디지털 방송 서비스를 위한 AXMEDIS 암호화 툴 전송 모듈 설계 및 구현

황경민* · 안상우** · 이주영** · 남제호** · 홍진우** · 정회경*

*배재대학교 컴퓨터공학과 · **한국전자통신연구원

The Design and Implement of Transport Module for Digital Broadcasting Service

Kyung-min Hwang* · Seng-woo Ahn** · Joo-Young Lee** · Je-ho Nam** · Jin-woo Hong** · Hoe-Kyung Jung*

*Dept. of Computer Engineering, Paichai University · **ETRI

E-mail : *{koukyo, hkjung}@pcu.ac.kr, **{asw, leejy1003, namjeho, jwhong}@etri.re.kr

요 약

디지털 콘텐츠 시장의 성장은 다양한 콘텐츠 소비 장치의 개발을 유도하였고, 이는 디지털 콘텐츠의 소비를 더욱 촉진시켰다. AXMEDIS(Automating Production of Cross Media Content for Multichannel Distribution)는 다양한 플랫폼에서의 디지털 콘텐츠 소비를 지원하기 위해 AXMEDIS 프레임워크를 통한 다양한 디지털 방송 솔루션을 제공하고 있다. AXMEDIS 프레임워크는 단말에서의 디지털 콘텐츠 보호 및 소비를 위해 암호화 툴을 사용한다. 그러나 단말간의 유동적인 암호화 툴의 전송이 불가능하여 단말에서의 암호화 툴 관리가 불가능하다.

이에 본 논문에서는 AMEDIS 프레임워크에서 디지털 방송 콘텐츠의 보호와 소비를 유동적으로 운용 가능하도록 AXMEDIS 암호화 툴 전송 모듈을 설계 및 구현하였다.

ABSTRACT

The growth of digital content market inducted to develope consuming device of various content, and through this, digital content consuming is more Promoted. AXMEDIS(Automating Production of Cross Media Content for Multichannel Distribution) offer various digital broadcasting solution via AXMEDIS Framework for supporting consumption of digital content on various platform. AXMEDIS Framework use cryption tool for protecting and consuming of digital content on device. But, cryption tool management is impossible on device, because of flexible transfer of cryption tool can't be supported between device. In this paper, we designed and implemented transfer module of AXMEDIS cryption tool for flexible operating protection and consumption of digital program on AXMEDIS Framework.

키워드

AXMEDIS, MPEG-21, 디지털 방송

Ⅰ. 서 론

멀티미디어 영상 매체 기술의 발달로 디지털 콘텐츠가 보급됨에 따라 장소에 구애 받지 않고 디지털 콘텐츠를 소비할 수 있게 되었다. PC, 셋탑박스 등에서만 소비가 가능했던 디지털 콘텐츠는 최근 각종 단말기기 성능의 발달로 인해 다양한 단말 기기에서의 소비가 가능해졌다. 그러나 디지털 콘텐츠는 불법복제와 인터넷을 통한 무분별한 콘텐츠의 유통에 취약점을 가지고 있으며 이는 디지털 콘텐츠 시장의 질서를 어지럽히고 있다. 또한 디지털 콘텐츠 제작자의 수익감소를

야기하여 양질의 디지털 콘텐츠 제작의욕을 상실케 할 뿐 아니라 디지털 콘텐츠 유통사업을 근본적으로 위협하고 있어 디지털 콘텐츠 시장의 불경기를 가져오는 악순환의 고리로 작용하고 있다. 이를 해결하기 위해 국내외 DRM(Digital Right Management) 개발 업체들은 디지털 콘텐츠를 보호할 수 있는 DRM 툴을 개발하였지만, 제작업체들간의 DRM 상호운용성 부재로 또다른 문제를 야기하였고 이는 국제 표준화 단체인 MPEG (Moving Picture Experts Group)에서 디지털 콘텐츠를 유통하기 위한 MPEG-21 Multimedia Frameworks를 표준화하기에 이르렀다. 표준화된 MPEG-21 Multimedia Frameworks의 IPMP (Intellectual Property Management and Protection) 기술은 각 회사들 간의 DRM 표준으로 정립되어 사용되고 있으며 디지털 콘텐츠의 무분별한 복제 및 불법 유통으로부터 디지털 콘텐츠 제작자, 유통업자, 소비자들의 권익을 보호하는 기술로 자리 잡고 있다.

