제21회 한국정보처리학회 춘계학술발표대회 논문집 제11권 제1호 (2004. 5)

ASF 파일 보호를 위한 DRM 툴킷 설계

박지현*, 김정현*, 윤기송*
*한국전자통신연구원 디지털콘텐츠연구단
e-mail : juhyun@etri.re.kr

Design of DRM Toolkit to Protect ASF Files

Ji-Hyun Park*, Jung-Hyun Kim*, Ki-Song Yoon*
*Digital Content Research Division, ETRI

요 약

디지털 정보의 생성과 배포의 용이성은 디지털화된 동영상의 제작을 증가시키고 있다. 그러나 디지털 콘텐츠는 복제가 쉽고 누구나 접근이 가능한 인터넷을 통해 전달될 수 있으므로 저작권 침해가 중요한 문제로 대두되고 있다. 이러한 문제점은 특히 사용자의 PC 에 저장되어 재생되는 동영상에 주로 발생하였으며, 종래에는 사용자의 PC 에 저장된 동영상 파일의 보호를 위해 동영상 파일의 전체를 암호화한 후 적법한 사용자에게만 암호화된 동영상 파일을 복호화 할 수 있는 정보를 제공하는 방법을 사용하였다. 사용자의 PC 에 저장되어 재생되는 동영상에 비하여 스트리밍 서비스를 통하여 재생되는 동영상은 사용자의 PC 에 저장되지 않도록 함으로써 불법 복제등의 문제를 해결하였다. 하지만 근래 스트리밍을 통하여 서비스되는 동영상을 저장할 수 있는 프로그램이 속속 등장하면서 스트리밍을 통하여 서비스되는 동영상에 대한 보호 방안도 필요하게 되었다. 본 논문에서는 스트리밍 컨텐츠로 가장 널리 사용되고 있는 마이크로소프트의 멀티미디어 파일 포맷인 ASF 를 기반으로한 스트리밍 시스템에 DRM 을 적용하는 방안에 대하여 설명한다..

1. 서론

인터넷 및 네트웍의 환경 변화는 고품질, 고용량의 콘텐츠의 실시간 서비스를 가능하게 만들었다. 디지털 콘텐츠에 대한 요구가 증가함에 따라 저작권에 관한 문제가 더욱 중요한 쟁점이 되고 있다. VOD 와 같은 스트리밍 서비스를 통하여 재생되는 동영상은 사용자의 PC 에 저장되지 않도록 함으로써 불법 복제등의 문제를 해결하였다. 하지만 근래 스트리밍을 통하여 서비스되는 동영상을 저장할 수 있는 프로그램이 속속 등장하면서 스트리밍을 통하여 서비스되는 동영상에 대한 보호 방안도 필요하게 되었다.

기존의 스트리밍 서버는 동영상을 스트리밍을 통하여 전송할 때 해당 동영상 파일의 정보를 이용하여 실시간으로 보내야할 미디어 데이터를 결정한다. 스트리밍을 통하여 서비스되는 동영상의 보호를 위하여 사용자의 PC 에 저장된 동영상 파일의 보호 방안[8,9]을 사용하여 파일의 전체를 암호화하게 되면 스트리밍 서버가 미디어 데이터를 읽어오기 위하여 필요한 정보까지도 변형되기 때문에 스트리밍이 불가능하게 된다. 따라서 다운로드되어 서비스되는 콘텐츠의 보호 방안과는 다른 방법의 암호화 방법이 스트리밍되는 콘텐츠에 적용되어야 한다.

본 논문에서는 현재 스트리밍 콘텐츠로 가장 많이 사용되고 있는 마이크로소프트의 ASF(Advanced Systems Format) 파일을 기반으로 스트리밍 컨텐츠의 보호 방법에 관하여 설명한다.

2. 시스템 구성

DRM[7] 기반 스트리밍 서비스를 위한 전체 시스템은 ASF 파일 보호 프로그램, 라이센스 발급 서버, 스트리밍 서버, 클라이언트로 구성된다.

ASF 파일 보호 프로그램은 ASF 파일을 암호화하여 보호하는 기능을 한다. 암호화된 파일이 스트리밍 서버에 의하여 전송되는데 문제가 없도록 하기 위하여 순수 미디어 데이터만을 암호화한다. 또한 암호화 과정에서 발생한 정보를 라이센스 발급 서버에 등록하는 기능을 수행한다.

라이센스 발급 서버는 사용자에게 보호된 ASF 파일을 재생하는데 필요한 정보들을 포함하고 있는 라이센스를 전송한다. 라이센스는 복호화키와 같이 노출

되어서는 안되는 정보를 포함하고 있으므로 안전한 전송 채널을 통하여 전달되도록 한다[10].

