UPnP 클라우드 기반 콘텐츠 공유

박용석* · 김현식* · 정다희*

*전자부품연구원

Contents Sharing based on UPnP Cloud

Yong-Suk Park* · Hyun-Sik Kim* · Dahee Jeong*

*Korea Electronics Technology Institute

E-mail : yspark@keti.re.kr

요 약

최근 UPnP(Universal Plug and Play) 포럼은 인터넷을 통해 기기 간 연결을 가능하게 하는 UPnP 클라우드(Cloud) 규격을 제정하여 기존 UPnP 규격의 기능과 범위를 확장시켰다. UPnP 클라우드는 XMPP(Extensible Messaging and Presence Protocol)를 이용하여 서로 다른 도메인에 위치하는 UPnP 기기들을 연결시켜준다. 본 논문에서는 UPnP 클라우드 규격의 세부내용을 소개하고 UPnP 클라우드가 어떻게 원격으로 콘텐츠를 공유할 수 있는지 그리고 DLNA(Digital Living Network Alliance)와 연계될 수 있는지 알아본다.

ABSTRACT

Recently, the Univeral Plug and Play (UPnP) Form has relased UPnP Cloud, which is an extension to the basic UPnP architecture that enables device to device connectivity across the Internet. UPnP Cloud uses the Extensible Messaging and Presence Protocol (XMPP) to connect devices located in different domains. In this paper, the details of the UPnP Cloud are presented. We demonstrate how UPnP Cloud enables contents to be shared remotely, and how it can be used in connection with DLNA.

키워드

UPnP, 클라우드 서비스, 콘텐츠 공유, DLNA

Ⅰ. 서 론

스마트 기기의 활성화와 함께 각종 가전기기, 센서장치 등에 네트워크 기능이 추가되면서 점점 더 많은 수의 장치들이 인터넷을 통해 연결되고 있다. 이로 인해 클라우드(cloud) 컴퓨팅과 사물 인터넷(IoT – Internet of Things)이 많은 주목을 받고 있으며 이를 지원하기 위한 다양한 표준규격이 생성되고 있다. 이에 대응하기 위해 최근 UPnP(Universal Plug and Play) 포럼은 인터넷을 통해 기기 간 연결을 가능하게 하는 UPnP 클라우드 규격을 제정하여 기존 UPnP 규격의 기능과 범위를 확장시켰다. UPnP 클라우드는 XMPP(Extensible Messaging and Presence Protocol)를 이용하여 서로 다른 도메인에 위치하는 UPnP 기기들을 연결시켜주거나 클라우드에 존재하는 가상장치와 연결할 수 있다. 본 논문에서는 UPnP 클라우드 규격의 세부내용을 소개하고 UPnP 클라우드를 이용한 콘텐츠를 공유 방법과 이것이 현재 홈 네트워크에서 미디어 공유에 가장 많이 사용되고 있는 DLNA(Digital Living Network Alliance) 가이드라인과 어떻게 연계될 수 있는지 알아본다.

Ⅱ. UPnP 클라우드 개요

UPnP 클라우드 기술은 기본 UPnP 구조(architecture)에 대한 확장이며 인터넷 또는 클라우드를 통한 기기 간의 연결(connectivity)을 가능하게 한다. UPnP 클라우드는 UPnP UDA(UPnP

Device Architecture) 2.0 규격의 부록(annex)에 포함되어있으며 UCA(UPnP Cloud Annex)라고 한다[1]. 기본적으로 UPnP의 Addressing, Discovery, Description, Eventing 그리고 Control 관련 프로토콜을 XMPP 표준규격과 바인딩(binding)하여 클라우드 기능을 제공한다. 바인딩은 UPnP device 또는 control point에 wrapper 형태로 추가할 수 있는 XMPP 프로파일의 정의를 통해 이루어진다. 이로 인하여 UPnP device는 UCCD(UPnP Cloud Capable Device)가 되고 control point는 UCC-CP(UPnP Cloud Capable Control Point)가 된다. 이들은 XMPP 서버(UCS - UPnP Cloud Server)를 통해 다른 기기와 control point에 연결할 수 있게 된다. 즉, UCA는 XMPP를 이용하며 모든 DCP(Device Control Protocol)를 클라우드 환경에서 동작하게 할 수 있게 한다.