현재 디지털 콘텐츠의 시장이 PC, 셋탑박스 등에 국한되지 않고 각종 단말 기기에까지 영역을 확대하고 있지만 현재 DRM 기술은 단말 기기 외부에서만 지원되고 있는 실정이다. 이에 플랫폼 독립적으로 사용 가능한 디지털 콘텐츠 보호관리 서비스 기술이 필요하며 해당 서비스를 이기종 단말간의 상호운용이 가능하도록 지원하는 메시지 프로토콜의 표준화가 필요하다.

이에 본 논문에서는 단말 환경에서 디지털 콘텐츠의 보호관리가 가능하도록 하기 위해 단말 상에서의 작동 가능한 보호관리 툴 적용 모듈과 해당 보호관리 모듈을 통합 관리하는 서버 시스템을 설계 및 구현하였다.

II. 관련연구

2.2 MPEG-21 IPMP(intellectual Property Management and Protection)

MPEG-21 IPMP는 MPEG-21 Multimedia Frameworks의 제 4부 규격으로 디지털 아이템이 네트워크 상에서 생성, 변형, 전달, 소비 단계를 거치는 과정에서 디지털 아이템을 안전하게 취급하여 외부의 위험 요소로부터 보호하는 것이다. 그리고 디지털 아이템이 다양한 종류의 네트워크 및 단말기로 처리되는 동안 사용자들에게 저작권과 디지털 아이템에 대하여 동의를 표현하고, 라이프 사이클이 소멸되기 전까지 지속적으로 안전성과 확신성을 제공한다.

이 분야는 암호화 알고리즘, 키, 키 관리 등의 IPMP 툴의 검색 방법과 툴 간의 메시지 교환 및 툴과 터미널 간의 메시지 교환 방법을 표준화의 대상으로 하고 있다. 특히 IPMP와 관련되어 표준화가 함께 진행되고 있는 세부 분야로는 저작권 표현 언어(REL)와 저작권 사전(RDD : Rights Data Dictionary)이 있다[2,3].

MPEG-21 IPMP Components 스키마 구조는 디지털 아이템 선언의 구조에 따라 DID(Digital Item Declaration) 규격에 IPMP 요소를 포함하는 확장된 개념으로 기술되도록 규격화 하였다.

2.3 MPEG-21 REL(Rights Expression Language)

REL은 RDD와 함께 MPEG-21 지적 재산권 관리 및 보호(IPMP)의 세부 요소로 분류 된다. REL은 MPEG-21 Multimedia Frameworks 내에서 디지털 콘텐츠 이용, 유통, 관리 및 사용 규칙 등에 관한 표현 언어로 저작권 처리 관련 용어에 대하여 신뢰도 높은 시스템을 제시한다. 또한 표준화된 용어를 제공함으로 타 시스템간의 상호 운용성의 증대 및 유연성과 함께 확장성을 제공하는 것을 목표로 한다. REL의 스키마는 REL Core, REL Standard Extension, REL Multimedia Extension로 구성된다. REL Core에서는 루트 엘리먼트 License를 비롯한 핵심 요소 및 REL 전체의 개념 정의 등을 포함하고 있고, REL Standard Extension 부분은 REL 소비에서 일반적이고도 광범위하게 사용할 수 있는 정보들을 정의 하고 있다. REL Multimedia Extension 부분에서는 멀티미디어 자원에 대한 사용, 삭제 및 수정 등의 자원에 대한 확장 처리 정보가 정의되어 있다. 각각은 필요한 경우 REL Core를 확장하여 사용할 수 있다. 하위 계층에서 RDD가 REL의 용어를 정확하게 정의하고 각각의 의미를 제공하게 되며 본 논문에서는 REL Core만을 제한적으로 사용한다[4,5].

III. 시스템 설계

본 논문에서는 보호관리 툴 서버에서 툴을 통합 관리하며, 보호관리 툴 클라이언트의 보호관리 툴 적용 모듈로부터 툴 전송을 요청받아 툴 전송 XML 메시지를 정의하여 툴을 전송하도록 설계하였다. 전체 시스템 구조는 그림 1과 같다.

