JDCS
디지털콘텐츠학회논문지
Journal of Digital Contents Society
Vol. 18, No. 8, pp. 1523-1527, Dec. 2017

폐쇄망 환경에서 1인 미디어 라이브 시스템 구현

김 대 진
동국대학교 영상문화콘텐츠연구원

Implementation of One-Person Media Live System in Closed Network Environment

Dae-Jin Kim
Research Institute for Image & Cultural Contents, Dongguk University, Seoul 04626, Korea

[요 약]

최근 사용자가 직접 촬영을 하여 콘텐츠를 만들고, 이 데이터를 인터넷을 통해 다수에게 방송하는 서비스들이 점점 많아지고 있다. 1인 미디어 라이브 서비스는 생활 속에서 인기를 끌고 , 비슷한 서비스들이 일반화 되고 있다. 이러한 서비스는 대부분 일반 사용자를 대상으로 한다. 그러나 서비스 모델이나 상황에 따라, 특히 보안에 민감한 서비스 환경에서는 공유의 대상이 제한이 되는 경우들이 있다. 특히 군대, 관공서, 보안기관등 폐쇄망 환경에 있는 경우, 이들만의 특별화 된 서비스 구성이 필요하다. 따라서 본 논문에서는 폐쇄망 환경에서 1인 미디어 라이브 시스템을 구현하였다. 이를 통해 서비스 환경에 적합한 시스템 구성 요소를 알아보고, 이들을 어떻게 개발해야 되는지 연구하였다.

[Abstract]

Recently, there have been a growing number of services for users to shoot content by themselves and to broadcast this data to a large number through the Internet. One-person media live services are popular in life, and similar services are becoming common. Most of these services are for general users. However, depending on the service model or the situation, there are cases where the object of streaming is limited depending on the security-sensitive service environment. Especially, in the case of closed networks such as military, government, and security organizations, these specialized services are needed. Therefore, in this paper, we implemented a live media system in a closed network environment. We study system components suitable for service environment and study how to develop components.

색인어 : 1인 미디어, 라이브 서비스, 폐쇄망, 미디어 시스템

Key word : One-person media, Live services, Closed Network, Media System

http://dx.doi.org/10.9728/dcs.2017.18.8.1523

Received 30 November 2017; **Revised** 15 December 2017
Accepted 25 December 2017

***Corresponding Author: Dae-Jin Kim**

Tel: 02-2264-5804
E-mail: sampoo00@dongguk.edu

Ⅰ. Introduction

Recently, One-person media service is activated, according as the personalities of individuals have become important, and more and more people are trying to share their advantages, information, and desires with others, and technologically various platforms are emerging. In particular, the generalization of smartphones has accelerated the phenomenon[1].

In this interest, media live broadcasting is being studied. Tran Thi Thu Ha proposed Hybrid model combining CDN(Content Delivery Network) and P2P[2], and Kwon has implemented a broadcasting system such as IPTV in a cloud environment, resulting in cost savings[3]. Karine Pires analyzed user usage pattern by comparing YouTube and Twitch among one-person media broadcasting system and studied basic data of difference and service direction[4]. Fei Chen has studied how to bring data from the Cloud to live streaming[5]. Metthew k. Mukerjee has studied how to control CDN-based live broadcasting in real time through center control[6]. Under these various circumstances, studies for media live broadcasting are continuing.

But, there are cases where the subject of sharing is restricted depending on the service environment. Especially, if you are in a specific environment such as military, government, and security organizations, you need special configuration of these services.

Therefore, this paper proposes one-person live media system in a closed network environment. In chapter2, we discuss the AWS media live system, which is the most representative in open network environment, and discuss the limitations of this system. In chapter3, we design a one-media live system in closed network environment. In chapter4, we show implementation environment and prototype for result. Finally, we concluded about this system in chapter5.

Ⅱ. Live System in Open Network

2-1 Whole System

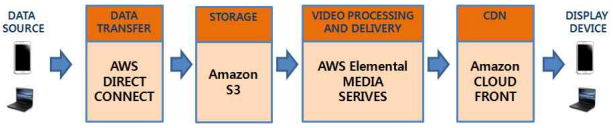

Fig. 1. Live broadcasting system using AWS[7]

Figure 1 shows the configuration of a live broadcasting system using AWS. AWS is the most representative Content Delivery Network(CDN) company and is the most influential company in the world in network services. Most CDN companies also have similar configurations when configuring live broadcast systems. In AWS, when an object is created in S3 and data is transferred in S3, the user can perform content service at the user end. In general, since streaming is performed using storage, HLS (HTTP Live Streaming) is a basic service that segments streaming data and transfers segments through web server. Due to the nature of live, RTMP (Real Time Messaging Protocol) is also supported to reduce latency. And VOD storage is also possible, so the saved contents can be seen in the display device. The biggest advantage of this system is that you can build live services using CDN for a small cost. CDN allows servers in multinational and large IDC centers to be accessed. So, we can reduce network latency by accessing the IDC, which is the most user accessible easily. This is possible because AWS is a global company. In addition to AWS, CDN companies such as CDNetwork, Limelight, and Akamai offer open network services.

