 관리 시스템을 조합한 아키텍처를 이용하여 민감한 정보를 사용자가 안전하게 보호할 수 있는 방안을 제시한다. 제안하는 아키텍처에서는 대칭키를 사용한 블록 암호화(block-cipher) 과정을 통해 영상 정보를 암호화하고, 이때 사용하는 대칭키를 사용자의 공개키로 암호화하여 블록체인의 레저(ledger)로 기록하는 기법을 사용한다. 영상 정보를 암호화하는 과정을 블록체인 네트워크의 특성(분산, 투명성, 데이터 변조 불가)을 활용하여 개인정보 영상의 생성부터 소멸까지 사용자가 추적이 가능하도록 한다.

키워드 : 블록체인, 개인정보보호, 영상 감시 시스템

1. 서 론

영상 감시 시스템은 지정한 장소를 상시 녹화하고 있고 불특정 다수의 정보를 수집하는 특성을 가지고 있다. 영상 감시 시스템은 본질적으로 운영되는 시간 동안 끊임없이 자동적으로 특정 구역을 실시간으로 모니터링하거나 발생한 이벤트를 사건 이후 확인하기 위하여 녹화하는 것을 기반으로 구현되어 있다. 저장된 영상 파일은 카메라가 설치된 장소의 정보와 해당 위치에 방문한 인물들에 대한 정보를 담고 있으며, 경우에 따라서는 개인 정보 혹은 설치된 공간의 민감 정보를 내포하게 된다.

현존하는 클라우드 기반 영상 감시 서비스들은 공통적으로 카메라에서 취득한 영상을 인터넷 구간을 통해 전송하여 저장하는 방식을 취하고 있다. 이 때 네트워크를 통한 데이터 전송의 경우 전송 프로토콜에 SSL (Secure Sockets Layer)를 적용하여 데이터 채널을 보호하거나, 데이터그램 패킷과 같이 SSL의 적용이 어려움이 있는 경우에는 TLS (Transport Layer Security)[1]를 사용하여 전송 과정에서 실수 혹은 악의적인 탈취 시도로부터 안전하게 보호되고 있음을 객관적으로 증명할 수 있다.

그러나 안전하게 영상이 전송된 이후 서비스 내부에 저장된 영상에 대해서는 객관적으로 명시할 수 있는 보호 체계 없이, 각 서비스 제공자 별로 자체적으로 마련한 보안 정책 혹

※ 이 논문은 2020년도 서울시립대학교 연구년교수 연구비에 의하여 연구되었음.
※ 이 논문은 2020년 한국정보처리학회 춘계학술발표대회에서 “개인정보보호를 위한 영상 암호화 아키텍처 연구”의 제목으로 발표된 논문을 확장한 것임.
† 준 회 원 : 서울시립대학교 전자전기컴퓨터공학부 박사과정
†† 종신회원 : 서울시립대학교 전자전기컴퓨터공학부 교수
Manuscript Received : July 20, 2020
First Revision : September 14, 2020
Second Revision : October 5, 2020
Accepted : October 8, 2020
* Corresponding Author : Jaeho Lee(jaeho@uos.ac.kr)

은 기술적 장치를 이용하여 있는 실정이다. 일반적으로 저장된 영상을 보호하기 위해서는 데이터 자체를 보호하기 보다는 접근을 통제하는 방식으로 해결책을 마련하고 있다. 먼저 방화벽과 같은 네트워크 장치를 이용하여 시스템에 대한 악의적인 접근을 통제하고, 내부에 저장된 영상 파일에 대한 접근 허가는 시스템 내에서 권한을 부여받은 특정 사용자로 한정하여, 시스템 운영자일지라도 임의로 사용자의 데이터를 복사하거나 재생하지 못하도록 한다.

영상을 안전하게 보호하는 방법으로 전체 시스템의 보안 정책을 강화하는 방법을 제시하고 있고, 이는 서비스 제공자에게는 서비스 유지를 위해서 TCB (Trusted Computing Base)의 크기를 지속적으로 늘려나아가야 한다는 의미[2]가 되며, 사용자는 여전히 서비스 제공자별로 각기 다른 방식으로 제공하고 있는 접근 제어를 객관적 증거 없이 신뢰해야하는 상황에 놓이게 된다. 저장된 영상의 관점에서 본다면 시스템 외부로 데이터가 이동, 복사된 이후에는 서비스 제공자가 제공하는 접근 제어 방식의 통제를 벗어나게 된다. 이렇게 시스템의 접근 통제를 벗어난 이후에는 해당 영상 파일의 관리 책임은 온전히 사용자에게 귀속되며, 이를 안전하게 관리할 수 있는 객관적인 시스템 혹은 서비스는 부재인 상태이다.

본 연구에서는 개인 정보 보호의 관점에서 영상 정보를 안전하게 저장하고, 서비스 제공자가 제공하는 시스템 외부로 영상 정보를 복사, 이동하는 경우에도, 사용자가 여전히 해당 파일의 접근을 통제할 수 있는 방법을 제공하여, 개인정보의 오남용을 방지할 수 있는 아키텍처를 제안하고자 한다.

제안하는 아키텍처의 핵심은 영상 암호화를 위하여 사용한 암호화 키를 블록체인 네트워크를 통해서 관리하는 것이다. 그러나 본 연구의 목적은 단순히 암호키의 생성과 관리가 아닌, 블록체인 네트워크의 특성[3]인 탈중앙화, 영속성, 익명성, 추적 가능성에 착안하여 영상 정보를 전송하거나 저장하는 순간부터 이동, 복사, 그리고 삭제할 때까지 일련의 과정을 추적할 수 있는 장치를 마련하고 전체 시스템에 대한 사용자 신뢰도를 향상 시키고, 더 나아가서는 개인 정보를 담고 있는 디지털 자산에 대한 온전한 관리가 가능하도록 하는데 있다.