스트리밍 서버는 사용자에게 콘텐츠를 스트리밍 서비스하는 일반적인 기능을 수행하며 마이크로소프트의 미디어서버를 이용한다.

클라이언트는 동영상 재생기, 복호화 모듈, DRM 모듈로 구성된다. DRM 모듈은 라이센스를 라이센스 발급 서버로부터 전송받아 관리하고, 복호화 모듈에 복호화에 필요한 정보를 제공하며, 사용자 PC 에서 발생할 수 있는 해킹을 막기 위한 여러가지 보안 기능을 제공한다.

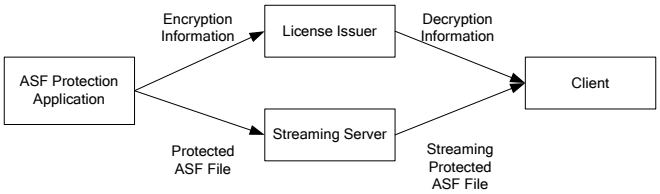

그림 1. 시스템 구성

3. ASF 파일 보호

3.1. ASF

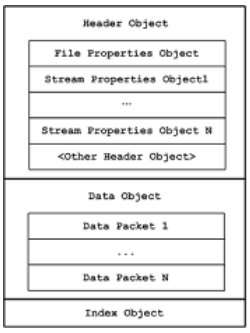

그림 2. ASF 구조

마이크로소프트사가 1996 년에 멀티미디어 스트리밍용으로 고안한 파일 포맷인 ASF 는 상당히 유연한 포맷으로 같은 시간으로 동기화 된 오디오, 비디오, 텍스트, 스크립트 등을 포함하고 있다. 즉, ASF 는 동기화된 멀티미디어 데이터를 저장할 수 있도록 설계된 확장 가능한 파일 포맷으로 로컬 재생에 적합하면서도 다양한 네트워크와 프로토콜 상에서의 데이터 전송을 지원한다. ASF 는 확장 가능 미디어 유형, 저작자 지정, 스트리밍 우선순위 부여, 다중 언어 지원, 문서 및 콘텐트 관리를 비롯한 광범위한 서지 기능 등 고급 멀티미디어 기능을 지원한다[1,6].

ASF 는 기능에 따라 다양한 객체들로 구성되어 있으며 사용자의 의도에 따라 객체의 확장이 가능하기 때문에 유연한 구조를 가진다. ASF 의 구조는 그림 2 과 같다[3]. 그림 2 에서와 같이 ASF 의 가장 상위 객체는 헤더객체, 데이터 객체, 인덱스 객체이다. 헤더객체는 일반적인 파일정보와 스트림 정보로 구성되며 미디어 재생기가 프레젠테이션을 올바로 렌더링하는데 필요한 정보를 포함한다. 데이터 객체는 실제적인 미디어 데이터를 가진다. 이들 데이터는 스트리밍시 보내질 패킷단위로 저장되어 있다. 인덱스 객체는 미디어 데이터의 시간단위의 전후 탐색이 가능하도록 하는 인덱스 정보를 가진다.

3.2. 암호화 방안

스트리밍 서버는 동영상을 스트리밍을 통하여 전송할 때 해당 동영상 파일의 정보를 이용하여 실시간으로 보내야할 미디어 데이터를 결정한다. 스트리밍을 통하여 서비스되는 동영상의 보호를 위하여 사용자의 PC 에 저장된 동영상 파일의 보호 방안을 사용하여 파일의 전체를 암호화하게 되면 스트리밍 서버가 미디어 데이터를 읽어오기 위하여 필요한 정보까지도 변형되기 때문에 스트리밍이 불가능하게 된다.

이 같은 문제를 해결하기 위해서 본 논문에서는 스트리밍 서버가 필요로 하는 정보를 변형하지 않고 순수한 미디어 데이터만을 암호화하는 방법을 사용한다.

그림 3 은 ASF 파일중 실제 데이터 패킷의 세부 구조를 나타낸다. 데이터 패킷은 스트리밍시 클라이언트에 전송되는 실제 패킷이다. 데이터 패킷은 에러 보정 데이터, 페이로드 파싱 정보, 페이로드 데이터, 패딩 데이터로 구성된다. 이중 페이로드 데이터에 실제 미디어 데이터가 포함되어 있으며, 미디어객체 번호가 같은 미디어 데이터를 모으면 오디오나 비디오의 한 프레임 데이터를 구성할 수 있다.