Open network makes it easier to build a service, but if you are in a service where information leakage issues such as military, government, and security are important, you should use a closed network. In this case, it is impossible to service using the CDN. Therefore, in this paper, we implement a one-person media live system in a closed network and design a broadcasting system that can be used in a restricted environment.

Ⅲ. Live System in Closed Network

2-1 Whole System

Figure 2 shows one-person live media system configuration in a closed network environment. The overall configuration is divided into four major components (Streaming Server, Common Management, Live Management, VOD Management). The main functions of each components are as follows.

1) Streaming Server

(1) RTMP Server

we can receive pushing streaming data from Encoder User in real time. At this time, the input stream is broadcast in RTMP format to minimize the delay time. The user should be able to watch within 3 seconds.

(2) HLS Server

we can receive pushing streaming data from RTMP Server in real time. At this time, the input stream is broadcast in HLS format. In the case of HLS, the segmentation attribute must be adjustable because the degree of delay is determined when the

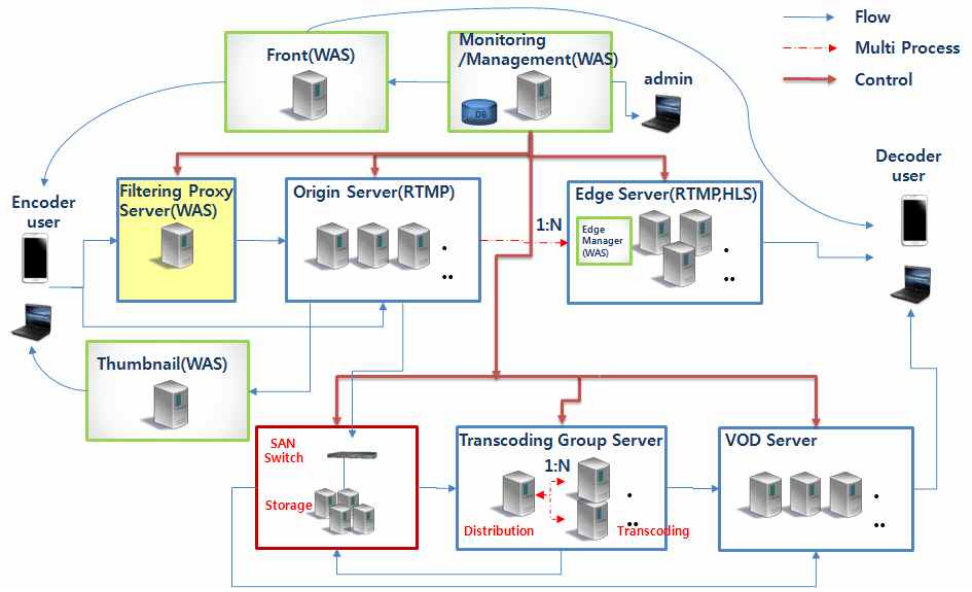

Fig. 2. One-person Media Live System Configuration in Closed Network Environment

user play content according to the segmentation attribute.

2) Common Management

(1) Front Server

we compose manager for the current input/output URL information. Each time a broadcast is started, a new channel is created, so the URL information must be generated dynamically. Therefore, when a request for streaming is received from the Encoder, it is necessary to transmit Edge URL(RTMP, HLS) and Thumbnail URL information.

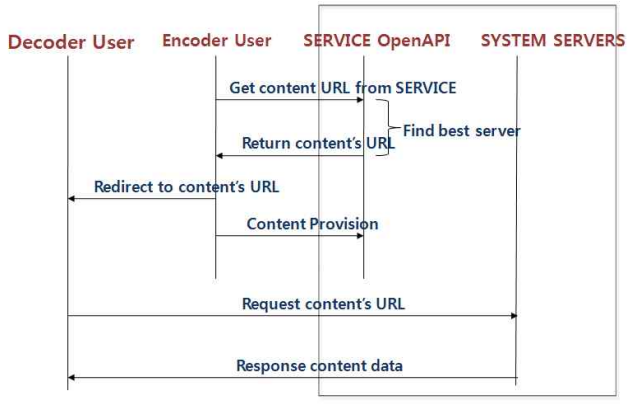

Fig. 3. Dynamic URL Request Process

The process of requesting a URL is shown in Figure 3. Encoder User uses OpenAPI, Monitoring/Management server monitor each server to find the best streaming server, informs the URL and delivers the URL for Decoder User. The Encoder User will then supply the stream data. If Decoder User want to find the broadcast you want to watch, you can request URL from SYSTEM SERVERS and get it.

(2) Monitoring/Management Server

It periodically fetches physical&connection information from all servers. The status of each server is determined, and the Active server is determined and forwarded the information of server's status to the Front Server. To do this, we configure and operate an information manager for all servers. In addition, it determines whether there is a fail over in the edge servers, replaces the server, and determines the auto scaling of stream.

(3) Admin Application

We manage the registered server pool. Physical&connection information of all servers is visualized from Monitoring/Management Server. At this time, auto scaling of the stream is performed through the physical&connection information.