2. 관련 연구

개인정보보호를 위한 대표적인 아키텍처로는 G. Zyskind et al.[4]이 블록체인 아키텍처를 기반으로 하는 개인 데이터 보호 방안을 제시하고 있다. 개인이 휴대한 단말기에서 생성한 사진 혹은 메시지와 같은 간단한 데이터를 SNS와 같은 서비스로 전송할 때 사용자-서비스 상호간의 신원을 확인하기 위한 Compound Identity를 정의하고 데이터를 전송 상태를 블록체인에 기록하는 방안을 제시하고 있다.

또한 영상과 음성 데이터를 보호하기 위한 방법으로는 MPEG-CENC 표준[5]으로 제시되고 있으며, 단일 혹은 여러 개의 AES Key를 이용하여 멀티미디어 데이터를 암호화하는 방법으로 통용되고 있다. 이러한 멀티미디어 데이터 암호 기법은 데이터 자체의 보호보다는 Widewine, PlayReady 등과 같은 디지털 저작권 관리(Digital Right Management)의 관점에서 발전하고 있다. Vishwa et al.[6]은 블록체인 기반의 DRM을 연구하여 저작권을 보호하는 방법을 제시하고 있다.

분산 환경에서 컨텐츠를 암호화하고 사용자의 키를 관리하는 방법에 대해서는 블록체인 기반의 PKI (Public Key Infrastructure)[7]를 사용자-서비스 간의 신원확인 및 데이터 보호에 사용하는 방안이 제시되고 있다. 또한 민감한 데이터를 보호하는 방법에 대한 연구는 EMR (Electronic Medical Records)처럼 정보의 소유자가 아닌 제3자가 데이터를 수집하고 처리할 때 발생할 수 있는 정보 보호 이슈[8]를 해결하고자 새로운 아키텍처를 수립하기도 하였다.

또한 실질적인 데이터를 관리하는 상황에 있어서, 블록체인 기반의 접근 방법은 위변조가 불가능하고 개인정보보호에 사용이 가능하다는 것을 이야기하지만, 시스템을 설계할 때 보호하려는 데이터의 크기보다는 기록하는 데이터의 수에 따라 전체 시스템의 성능이 좌우된다는 연구 결과[9]를 고려하여 영상 암호화 아키텍처를 제안하고자 한다.

3. 영상 감시 시스템에서 데이터 보호 문제

일반적으로 데이터를 보호하기 위해서는 사용자가 능동적으로 생성한 데이터에 대해서 접근을 통제하거나 위변조를 방지하는 방법을 생각할 수 있다. 그러나 본 연구에서 다루는 영상 감시 시스템은, 사용자가 데이터를 능동적으로 생성하여 배포한다기 보다는, 사용자는 카메라의 물리적 설치 위치를 결정하게 되고, 이후에는 모든 데이터의 생성과 전송, 삭제는 시스템에 의해서 관리되는 특성을 가지고 있다. 이는 제3자에 의하여 생성된 데이터에 대한 소유권을 가지는 EMR 시스템과 유사한 특징을 가지고 있다.

3.1 Compound Identity의 복잡성

네트워크를 통한 안전한 데이터 전송을 위해서는 데이터 암호화가 필수적이다. 대칭키 암호화를 사용하는 경우에는 사전에 합의된 암호키를 교환하고 있어야 하기 때문에, 키 교환의 문제가 발생하며, 이를 해결하기 위하여 공개키-개인키 기반의 비대칭 암호화를 이용할 수 있다.

비대칭 암호화는 둘 이상의 관련자간의 공개키와 개인키를 사용하고 있으며, 암호화 데이터 전송이전에 공개키의 교환은 필수적인 절차이다. 그렇기 때문에 서로간의 공개키를 통하여 서로를 식별하는 Compound Identity를 구성하게 된다. Compound 집합은 공개키(pk)와 개인키(sk)의 2-tuple 혹은 완전한 식별데이터를 요구하는 경우 5-tuple(공개키, 개인키, 상대방 공개키와 개인키, 공유하는 대칭키)로 구성된다. 그러나 이러한 집합은 단순히 양방향 데이터 교환을 위해서는 강력한 암호화 메커니즘을 제공하는 기반이 되지만, 사용자와 다수의 서비스간의 경우로 환산한다면, Compound Identity 자체의 복잡도는 $O(n!)$으로 수렴하게 된다.

$$Compound^{(public)}_{u,s_1,s_2,...,s_n} = Compound^{(public)}_{u,s_1} + ... + Compound^{(public)}_{u,s_n} + Compound^{(public)}_{s_1,u} + ... + Compound^{(public)}_{s_1,s_n} + Compound^{(public)}_{s_2,u} + ... + Compound^{(public)}_{s_2,s_{n-1}} \quad (1)$$

영상 감시 시스템은 사용자가 직접 컨텐츠를 생성하는 것이 아니라, 카메라와 시스템이 자동적으로 생성하는 구조이기 때문에, 사용자-서비스 혹은 사용자-카메라간의 Compound Identity를 구성한다고 하면, 이러한 복잡도의 증가는 시스템의 확장성을 저해하는 요소가 된다.