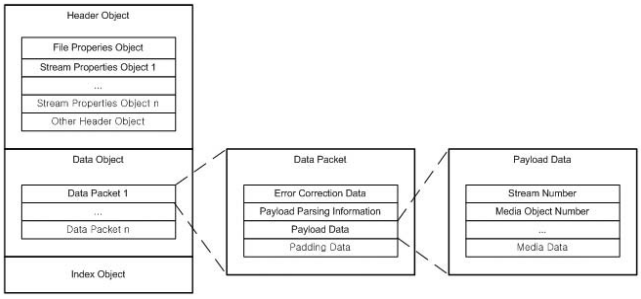

그림 3. 데이터 패킷의 구조

그림 4 는 암호화 과정을 나타낸다. 미디어 데이터를 프레임 단위로 암호화하기 위해서 먼저 미디어 객체 번호가 같은 페이로드로부터 하나의 프레임 데이터를 만든다. 프레임 데이터를 암호화한 후 암호화 여부를 표시하기 위한 1 바이트 데이터를 프레임의 마지막에 추가한다. 암호화된 프레임 데이터를 원래 미디어 데이터의 크기만큼 나누어 다시 각 페이로드에 분배한다. 암호화 여부를 표시하기 위한 1 바이트 데이터를 추가하였으므로 마지막 페이로드 데이터의 크기는 1 만큼 증가하게 된다. 페이로드의 크기가 변경되었으므로 관련되는 메타데이터들을 수정한다.

비디오 데이터는 키프레임인 것과 아닌 것의 두가지 종류로 구분할 수 있다. 키프레임은 디코딩시 다른 프레임을 참조하지 않고 그 프레임 자체만으로 디코딩이 가능한 것들이다. 복호화에 의해서 발생할 수 있는 스트리밍시의 성능 저하를 감안한다면 키프레임만을 암호화하는 것으로 콘텐츠를 보호할 수 있다. 이렇게 암호화된 콘텐츠는 ASF 파일을 재생하는 플레

이어로 재생가능하지만 복호화에 관한 정보 없이는 제대로된 화면을 볼 수 없다.

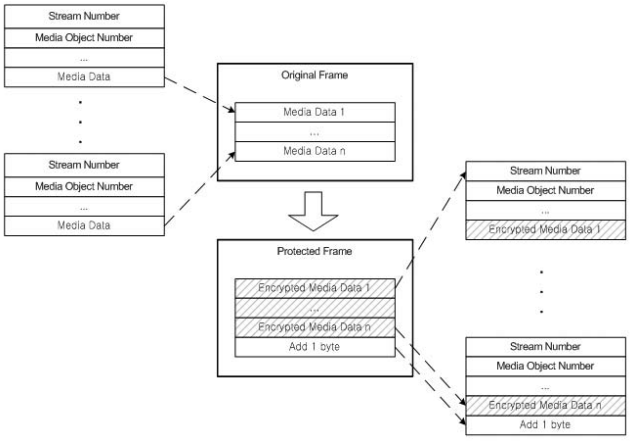

그림 4. 미디어 데이터 암호화

본 논문에서는 8 바이트 키를 사용하는 DES[2] 알고리즘을 사용하여 데이터를 암호화하였다. 암호화에 의한 크기 변화를 최소화하기 위하여 프레임 데이터에서 8 의 배수에 해당하는 데이터만을 암호화하고 나머지 데이터는 암호화하지 않는다. 그림 5 는 암호화된 ASF 비디오 프레임의 구조를 나타낸다.

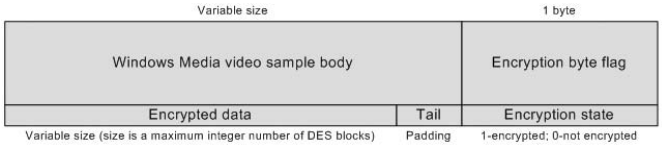

그림 5. 미디어 데이터 암호화

이 같은 암호화 방법을 통하여 암호화시 여러가지 옵션을 주는 것이 가능하다. 표 1 은 가능한 암호화 옵션의 목록을 보여준다. 암호화시 각 프레임에 1 바이트 데이터를 추가함으로써 파일 크기와 스트리밍시 전송되어야 하는 데이터의 크기는 다소 증가하였지만 다양한 암호화 옵션을 주는 것이 가능하게 한다.

표 1. 암호화 옵션

Option Type	Option List
Media Type	- Video Only
	- Audio Only
	- Both Video and Audio
Video Media	- Key Frames Only
	- All Frames
Range	Time based encryption range

위와 같은 암호화 방법은 기존의 스트리밍 서비스 환경을 그대로 사용하면서도 동영상을 보호할 수 있고 전체 시스템의 성능에 별다른 영향을 주지 않는 장점을 가진다.