(4) Encoder Application

We receives the stream data from the client device (PC, terminal, etc.) and delivers it to the streaming server. The video obtained from the device is compressed to H.264 and the audio is compressed to AAC and output in RTMP format. we can adjust the resolution, profile level, bit rate and other properties for output streaming.

(5) Thumbnail Server

We receive RTMP input from Origin Server. When extracting

thumbnail, we can specify the extraction cycle and resolution for live stream. And we install an HTTP server and get thumbnail file via URL link.

3) Live Management

(1) Origin Server

We receive RTMP input from Encoder User. It also delivers the stream to the Edge/Filtering Proxy Server

(2) Edge Server

We receive RTMP input from Origin Server. To support HLS for this input stream, we convert it into HLS service file(ts, m3u8) in publish folder of HTTP server. If the Encoder User has finished streaming, this server delete the publish folder.

(3) Filtering Proxy Server

We receive RTMP input from Origin Server. This input should be validated as a stream from the Encoder User. A validity check is necessary because the streaming server may be intentionally caught by an abnormal user. The validity check procedure is as follows.

A. Origin Input Manager periodically checks URL generation rule of input that is actually pushed to Origin.

B. This server turns away input stream with the abnormal id or URL

4) VOD Management

(1) Transcoding Server

We receive RTMP input from Origin Server. It is necessary to determine the attributes such as resolution, profile level, and bitrate according to the user's network environment and transcoding each input stream. In the case of transcoding, if the load of each server exceeds the standard, it is distributed to other transcoding servers and converted. The converted data is stored in mp4 format.

(2) VOD Server

We install the HTTP server. we transmit the mp4 file stored in storage with the HTTP server.

Ⅳ. System Implementation

1) Configuration

Ten servers were used to implement the proposed system. Each server configured WAS(Web Application Server) to perform functions in the proposed system. In case of edge server, two servers are operated for auto scaling. In case of transcoding server, two servers are also operated for distributed conversion. In Edge Server, HLS recommendation attribute(10 seconds) is set for segmentation, and input stream is converted into mp4 for low, medium and high quality data in transcoding. When physical& connection is checked in Monitoring/Management Server, stream is auto scaling or distributed transcoding when 80% of memory and 90% of cpu is over in the currently installed server.

2) Prototype

Fig. 4. Encoder User Application

Figure 4 shows an Encoder User Application that can verify that the Encoding User receives the user's data captured by the camera through the process of Figure 3. The left screen is the screen transmitted from the camera, and the right screen is the stream transmitted through the Edge Server. As a result, it was verified that one person can live service even in the closed network

Ⅴ. Result

As the service environment is changed, the implementation method of one-person live system can be applied variously. In particular, due to the importance of security, it is impossible to use CDNs used in open networks in government offices or military organizations operating closed networks. Therefore, in this paper, we propose one-person media live system suitable for a closed network. Especially, this system has the advantage that the server operation can be efficiently performed by increasing and decreasing the server even if the user access is increased in the closed network environment.

Reference

[1] Shin Hee-Yeun, Jung Won Sic, “A Study on Characteristics of Mobile Video Contents Brand Strategy," *The Journal of Image and Cultural Contents*, Vol. 9, pp. 23–46, 2015

[2] Tran Thi Thu Ha, Jinsul Kim, Jiseung Nam, “Design and Deployment of Low-Delay Hybrid CDN–P2P Architecture for Live Video Streaming Over the Web," *Wireless Personal Communications, Springer*, Vol. 94, No. 3, pp. 513–525, 2017

[3] Myung-Kyu Kwon, “Softwarization of Cloud-based Real-Time Broadcast Channel System,” *Journal of the Korea Society of Computer and Information,* Vol. 22, No. 9, pp. 25-32, 2017.

[4] Karine Pires, Gwendal Simon, “YouTube Live and Twitch: A Tour of User-Generated Live Streaming Systems,” *Proc. 6th ACM Multimedia Syst. Conf. (MMSys),* pp. 225-230, 2015.

[5] Fei Chen, Cong Zhang, Feng Wang, Jiangchuan Liu, “Crowdsourced Live Streaming over the Cloud,” *IEEE Conference on Computer Communications (INFOCOM),* pp. 2524-2532, 2015.

[6] Matthew K. Mukerjee, David Naylor, Junchen Jiang, “Practical, Real-time Centralized Control for CDN-based Live Video Delivery,” *Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication*, pp. 311-324, 2015.

[7] AWS. AWS Elemental[Internet]. Available: https://aws.amazon.com/ko/digital-media/aws-elemental/, 2017.

Dae-Jin Kim

2000: Electronics, Graduate School of Dongguk University(M.S Degree)
2010: Electronics, Graduate School of DaeJin University(Ph.D Degree)

2000～2003: Hanbitsoft Co., Ltd.
2003～2007: Motorola Co., Ltd.
2013～2014: PandoraTV Co., Ltd.
2014～2015: Markany Co., Ltd.
2017～now: Reserch Institue for Image&Cultural Contents in Dongguk University

※Research Interests: CODEC, Multimedia Platform, Fingerprint, Watermark, Deep learning, License Recognition, Parking Guide Information