3.2 장치 내 영상의 저장과 유출

클라우드 기반 영상 감시시스템에서 고려해야하는 또 다른 상황은 네트워크 단절 상황에서도 감시 카메라의 저장 기능은 동작해서 영상 유실을 최대한 방지 해야한다는 점이다.

카메라 내부에 저장된 영상은 영상 유실을 방지하기 위해한 별다른 보호조치가 없는 한, 물리적인 접근에 의하여 추출이 가능하며, 이렇게 유출된 영상에 대한 통제는 사실상 불가능에 가깝다. 보안 등급이 높은 구역 혹은 개인 정보에 매우 민감한 구역을 촬영하고 있는 감시 카메라는 장치 내부에 저장하는 영상을 사용자만 접근이 가능하도록 조치할 필요가 있다.

3.3 영상 데이터 암호화

비대칭 암호화 방식은 Compound Identity 복잡성 문제로 제기한 바와 같이 전송하려는 대상 모두의 공개키로 암호화와 복호화를 수행해야하는 복잡성 갖기 때문에, 데이터 자체의 암호화에는 사용하기가 어렵다.

P. Patila at el.[10]이 시험한 바와 같이 비대칭키 암호화 알고리즘의 처리 속도는 대칭키에 비하여 현저히 느리기 때문에 영상 데이터의 암호화에는 Asset Identity와 Compound Identity로 교환한 공개키는 사용하지 않으며 데이터를 암호화할 수 있는 대칭키를 교환하기 위한 수단으로 사용한다. 실제로 실시한 실험에서도 Fig. 1에서 보이는 바와 같이 대칭키에 사용하는 블록 암호화 알고리즘(AES-128)은 암호화와 복호화 모두 157 MB/s의 처리능력을 보이며, 비대칭키 암호화 알고리즘(RSA-2048)은 암호화시 0.3 MB/s, 복호화시 10 MB/s의 처리 능력을 보여, 대칭키 암호화 연산은 비대칭키 암호화는 연산 속도에서 암호화시 약 500배, 복호화시 약 15배 차이가 발생하며, 통상적으로 사용하는 1080p 영상의 실시간 스트리밍 대역폭[11]이 3Mbps 내외인 점을 감안한다면, 대칭키를 이용한 실시간 암/복호화는 스트리밍에 영향을 미치지 않는 수준이며, 카메라와 내부에 영상을 저장할 때에도 대칭키 암호화 방식을 사용한다면 임베디드 장치의 연산 성능을 크게 고려하지 않아도 사용할 수 있음을 알 수 있다.

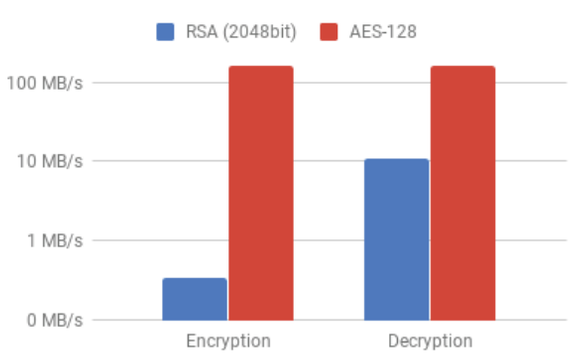

Fig. 1. Performance Comparison of RSA-2048 and AES-128

4. 제안 아키텍처

본 논문에서 제안하는 아키텍처는 사용자, 사용자가 소유한 카메라, 서비스 제공자, 그리고 블록체인 네트워크로 구성하고 있으며, 각 구성 요소간의 관계는 Fig. 2에 도식화한 것처럼 데이터의 양방향 전송이 필요한지 유무에 따라 결정된다.

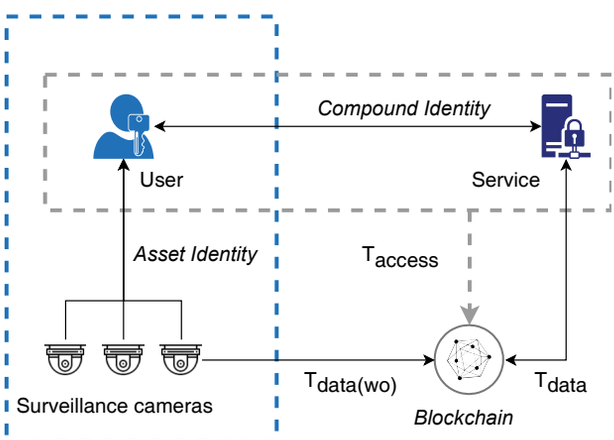

Fig. 2. Overview of Privacy Protection Architecture for Surveillance System

사용자-서비스처럼 양방향 데이터 전송이 필요한 경우에는 Compound Identity 관계를 설정하여 상호간 공개키를 교환할 수 있도록 하고, 사용자-카메라와 같이 단방향 전송만 필요한 경우에는 4.1 Asset Identity에서 후술하고 있는 관계로 설정하여 공개키 교환의 복잡성을 제한하고 있다.

서비스는 Compound Identity를 관계를 이용하여 서비스를 제공하는 모든 사용자의 공개키를 획득한 상태이며, 사용자 또한 사용하는 서비스의 공개키를 가지고 있다. 서비스는 사용자의 공개키를 이용하여 사용자에게 허가를 요청할 수 있으며, 사용자는 서비스의 공개키를 이용하여 데이터의 안전한 전송을 수행할 수 있다. 이때 사용자-서비스 상호간에 발생하는 요청과 허가는 하나의 블록체인 트랜잭션(T_{access})으로 관리된다.