4. 보호된 파일의 재생

복호화는 마이크로소프트의 DMO(DirectX Media Objects) 기술에 기반한 필터에서 수행된다.

4.1. DirectShow 기술

DirectShow 는 마이크로소프트에서 개발한 멀티미디어 처리 기술이다. DirectShow 는 멀티미디어가 갖는 다양한 입력, 다양한 포맷, 다양한 출력에 대한 문제를 해결하기 위해 컴포넌트 구조를 도입하였다. 컴포넌트 구조가 갖는 유연성을 활용하여 다양한 환경에 대해 컴포넌트를 적절히 조합함으로써 필요한 상황에 다양한 형태로 멀티미디어 데이터를 처리할 수 있도록 하였다[3].

DirectShow 는 필터(filter)라는 구조의 컴포넌트를 도입하고 이들을 조합하여 다양한 멀티미디어 환경에 대응할 수 있도록 설계되었다. 필터는 마이크로소프트의 COM(Component Objet Model)의 기술을 기반으로 제작되어 컴포넌트의 장점을 가질 수 있도록 하였다. 윈도우즈에서 사용되는 멀티미디어 재생기는 대부분이 기술을 이용하여 개발되며 사용자의 의도에 맞는 필터를 쉽게 개발하고 적용시킬 수 있다.

4.2. DMO

DirectX Media Objects(DMO)는 데이터 스트리밍 컴포넌트를 만드는 방법으로 DirectShow 필터와 유사한 부분이 있다[5,6]. DirectShow 필터와 같이 DMO 는 입력된 데이터를 처리해서 출력 데이터를 생성한다. 그러나, DMO API 는 DirectShow API 보다 간단한 구조를 갖기 때문에 좀 더 쉽게 개발할 수 있다는 장점을 가지고 있다. DMO 는 DirectShow 와 함께 사용될 수 있지만 DirectShow 를 사용하지 않고도 어플리케이션을 이용할 수 있다. DirectShow 를 사용하지 않는 어플리케이션은 DMO 만을 직접적으로 사용가능하다. 따라서, DMO 를 사용함으로써 여러 종류의 어플리케이션을 생성할 수 있다. 윈도우 미디어 플레이어는 ASF 파일 재생시 DMO 필터만을 이용 가능하도록 만들어져 있으므로 본 논문에서는 DMO 필터를 이용하여 복호화를 수행하도록 하였다.

4.3. 복호화 방안

DMO 방식의 복호화는 영상 파일 각각의 스트림에 명시된 미디어 타입을 디코딩하는데 사용되는 디코더를 렌더링 시스템이 찾아내기 때문에 가능하다. 일반적으로 동영상 파일은 필터 그래프에서 적당한 코덱과 연계된다. 그러나 본 개발에서는 재생기가 디코딩을 위해 정해진 코덱이 아닌 가상 코덱인 DMO 와 연계되는 방식을 이용한다. 본 논문의 경우는 동영상 데이터가 압축되었다고 가정하기 때문에 특별한 문제는 없다. 위와 같은 DMO 문제점을 해결하기 위해 미디어 스트림 출력에서는 원래의 미디어 타입을 복원한다. 이는 재생기가 원래의 데이터를 적절히 재생하는데 필수적이다.

코덱 ID 를 대체하기 위해 ASF 파일 구조와 미디어 타입 매핑 부분을 수정한다. WMF 와 DirectShow 라이브러리의 미디어 타입에서메이저 타입과 서브타입을 이용해서 데이터를 디코딩하는 필요한 코덱을 결정하

고 나머지 데이터들을 이용해서 코덱에서 포맷 블록에 저장되어 있는 스트림의 특성들을 결정하는데 사용된다. 그림 6 은 원본 콘텐츠를 재생하기 위한 필터 그래프 구조이고 그림 7 은 보호된 콘텐츠를 재생하기 위한 필터 그래프 구조이다.

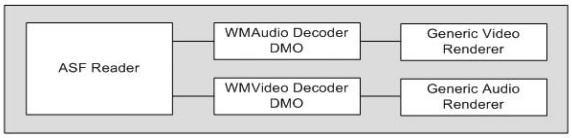

그림 6. 원본 콘텐츠의 필터 그래프

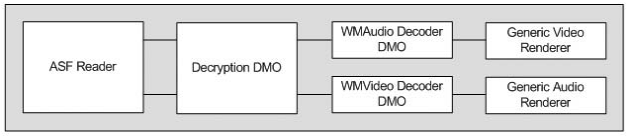

그림 7. 보호된 콘텐츠의 필터 그래프

그림 8 은 복호화되지 않고 재생하는 화면과 복호화를 통하여 재생되는 화면을 비교한 것이다.