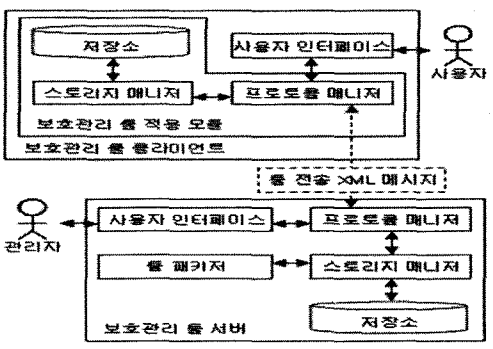

그림 1. 보호관리 툴 전송 모듈 및 보호관리 툴 서버 전체 아키텍처

3.1 보호관리 툴 서버/클라이언트

보호관리 툴 서버는 관리자로부터 툴을 등록/수정/삭제 등 툴 관리를 위한 인터페이스를 제공하며 보호관리 적용 모듈에서의 툴 요청에 따라 툴을 탐색하여 패키징된 툴을 전송한다. 보호관리 툴 서버는 사용자 인터페이스, 스토리지 매니저, 프로토콜 매니저, 툴 패키저 4개 부분으로 구성되어 있다.

사용자 인터페이스는 관리자가 직접적으로 서버에 접촉할 수 있는 통로로서 사용자 인터페이스를 통해 관리자는 툴 등록/수정/삭제 등 툴에 대한 관리가 가능하며, 서버 운용을 위한 구동 관련 작업 수행이 가능하다. 프로토콜 매니저는 클라이언트 접속 모듈을 통해 외부의 클라이언트들과의 통신에 관련한 모든 작업을 담당하며 클라이언트 관리 모듈은 접속된 모든 클라이언트들에 대한 접속을 관리한다. 스토리지 매니저는 툴 패키저로부터 병합된 XML 메시지를 툴 저장소에 저장하는 역할을 담당하며, 이외에 툴에 대한 탐색, 삭제, 그리고 관리자의 접근 인증을 위한 작업을 수행한다. XML 메시지는 툴 저장소에 저장 시 보다 원활한 툴 탐색을 위한 구조로 저장된다. 이벤트 로그 매니저는 서버에서 발생되는 모든 로그 기록을 저장하며 실시간으로 발생된 모든 이벤트를 출력하여 관리자에게 현재 서버의 상태를 통지한다. 툴 저장소는 관리자 로그인에 필요한 관리자의 계정과 비밀번호를 저장하고 있으며, 인증된 관리자로부터 등록되는 모든 툴을 보유한다. 외부로부터 툴에 대한 탐색 요청 시 해당 툴 ID를 탐색하여 툴을 반환하며 불필요한 툴을 관리자의 권한으로 삭제가 가능하다.

보호관리 툴 클라이언트는 서버로부터 툴 정보를 수신하여 필요한 툴을 요청 및 다운로드하는 역할을 담당한다. 특히 보호관리 툴 적용 모듈은 프로토콜 매니저 부분으로서 보호관리 툴 서버와의 툴 전송을 위한 툴 전송 XML 메시지를 생성 및 파싱하여 통신하며 내부 설계 구조는 보호관리 툴 서버에서 설계된 프로토콜 매니저와 동일하다.

3.2 툴 전송 XML 메시지 정의

툴 전송 XML 메시지는 보호관리 툴 서버와 툴 적용 모듈 간의 통신에 필요한 메시지를 고려하여 GetToolList, GetToolListResponse, GetTools, GetToolsResponse 총 4가지로 나누어 정의하였다. GetToolList 엘리먼트는 서버측에 보유중인 툴 목록을 툴 적용 모듈에서 확인하기 위해 요청하는 메시지로서 보호관리 툴 서버의 툴 저장소에 저장되어 있는 툴의 툴ID 리스트를 수신하기 위해 전송한다. GetToolListResponse 엘리먼트는 보호관리 툴 서버에서 GetToolList 메시지를 수신 시 생성하는 메시지로서 보호관리 툴 서버의 툴 저장소에 보유중인 툴의 툴ID 들의 목록을 리스트로 작성하여 생성한다. 툴ID는 MPEG-21 DII(Digital Item Identification)의 표준에 따라 URN(Uniform Resource Name)으로 구성되며 툴의 정보를 명확히 파악할 수 있도록 툴에 대한 이름과 설명을 Description 엘리먼트를 통해 제공한다. GetTools 엘리먼트는 디지털 콘텐츠 소비에 필요한 보호관리 툴의 툴ID를 구성하여 보호관리 툴 서버로 툴을 요청하기 위한 전송 메시지이다. 필요한 툴을 확인 후 해당 툴의 툴ID를 리스트로 작성하여 서버로 전송하기 위해 사용된다. GetToolsResponse 엘리먼트는 보호관리 툴 서버 측에서 GetTools 메시지를 수신 시 생성되는 메시지이다. 툴 저장소에 툴 패키지 형태로 저장된 XML 메시지들을 탐색하여 요청된 툴의 툴ID와 일치하는 툴 패키지를 조합하여 생성한다.