영상 데이터를 전송하는 주체인 카메라는 Asset Identity 관계를 이용하여, 사용자의 공개키를 획득하여 데이터를 암호화하여 전송할 수 있는 상태가 되며, 실제 암호화된 데이터의 전송 또한 블록체인 네트워크의 하나의 트랜잭션으로 간주하게 되나, 카메라가 생성한 데이터를 다시 읽을 이유는 없

기 때문에 쓰기 전용 트랜잭션($T_{data(\mathrm{wo})}$)으로 블록체인 네트워크에 기록된다.

카메라와 달리 서비스와 사용자는 블록체인 네트워크에 기록하거나 읽을 수 있어야하기 때문에 데이터 트랜잭션에 읽기 권한을 포함(T_{data})할 수 있다.

4.1 Asset Identity

Compound Identity의 복잡도는 공개키를 공유를 필요로하는 양방향의 데이터 보호 채널을 구성하기 때문에 발생하게 된다. 영상 감시 시스템에서는 카메라와 사용자간의 관계상 사용자만이 카메라의 영상을 확인할 수 있도록 한정하는 Asset Identity를 정의하여 시스템의 복잡도를 획기적으로 낮추도록 하였다.

Require: $A \neq \varnothing$
procedure $ASSETIDENTITY(U, A)$
  if $(pk_{sig}^{U}, sk_{sig}^{U}) = \varnothing$ **then**
    $(pk_{sig}^{U}, sk_{sig}^{U}) \leftarrow G(sig)$
    $sk_{enc}^{U} \leftarrow G(enc)$
  end if
  for each $a_k \in A$ **do**
    $Nonce \leftarrow G(nonce)$
    $Nonce_{enc}^{a_k} \leftarrow Encrypt(pk_{sig}^{U}, Nonce)$
  end for
  return $(pk_{sig}^{U}, Nonce_{enc})$
end procedure

Algorithm 1. Generating Asset Identity

Algorithm 1에서는 **사용자-카메라** 혹은 **사용자-암호화된 비디오**의 관계를 설정하여 사용자의 공개키와 영상 암호화에 사용할 대칭키만을 조합하는 Nonce 개념을 소개하고 있다.

비디오를 암호화하기 위해 사용하는 대칭키로 Nonce를 사용하고, 사용자의 공개키로 암호화된 $Nonce_{enc}$와 장치 혹은 암호화된 비디오 파일의 관계를 Asset Identity로 정의하고, 이를 기반으로 암호화 키와 암호화 영상을 관리하는 트랜잭션을 구성할 수 있다.

카메라 혹은 서비스가 생성한 Nonce를 토대로 영상 파일을 암호화하는 경우 사용자만이 대칭키 정보를 알고 있기 때문에 영상 파일을 공개된 공간으로 전송하더라도 사용자의 개인키를 알지 못하는 한 영상 정보를 복호화 할 수 없게 된다.

결국 Equation (2)에서 표시한 것처럼 암호화된 Nonce를 복호화할 수 있는 개인키를 가진 사용자 혹은 시스템은 해당 Nonce를 사용하는 장치 혹은 파일에 대한 소유권을 가지고 있다고 가정할 수 있음을 의미한다.

$$\begin{aligned} Asset_{u,a} &= (pk_{sig}^{u}, Nonce_{pk(enc)}^{a}) \\ Asset_{u,a_1,a_2,\ldots,a_n} &= (pk_{sig}^{u}, Nonce_{pk(enc)}^{A}) \end{aligned} \tag{2}$$

Nonce는 사용자가 정보의 소유권을 가진 장치나 서비스 등 사용자의 공개키를 획득할 수 있는 제 3자에 의해서 생성이 가능하며, 사용자는 대칭키를 관리해야하는 부담에서도 동시에 벗어나게 된다. 또한 Nonce생성에는 단지 사용자의 공개키만을 요구하기 때문에 사용자와 암호화 채널을 생성해야하는 카메라 혹은 서비스가 증가함에도 그 복잡도는 여전히 $O(n)$으로 수렴한다.

4.2 프로토콜

개인정보보호를 위해 생성하는 일련의 정보는 블록체인 메모리(L)에 저장하는 것을 기본 전제로 한다. 영상 스트림의 경우, 저장하는 영상 정보는 대게 그 사이즈가 크기 때문에 L에 저장하는 것보다는 일반적인 저장소(ds)에 저장하고, 해시 함수(H)를 통해 매핑한 정보를 L에서 관리하도록 한다.

Nonce에 대한 공유 혹은 허가 정보를 L에 기록하여, 개인정보보호 관점에서 저장된 영상의 접근 제어뿐만 아니라 영상 정보의 생성부터 소멸까지 전반적인 라이프 사이클에 대한 추적이 가능하도록 한다.

특히 Fig. 3에서는 암호화된 영상을 요청하는 경우를 도식화 한 것으로, 요청자(Requestor)와 소유자(Owner)가 다른 경우에는 (8)Approval 절차를 거치도록 하여, 서비스 제공자가 명시적인 승인을 획득하지 못한 경우에는 해당 암호화 영상을 제공할 수 없도록 조치할 수 있음을 보여준다.