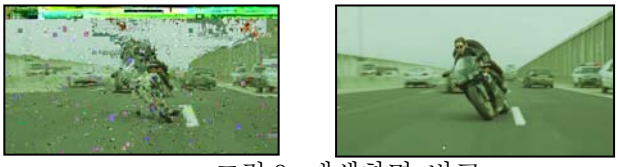

그림 8. 재생화면 비교

4.4. DRM 모듈과의 연동

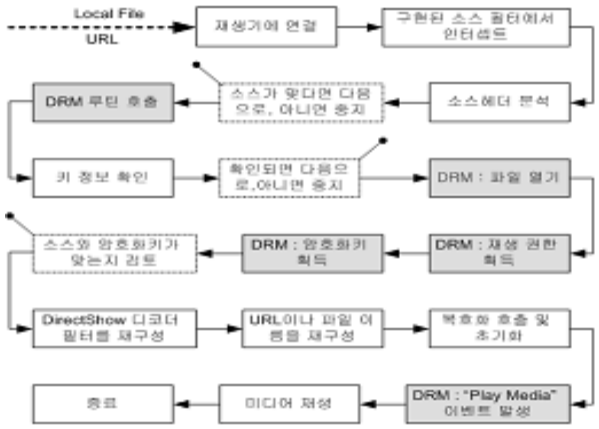

그림 9. 클라이언트 흐름도

DRM 모듈은 DRM 서버와의 통신을 통하여 DRM 관련 정보들을 주고 받으며, 중요한 데이터들을 관리한다. 또한 콘텐츠 재생기를 제어하여 사용자가 획득한 권리에 따라 올바르게 콘텐츠가 사용될 수 있도록 한다. 특히, 사용자로부터 발생할 수 있는 원본 콘텐츠의 누출을 방지할 수 있도록 고도의 보안 메커니즘이 적용되어야 한다.

스트리밍 시스템과 DRM 시스템이 연동하기 위해서 복호화 모듈은 DRM 이 제공하는 API 에 맞추어 개발되어야 하며, 이 인터페이스를 통하여 원하는 정보를 주고 받으며 서로 연동하게 된다.

5. 결론

본 논문은 ASF 파일 포맷의 분석을 통해서, 파일 구조를 변경시키지 않고 ASF 데이터 자체를 보호하는 방안과 이를 클라이언트에서 재생하기 위한 구조를 설계하고 구현하였다. 복호화에 필요한 정보는 컨텐츠의 암호화와 동시에 DRM 서버로 전송되며, DRM 서버는 암호화 메타데이터에 복호화키를 포함한 DRM 정보를 추가하여 인증된 사용자에게 전송하면, 사용자는 복호화 키를 이용 데이터를 복호화한 후 서비스를 이용한다.

본 연구의 결과를 토대로 다양한 스트리밍용 매체에 대한 연구와 높은 성능을 제공하기 위한 연구가 가능하다. 향후에는 스트리밍 서비스의 성능을 증가시키기 위하여, 암/복호화 기술의 부하 방지, 결함 감지, 복구 및 보다 안전하고 효율적인 분산 키 관리 시스템 연구가 수행될 것이다.

참고문헌

[1] ASF Specification, Microsoft.
[2] Data Encryption Standard(DES), FIPS Publication 46-2 1993.
[3] DirectShow, http://www.microsoft.com/Developer/PRODINFO/directx/dxm/help/ds/c-frame.htm#default.htm.
[4] DirectX SDK, http://www.microsoft.com/directx.
[5] DMO, http://msdn.microsoft.com.
[6] http://www.microsfot.com/windows/windowsmedia.
[7] Joshua Duhl, Susan Kevorkian, Understanding DRM Systems, IDC 2001.
[8] Kenneth Louis Milsted, Automated Method and Apparatus to Package Digital Content for Electronic Distribution using the Identity of the Source Content, United States Patent 6,345,256.
[9] Olin Sibert, DigiBox: A Self-Protecting Container for Information Commerce, 1st USENIX Workshop on Electronic Commerce, 1995.
[10] Seongoun Hwang, Kisong Yoon, Changsoon Park, “Design and Implementation of a Licensing Architecture for Distribution of Copyright-Protected Digital Contents,” Telecommunications Review, Vol.12, No.5, October, 2002.