Ⅲ. UPnP 클라우드 프로토콜 구조

그림 1은 UDA와 UCA의 프로토콜 구조를 비교하여 보여준다. UDA와는 달리 UCA의 통신은 모두 TCP/IP를 통해 이루어지기 때문에 unicast 이며 전송 계층 보안(TLS - Transport Layer Security, SASL - Simple Authentication and Security Layer)이 적용되어 신뢰성이 높다[2]. 따라서 UCA에서는 UDP/IP 관련 요소들이 없다. 또한 SOAP(Simple Object Access Protocol)과 GENA(Generic Event Notification Architecture) 요소 관련 전송은 HTTP 대신 XMPP를 통해 이루어진다.

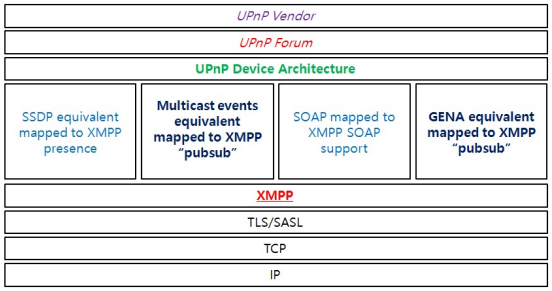

그림 1. UCA의 프로토콜 구조

Ⅳ. UPnP 클라우드 통신 경로

UCA는 두 가지 통신 방법을 제공한다. 첫째는 동일한 사용자(user) 소유의 UCCD와 UCC-CP들이 동일한 계정(account)에 등록되어 서로 통신하는 것이다. 두 번째는 서로 다른 사용자 소유의 UCCD와 UCC-CP들이 서로 다른 계정과 서버에 등록되어 통신하는 것이다. 이와 관련하여 그림 2와 같이 기기(device), control point, 서버, 서비스, 사용자 간 총 8개의 메시지 통신 경로가 생성될 수 있다.

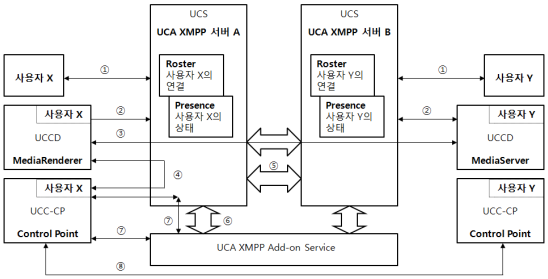

그림 2. 메시지 통신 경로

첫 번째 통신 경로는 사용자와 UCS간의 통신으로 계정 관리를 위해 기 정의된 UCA API를 사용한다. 두 번째 통신 경로는 UCA 프로토콜을 사용하여 UCC-CP 또는 UCCD가 UCS와 통신하는 것이다. 세 번째 통신 경로는 UCA 프로토콜을 사용하여 사용자 X의 크리덴셜(credential)을 가진 UCC-CP 또는 UCCD가 사용자 Y의 크리덴셜을 가진 UCC-CP 또는 UCCD와 통신을 하는 것으로 동일한 UCS 또는 두 개의 다른 UCS를 이용하는 것이다. 네 번째 경로는 같은 UCS에 있는 동일 사용자의 크리덴셜을 가진 UCC-CP와 UCCD 간의 통신이다. 다섯 번째 경로는 XMPP 프로토콜을 이용한 두 UCS 간의 직접적인 통신이다. 여섯 번째는 UCS와 PubSub 서비스 또는 MUC(Multi-User Chat)과 같은 UCA add-on 서비스와의 통신이다. 일곱 번째는 UCC-CP 또는 UCCD와 UCA add-on 서비스와의 직접 통신이다. 마지막은 UCA 프로토콜을 통해 직접 통신을 설정한 다음 사용자 X의 크리덴셜을 가진 UCC-CP 또는 UCCD가 사용자 Y의 크리덴셜을 가진 UCC-CP 또는 UCCD와 직접 데이터 전송을 하는 것이다.