서비스 제공자는 Nonce와 Asset의 관계를 별도로 저장하지 않는 한 알 수 없으므로, AssetIdentity 정보를 얻기 위해서는 블록체인 메모리에 반드시 접근하여야 하며 이 절차를

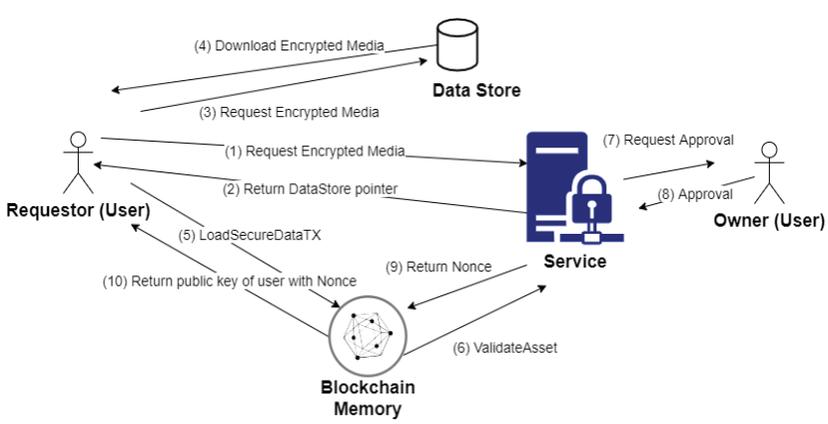

Fig. 3. The Loading Secure Data Sequence between Owner and Requestor

이용하여, Nonce에 관련된 모든 요청을 기록하여 어떠한 사용자 혹은 서비스가 해당 정보에 접근하는지 추적이 가능하도록 한다. 하지만 여전히 를 해석하기 위해서는 사용자의 허가가 필요하므로, 암호화된 영상의 복호화는 사용자의 결정에 따르게 된다.

4.3 저장 트랜잭션

Asset Identity절차를 통하여 Asset으로 분류되는 카메라는 Nonce 정보를 가지고 있기 때문에, $Nonce_{enc}$ 와 Asset 정보(a)를 암호화된 미디어 파일, M_{enc}, 내의 메타데이터로 기록하여 암호화된 미디어 파일 단독으로 off-chain을 통해 공유 가능한 상태가 된다.

Require: $M_{enc} \neq 0$
procedure $STORESECUREDATATX(pk_{sig}^{k}, M_{enc})$
  $(a_p, Nonce_{enc}^{p}) \leftarrow Parse(M_{enc})$
  if $ValidateAsset(pk_{sig}^{k}, a_p, Nonce_{enc}^{p}) \neq True$ **then**
    return 0
  end if
  $h_{M_{enc}} \leftarrow H(M_{enc})$
  $L[H(pk_{sig}^{k})] \leftarrow L[H(pk_{sig}^{k})] \cup h_{M_{enc}}$
  $ds[h_{M_{enc}}] = M_{enc}$
  return $h_{M_{enc}}$
end procedure

Algorithm 2. Storing Secure Data

Algorithm 2는 StoreSecureDataTX를 이용한 데이터 저장소와 블록체인 메모리간의 상호 운영에 대한 절차를 설명하고 있다. 위에서 언급한대로 암호화된 미디어에 기록된 Nonce는 아무런 제약없이 추출이 가능한 메타데이터이기 때문에 본 연구에서 제안하는 시스템은 카메라 내부 혹은 클라우드 기반의 데이터 저장 서비스를 블록체인 네트워크 혹은 암/복호화 과정과 분리하여 수행할 수 있도록 하여 시스템의 확장성을 고려하고 있다.

StoreSecureDataTX내에서 수행하는 ValidateAsset은 트랜잭션 내에서 비즈니스 로직이 개입할 수 있는 보조적인 장치로 사용되고 있으며, 이를 통하여 사용자, Asset, Nonce정보가 일치하는지 확인할 때 블록체인 네트워크에 M_{enc}에 대한 접근 상황을 추적할 수 있다.

4.4 읽기 트랜잭션

Algorithm 3 은 ds와 H를 이용하여 M_{enc}를 획득한 후 복호화를 수행하는 과정을 설명하고 있다.

StoreSecureDataTx를 이용하여 저장된 미디어의 정보는 요청하기 위해서는 요청자의 공개키(pk_{sig}^{k})를 사용해야 하며 암호화된 미디어의 메타데이터(m)에는 암호화된 미디어를 가져올 수 있는 실제 위치 정보($h_{M_{enc}}$)와 접근 권한이 있는 공개키 정보(x_p)가 이미 나열 되어 있어, 권한이 없는 사용자의 접근을 사전에 차단할 수 있다.

또한 제 3자에 의한 복호화 요청에도, *CheckPolicy*를 정의하여 본래 Asset의 소유주인 사용자에게 요청을 허가할지 결정할 수 있도록 하여, M_{enc}의 복호화 과정을 블록체인 네트워크에 기록할 수 있도록 한다. 또한 사용자가 허가할 수 있는 권한의 종류는 downloadable, readable 등으로 세분화하여 세밀한 권한 제어가 가능하도록 한다.

요청자가 M_{enc}를 획득한 이후에도 ValidateAsset을 수행하여 사용자가 허가한 경우에 한하여 평문의 대칭키를 획득 가능하도록 하여 미디어 파일의 획득과 복호화 과정을 분리하여 추적할 수 있다.