Ⅳ. 시스템 구현

본 시스템을 구현하기 위해 IBM-PC의 Windows XP 상에서 Visual Studio .NET 2003 MFC, MSXML 4.0, MySQL 5.0을 사용하였다.

4.1 보호관리 툴 서버 인터페이스

보호관리 툴 서버의 인터페이스는 관리자 인증부, 서버 구동 제어부, 툴 관리부, 로그 기록부로 나누어 구성하였다. 구현된 보호관리 툴 서버의 전체 구성은 그림 2와 같다.

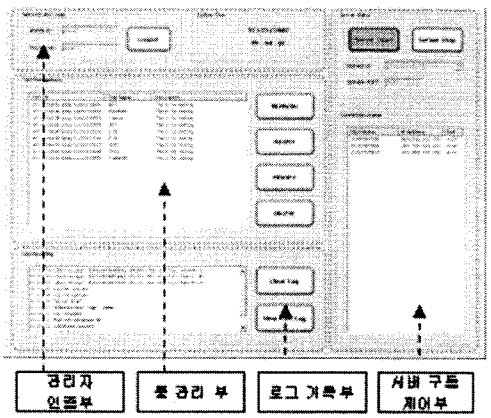

그림 2. 보호관리 툴 서버 인터페이스

관리자 인증부는 관리자 권한 판별 인증과정을 수행하기 위한 인터페이스이다. 인증에 성공 또는 실패하면 해당 기록이 로그 기록부에 출력됨과 동시에 로그 파일로 기록된다. 서버 구동 제어부는 보호관리 툴 서버를 제어하는 인터페이스를 제공한다. 서버가 구동 시 접속된 클라이언트를 "Connection Status" 창에서 확인 가능하다. 툴 관리부는 권한 인증을 통과한 관리자에게 툴을 등록/수정/삭제하기 위한 인터페이스를 제공한다. 보유 툴 리스트를 확인하는 리스트컨트롤 박스, 툴 목록 갱신을 위한 "Refresh"버튼, 툴 등록

을 위한 "Regist"버튼, 툴 수정을 위한 "Modify" 버튼, 툴 삭제를 위한 "Delete"버튼으로 구성되어 있다. 툴 등록/수정 시 툴 등록/수정 다이얼로그가 생성되며 각 툴에 대한 정보를 기입후 "Submit"버튼을 클릭하여 툴 저장소에 툴 패키지를 저장한다. 로그 기록부는 서버에서 발생하는 모든 이벤트 기록을 관리자에게 통지할 수 있는 로그 리스트 박스, 로그 리스트 박스를 초기화하는 "Clear Log"버튼, 지난 로그기록의 열람을 위한 "View past Log"버튼으로 구성되어 있다.

4.2 보호관리 툴 클라이언트 인터페이스

구현된 보호관리 적용 모듈을 적용한 보호관리 툴 클라이언트를 구현하였으며 구현된 보호관리 툴 클라이언트는 그림 3과 같다.