Ⅴ. 클라우드를 통한 상호 작용

UCA의 기본적인 동작은 다음과 같다. 사용자는 원하는 UCS에 계정을 생성하고 UCC-CP 또는 UCCD를 UCS에 등록한다[3]. 이는 UDA에서의 Addressing과 유사하다. 등록된 UCCD 또는 UCC-CP는 XMPP <presence> stanza를 발급하여 자신의 상태(available, unavailable)를 알린다[4]. 이는 UPnP SSDP(Simple Service Discovery Protocol) 메시지와 유사하다. UCCD는 자신의 기기 타입 정보를 명칭과 버전 정보와 함께 XMPP <presence> stanza를 통해 UCC-CP에게 제공하고 <iq> stanza를 이용하여 기능(description) 정보를 제공한다. 이것은 UPnP Description과 유사하다. 이벤트 생성 및 관리를 위해 UCCD는 UCA PubSub collection을 생성하고 관리한다. 이벤트

가 존재할 경우 UCC-CP는 해당 collection의 node에 subscribe한다. UCC-CP는 SOAP 메시지 교환을 통해 UCCD에게 제어명령을 내리게 된다. 이때 SOAP는 XMPP에 바인딩 되게 된다.

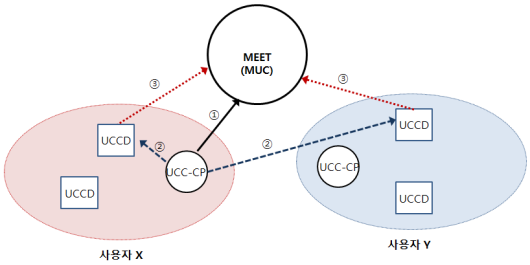

그림 3. 클라우드를 통한 상호 작용 예

기기, 서비스 간 상호작용은 그림 3과 같다. 사용자 X는 UCS를 통해 방을 생성한다(MUC). 이것은 채팅방을 만드는 것과 유사하다. 사용자 X는 자신과 타 사용자(사용자 Y) 소유의 UUCD와 UCC-CP들을 생성된 방으로 초대한다. 크리덴셜 확인 후 생성된 방에 초대된 기기들이 모이게 되면 사용자 X와 Y는 허락된 정보와 콘텐츠를 공유할 수 있게 된다.

DLNA 규격의 기기들도 현재 단일 도메인 내에서 동작되도록 구성되어있지만, UPnP 클라우드가 UHOD(UPnP Home Only Device)와 같은 legacy 장치를 지원하기 때문에 현재 UCA에 쉽게 통합될 수 있다..

Ⅵ. 결 론

UPnP 클라우드는 모든 UPnP 규격을 클라우드화 시킬 수 있는 해법을 제시한다. 검증된 XMPP 프로토콜을 사용함으로써 UPnP 기능을 클라우드로 확장시킴과 동시에 안정성과 빠른 구현의 용이성을 제공한다. UPnP와 XMPP 생태계 간 연동함으로서 시너지 효과로 인한 새로운 서비스 가능성의 장을 열 수 있게 되었다. UPnP 클라우드는 IoT 지원을 주 목표로 하고 있으나 TCP/IP의 사용은 일부 자원제약이 심한 IoT 환경에서 사용하기에 적합하지 않을 수 있다. UPnP 프로토콜이 DLNA와 같은 홈 네트워크 환경에서 널리 사용되고 있는 만큼 향후 IoT에 어떤 영향을 미칠지 귀추가 주목된다.

Acknowledgement

본 연구는 미래창조과학부 및 정보통신기술진흥센터의 산업융합원천기술개발사업(SW·컴퓨팅)의 일환으로 수행하였음. [10041771, DLNA(스마트 기기간 콘텐츠공유 규격) 자동 시험 인증 소프트웨어 개발]

참고문헌

[1] UPnP Forum, "UPnP Device Architecture 2.0", UPnP Specification, 2014년 9월
[2] IETF, "Extensible Messaging and Presence Protocol XMPP: Core", RFC6120, 2011년 3월
[3] IETF, "Extensible Messaging and Presence Protocol XMPP: Address Format", RFC6122, 2011년 3월
[4] IETF, "Extensible Messaging and Presence Protocol XMPP: Instance Messaging and Presence", RFC6121, 2011년 3월