Require: $m \neq 0$
procedure $LOADSECUREDATATX(pk_{sig}^{k}, m)$
  $(h_{M_{enc}}, x_p) \leftarrow Parse(m)$
  if $\widetilde{Policy(pk_{sig}^{k}, x_p)} \neq True$ **then**
    return $Error$
  end if
  if $h_{M_{enc}} \in L[H(pk_{sig}^{k})]$ **then**
    $M_{enc} \leftarrow ds[h_{M_{enc}}]$
    $(a_p, Nonce_{enc}^{p}) \leftarrow Parse(M_{enc})$
    if $ValidateAsset(pk_{sig}^{k}, a_p, Nonce_{enc}^{p}) \neq True$ **then**
      return 0
    end if
  end if
  return $(pk_{sig}^{k}, h_{M_{enc}}, a^p, Nonce_{enc}^{p})$
end procedure

Algorithm 3. Loading Secure Data

4.5 추적 트랜잭션

영상 정보의 생성 이후 M_{enc}의 소유권 이전과 소멸에 대한 관리를 위한 트랜잭션으로 대용량의 미디어 파일의 반복적인 암복호화 과정 없이 Nonce정보를 추가하여 off-chain상에서도 전달 과정을 Algorithm 4를 통해 제시하고 있다.

Require: $T \neq \varnothing \vee m \neq 0$
procedure
  $TRANSFERSECUREDATATX(pk_{sig}^{k}, T, m)$
  $(pk_{sig}^{k}, h_{M_{enc}}, a^p, Nonce_{enc}^{p})$
  $\leftarrow LOADSECUREDATATX(pk_{sig}^{k}, m)$
  $M_{enc} \leftarrow ds[h_{M_{enc}}]$
  if $M_{enc} \neq 0$ **then**
    $(pk_{sig}^{t}, Nonce_{enc}^{t}) \leftarrow ASSETIDENTITY(T, a^p)$
    $M_{enc}^{t} \leftarrow M_{enc} \cup Nonce_{enc}^{t}$
    $h_{M_{enc}} \leftarrow STORESECUREDATATX(pk_{sig}^{t}, M_{enc}^{t})$
  end if **return** $h_{M_{enc}^{t}}$
end procedure

Algorithm 4. Transferring Secure Data

제시된 알고리즘에서 설명하듯 M_{enc}^{t}는 $Nonce_{enc}^{k}$와 $Nonce_{enc}^{t}$의 정보를 모두 가지고 있기 때문에 사용자k와 t는 M_{enc}^{t}를 복호화할 수 있으나, M_{enc}^{k}를 M_{enc}^{t}로 전달한다고 하더라도 M_{enc}의 복호화 혹은 복호화 후 재 암호화 과정을 필요로 하지 않는다.

4.6 무효화 트랜잭션

영상 감시 시스템에서 또 다른 주요 이슈는 생성된 영상 정보를 영구히 제거하는 것이다. 제안된 시스템에서도 임의로 저장한 M_{enc}의 복사를 제한할 수 있는 방법은 없으나, Algorithm 5는 Nonce 자체를 무효화하여 결과적으로는 의 복호화 방법을 차단하는 간접적인 절차를 통해 해당 영상 정보에 접근을 영구히 제거하는 방안을 제안한다.

물론 Nonce 조차도 임의의 공간에 별도로 보관하는 경우 무효화된 M_{enc}를 복호화 하려는 시도는 가능하나 일반적으로 난수 생성기(Random Number Generator)를 통해 생성된 값으로 공격의 대상이 되는 모든 M에 대하여 무효화 이전에 예측 불가능한 Nonce를 별도의 시스템에서 관리하는 것은 사실상 불가능에 가깝다.

procedure
$\in VALIDATESECUREDATATX(pk^{k}_{sig}, h_{M_{enc}})$
$M_{enc} \leftarrow ds[h_{M_{enc}}]$
$(a_p, Nonce^{p}_{enc}) \leftarrow Parse(M_{enc})$
$M_{enc} \leftarrow M_{enc} - Nonce^{t}_{enc}$
if M_{enc} *has no Nonce* **then**
$ds[h_{M_{enc}}] = \varnothing$
end if
end procedure

Algorithm 5. Invalidating Secure Data

4.7 데이터 처리 연산 기반 서비스 컴포넌트 설계

앞서 정의한 데이터 트랜잭션은 블록체인 메모리와 대용량 저장 장치에 암호화 처리된 영상을 안전하게 저장하고 추출하는 기본적인 절차를 정의하고 있다. 제시한 절차들이 실제 시스템을 구현하기 위한 필수 요소들을 포함하고 있는지 검증을 하기 위해서는 일반적인 데이터 처리 연산인 CREATE, READ, UPDATE, DELETE (CRUD)에 해당하는 기능과 일치하거나 유사하게 구현이 가능한지 살펴볼 필요가 있다. 본 제안 아키텍처에서는 Fig. 4에서 유즈케이스로 도식화 한 것처럼 각 연산을 다음과 같이 정의하여 암호화된 영상을 관리하도록 한다.

- CREATE: 암호화된 영상 및 Nonce를 생성하고 저장
- READ: 암호화된 영상과 복호화에 필요한 정보를 가져오는 연산
- UPDATE: 암호화된 영상의 Nonce를 수정
- DELETE: 암호화된 영상의 복호화 정보를 영구히 제거

또한 사용자가 암호화된 영상을 관리하기 위한 추가적인 연산을 정의할 필요가 있으며, 해당 연산은 UPDATE의 확장 형태로 정의가 가능하다.