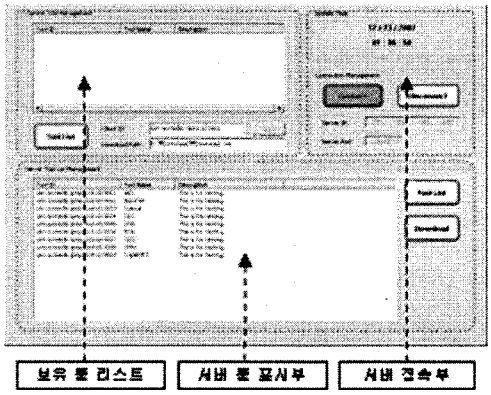

그림 3. 보호관리 툴 클라이언트 인터페이스

보유 툴 리스트는 사용자에게 현재 보유중인 툴의 정보를 출력하기 위한 인터페이스로서 현재 클라이언트에서 보유중인 툴 리스트를 출력하기 위한 리스트 컨트롤 박스와 보유 툴 리스트를 갱신하기 위한 "Tool List" 버튼으로 구성되어 있다. 사용자는 해당 인터페이스를 통해 보호관리 툴의 보유 현황을 알 수 있으며, 서버와의 접속 시 서버에 클라이언트의 고유 ID를 통보하기 위한 클라이언트ID 입력창과 툴 다운로드 시 저장될 경로를 지정하는 "Download Path" 입력창으로 구성되어 있다. 서버 접속부는 서버에 접속하기 위한 인터페이스를 제공하며 접속할 서버의 IP와 포트를 입력하기 위한 "ServerIP" IP 주소 입력창과 "Server Port" 입력창으로 구성하였다. "Connect" 버튼을 클릭하여 서버에 접속이 가능하며, 서버와의 접속이 성공하면 서버 툴 표시부의 버튼들이 활성화된다. 서버와의 접속을 종료하기 위해서는 "Disconnect" 버튼을 클릭한다. 서버 툴 표시부는 서버에서 보유중인 보호관리 툴의 목록을 출력하며 이를 선택하여 다운로드를 가능하게 하는 인터페이스이다. 서버에 보유중인 보호관리 툴 리스트를 요청하기 위해서는 서버 접속부를 통해 서버와 접속이 선행되어야 한다. 서버와 접속된 상태에서 "Tool List"을 클릭하면 서버에서 보유중인 보호관리 툴이 왼쪽의 리스트 컨트롤창에 출력된다. 출력된 보호관리 툴 목록을 선택하여 "Download"버튼을 클릭하면 서버로 지정된 보호관리 툴의 다운로드 요청 메시지가 전달되어 서버로부터 해당 툴을 다운로드 받아 보유툴 리스트에서 설정된 "Download Path"의 경로로 지정된 폴더에 저장된다.

V. 결 론

본 논문은 단말 디지털 콘텐츠 보호관리 모듈 적용 및 상호 운용성 확보를 위한 것으로 툴 전송 XML 메시지를 정의 및 이를 적용한 디지털 콘텐츠 보호관리 적용 모듈과 보호관리 툴 서버 참조 모델을 설계 및 구현하였다.

기존의 보호관리 모듈은 단말 외부에서만 적용되는 한계를 가지고 있어 단말상에서의 보호관리 모듈의 효율적이고 상호운용 가능한 툴의 관리에 취약점을 가지고 있다. 이러한 문제를 해결하기 위해 단말의 보호관리 적용 모듈과 보호관리 툴 서버간의 상호운용 가능한 메시지 표준을 정의할 필요가 있다. 이는 모든 보호관리 툴의 전송이 표준을 준수하도록 유도하며, 툴 전송 XML 메시지를 통해 프로토콜의 제약 없이 다양한 보호관리 툴의 전송이 가능하다.

이에 본 논문에서는 상호운용 가능한 보호관리 툴 전송 XML 메시지 정의를 위해 국제 표준단체인 MPEG에서 정의한 MPEG-21 IPMP, REL, DII 스키마 구조를 상속받아 메시지를 정의하였으며, 이를 통해 이기종 단말간의 모든 보호관리 툴 전송 상호운용성을 확보하였다. 본 논문을 통해 이기종 단말과 보호관리 툴 서버간의 보호관리 툴 전송 상호운용성이 확보될 것으로 기대되며 단말기기에 운용 가능한 보호관리 툴의 개발 및 적용을 통해 단말에서의 디지털 콘텐츠 보호의 개념이 확장될 것으로 사료된다.

참고문헌

[1] 김해광, "MEPG-21 멀티미디어 프레임워크", 한국정보통신기술협회, TTA 저널통권 82호, 2003

[2] Rob Koenen, "IPMP in MPEG Standards.", Workshop on DRM for the Web, W3C, INRIA - Sophia Antipolis, France, 22-23. Jan. 2001.

[3] "ISO/IEC 21000-4 FCD IPMP Component ", ISO/IEC/JTC1/SG29/WG11/N7196, MPEG MDS Group, April 2005.

[4] "REL", ISO-IEC_21000-5_(E)_FDIS

[5] "RDD", ISO-IEC JTCl_SC29_M105 74