- SHARE: 다른 사용자가 영상 복호화를 할 수 있도록 Nonce 정보를 전달
- GRANT PERMISSION: 다른 사용자가 영상 복호화를 할 수 있도록 암호화 영상의 메타데이터로 Nonce 정보를 추가하고 데이터 저장소와 블록체인 메모리에 반영

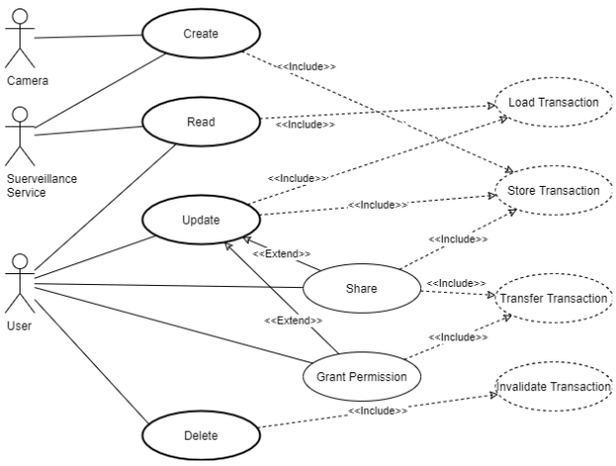

Fig. 4. Use Case Diagram of the Basic Data Operations Based on the Proposed Transactions

최종적으로 암호화된 영상을 생성하고 관리하기 위한 각 CRUD 연산은 앞서 정의한 프로토콜을 따르도록 되어 있으나, 데이터 트랜잭션을 이용하여 개인 정보 보호 문제를 해결하기 위한 시스템을 구현하기 위해서는 Fig. 5에서 보이는 바와 같이 공통 서비스 컴포넌트들을 필요로 하게 된다.

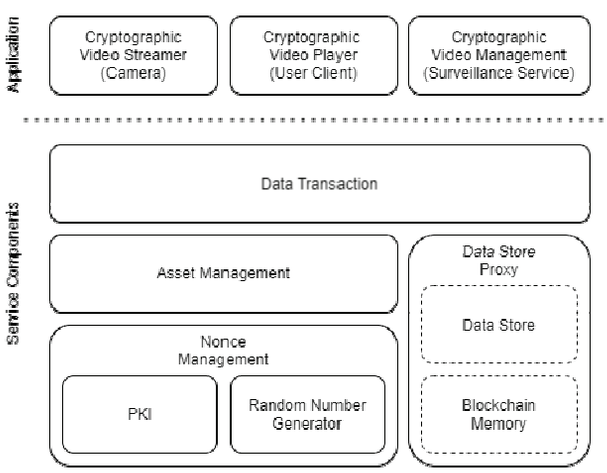

Fig. 5. Service Components Deployment

앞서 제시한 Asset Identity를 기반으로 각 Asset과 사용자의 관계 및 블록 암호화에 사용하는 대칭키를 관리하고자 할때는 Nonce Management 만으로 충분할수 있다. 그러나 실제 구현하는 시스템에서는 사용자-Asset 관계에서 암호화 정보가 불필요한 경우, 별도의 Asset Management 컴포넌트를 구축할 수 있다.

또한 시스템을 설계할때 데이터의 특성에 따라 서비스 자체적으로 구축한 데이터 저장소를 사용하거나 블록체인 메모리에 접근할지를 결정해야한다. 그러나 블록체인 네트워크의

탈중앙화 특성으로 인하여 모든 서비스 컴포넌트들은 임의로 블록체인 메모리에 접근할 수 있기 때문에 시스템 설계자의 의도와 다르게 구현자의 즉흥적인 결정으로 임의의 데이터 저장소를 사용하게 되는 경우가 발생할 수도 있다. 이를 방지하기 위하여, 저장하려는 데이터 특성을 결정하는 기준을 각 서비스 컴포넌트에 위임하기 보다는 데이터 저장을 위한 프록시 컴포넌트(Data Store Proxy)를 제공하여 데이터가 저장되는 공간을 시스템 설계자의 의도에 맞게 선택할 수 있도록 처리할 필요가 있다

결과적으로 제공되는 서비스 컴포넌트들은 영상 감시 시스템을 구현하기 위한 공통 컴포넌트로 제공되며, 이를 기반으로 카메라(Cryptographic Video Streamer), 영상 감시 시스템 내에서의 보안 영상 관리(Cryptographic Video Management), 암호화된 영상의 재생 및 관리 (Cryptographic Video Player) 등을 위한 응용 프로그램을 구축 할 수 있게 된다.

5. 결 론

사회 안전을 위하여 널리 사용되고 있는 영상 감시 시스템의 시스템 자체가 개인정보보호를 위해 보호되어야 하는 대상이 되고 있다. 시간이 지남에 따라 더 많은 사용자와 시스템이 관련되기 때문에 악의적인 접근 혹은 단순한 실수에 의해서도 민감한 영상이 공유되는 상황은 발생할 수 있으나 시스템에 대한 접근 차단 외에는 뚜렷한 보호 장치는 없는 상태이다.

Table 1은 영상 감시 시스템에서 개인 정보를 담고 있는 영상을 보호하는 필수 요소를 다음과 같이 나열하고 비교하고 있다.

- 시스템에 적용할 수 있는 암호화 방법(Encryption)
- 실시간 전송시의 암호화 방법(Live Streaming)
- 외부 저장시 영상 암호화 여부(Export Encrypted Video)
- 오프라인 상태에서 개인 정보 보호 지원여부(Off-Network Privacy Protection)
- 객관적으로 신뢰할 수 있는 개인 정보 보호 방법 제공 여부(TCB Free)

이러한 기준에서 제안된 아키텍처와 비교를 해본다면, 저장된 영상을 보호하는 방법은 DRM이 가장 유사하며 강력한 보호 방법을 제시하고 있다. 그러나 저작권 보호를 기반으로 하여 하나의 저작권을 다수의 사용자에게 배포하는 방법을 중심으로 발전하고 있어 실시간성과 다수의 영상을 소수의 사용자에게 공유하는 영상 감시 시스템 적용에 다소 무리가 있다.

또한 기존에 연구 개발되어 배포되고 있는 영상 감시 시스템의 경우 실시간성과 저장 영상에 대한 관리는 매우 철저하게 이루어지고 있으나, 시스템 외부로 유출된 경우에 대한 대비와 시스템 내부의 개인정보 보호 문제에 대한 객관적 증명이 이루어지지 않아 TCB의 규모를 무한정 확장하더라도 개인 정보 보호는 시스템 내에서만 가능하다는 한계를 가지고 있다.

Table 1. Comparison of Privacy Protection Approach

	DRM[1]	VS[2]	VSaaS[3]	Proposed Architecture
Encryption	Symmetric Key	SSL	SSL	SSL Symmetric Key
Live Streaming	◑	◑	◑	○
Export Encrypted Video	○	◑	◑	○
Off-Network Privacy Protection	○	×	×	○
TCB[4] Free	○	×	×	○

[1] Digital Right Management
[2] On-premise Video Surveillance Service
[3] Video Surveillance as a Service
[4] Trusted Computing Base

본 연구에서 제안한 아키텍처는 사용자의 정보를 담고 있는 영상 정보의 관리에 중점을 두고 있다. 사용자가 소유한 영상 파일에 대해 저장, 읽기를 수행하는 일련의 과정을 투명성이 보장된 블록체인 네트워크에 저장하여, 소유한 디지털 자산에 대한 모든 접근 정보를 알수있게 할 뿐만 아니라, 소유자가 원하지 않는 경우 해당 파일의 복호화를 할 수 없도록 하여 사실상 해당 정보가 영구히 삭제할 수 있는 기능을 제공한다.

이렇듯 사용자가 직접 생성하거나, 아니면 권한을 위임받은 제 3자가 생성한 데이터에 대한 전반적인 라이프 사이클의 통제가 가능하도록 하여, 사용자의 의도와 다르게 해당 정보가 사용되는 것을 미연에 방지하는 것을 목적으로 하는 시스템을 제안하였다. 그러나 제안된 아키텍처가 제공하는 기능은 영상 감시 시스템 뿐만 아니라 추후 연구를 통하여 데이터의 생성과 소유가 분리되는 일반적인 경우에도 적용할 수 있을 것으로 기대된다.

References

[1] B. Barnes, M. Thomson, A. Pironti, and A. Langley, “Deprecating secure sockets layer version 3.0”, RFC 7568, doi: 10.17487/RFC7568, June 2015, 〈https://www.rfc-editor.org/info/rfc7568〉.

[2] D. C. Latham, “Department of defense trusted computer system evaluation criteria. Department of Defense,” 1986.

[3] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, “An overview of blockchain technology: Architecture, consensus, and future trends,” *2017 IEEE International Congress on Big Data (BigData Congress)*, Honolulu, HI, pp.557-564, 2017.

[4] G. Zyskind, O. Nathan, and A. Pentland, "Decentralizing privacy: Using blockchain to protect personal data," *In 2015 IEEE Security and Privacy Workshops*, pp.180-184, May 2015.
[5] ISO/IEC 23001-7:2016, Part 7: Common encryption in ISO base media file format files In Information technology - MPEG systems technologies Retrieved from https://www.iso.org/standard/68042.html
[6] Alka Vishwa and Farookh Hussain, "A blockchain based approach for multimedia privacy protection and provenance," *In 2018 IEEE Symposium Series on Computational Intelligence (SSCI)*, pp.1941-1945, Nov. 2018.
[7] R. Wang, J. He, C. Liu, Q. Li, W. Tsai, and E. Deng, "A privacy-aware PKI system based on permissioned blockchains," *In 2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS)*, pp.928-931, Nov. 2018.
[8] A. Azaria, A. Ekblaw, T. Vieira, and A. Lippman, "Medrec: Using blockchain for medical data access and permission management," *In 2016 2nd International Conference on Open and Big Data (OBD)*, pp.25-30, Aug. 2016.
[9] X. Liang, S. Shetty, D. Tosh, C. Kamhoua, K. Kwiat, and L. Njilla, "Provchain: A blockchain-based data provenance architecture in cloud environment with enhanced privacy and availability," *In 2017 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID)*, pp.468-477, May 2017.
[10] P. Patila, P. Narayankar, N. D G and M. S M, "A comprehensive evaluation of cryptographic algorithms: DES, 3DES, AES, RSA and Blowfish," *Procedia Computer Science*, Vol.78, pp.617-624, 2016.
[11] Microsoft Stream Video Delivery and Network Overview [Internet], https://docs.microsoft.com/en-us/stream/network-overview

김 정 석
https://orcid.org/0000-0001-6431-7910
e-mail : justinkim@uos.ac.kr
jeongseok.kim@sk.com
2001년 서울시립대학교 환경공학부(학사)
2003년 서울시립대학교 전자전기컴퓨터공학부(석사)
2009년 ~ 현 재 서울시립대학교 전자전기컴퓨터공학부 박사과정
2018년 ~ 현 재 에스케이텔레콤 연구원
관심분야 : 인공지능, 암호학, 미디어

이 재 호
https://orcid.org/0000-0002-3332-3207
e-mail : jaeho@uos.ac.kr
1985년 서울대학교 계산통계학과(학사)
1987년 서울대학교 계산통계학과(석사)
1997년 University of Michigan(박사)
1998년 ~ 현 재 서울시립대학교 전자전기컴퓨터공학부 교수
관심분야 : 인공지능, 지능 로봇