CHALLENGES WITH VIDEO STREAMING IN CLOUD COMPUTING

Sofia Singla, Dr. Isha Batra*, Dr. Arun Malik

Abstract: Video on Demand (VOD) is an important aspect in today's world due to its efficiency and availability. High demand for VOD leads to high computational and network requirements. Now-a-days, live streaming has many challenges like buffering, scalability, limited bandwidth, poor connectivity etc. Many frameworks and architectures are existing but they are still facing challenges regarding video streaming. Due to its high requirements there are some problems which are faced during video transmission. In this paper, various challenges has been discussed which can improve in future.

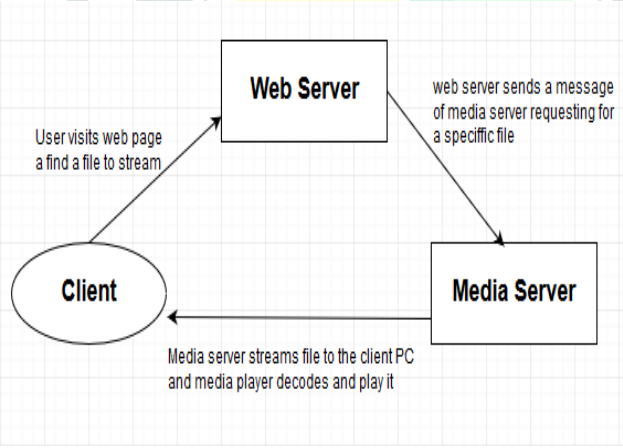
Keywords: video streaming, video on demand, digital rights management, quality of service.

1. INTRODUCTION

Video streaming technology on the internet come after long way of continuous buffering problems and all. High speed network connections, enhanced plugins, and good configuration of systems contributed to putting on the video streaming technology on the internet.

How video streaming works?

In order to successfully stream a video, end user own a video player (QuickTime, Real Media, Adobe flash) on the system and selects the video to be played. The copy of video never exists on end user's system, instead fragments of video are transmitted to end user's system and are immediately send to display drivers.



Streaming servers and real time protocols are used to watch the streaming video. The protocols used are RTP (real time protocol, RTSP (real time streaming protocol), or RTCP (real time transport control protocol). Streaming servers are not web server, rather they are media servers which send stream directly to end user.

2. RELATED WORK

In this paper, major challenges related to video streaming has been discussed. Video streaming has bring the dynamic change in today's world. Users are enjoying various advantages by using video streaming applications. But many drawbacks are also there with video streaming which are facing by users everyday. Numerous frameworks are already existing :-

An adaptive video streaming mechanism has been proposed [1] which includes sliding window protocol and Standard H.264 encoding technique. This framework evaluated by using various platforms to achieve better results. Now-a-days, users are using android applications more frequently. User is trying to do everything using only android applications. Video streaming is one the application which is done by user by using android applications. So, a new framework named Cloud based Multiple View Crowd source Streaming has been proposed[2].

Video Streaming is being done using internet. Video streaming in cloud computing shows maximum bandwidth issues and provides less security. To decrease this issue fog computing can be used to get better results. A new framework has been proposed to provide lower response time and better results in various terms[3].

Reliable video streaming is always recommended. An architecture has been proposed to solve the issues related with services of video streaming[4]. In this Paper, fog computing infrastructure has been introduced to improve the quality to end users[4].

3. CHALLENGES IN VIDEO STREAMING

3.1. Bandwidth Bottleneck

Network Bandwidth is one of the most required resource in cloud video streaming. Earlier when networks were not so fast, online video streaming doesn't used to exist. In some of the cases if video streaming existed, there used to be loads of buffering and pop-ups. With advancement in networking hardware and links, these days bandwidth that a normal user is getting is quite sufficient for video streaming.

However, video quality of streaming depends as well on the actual bandwidth. In 10 – 15 kbps, 720p video streaming is not possible, but for 240p streaming it is sufficient.

So, problem of bandwidth bottleneck is not actually a problem, if low quality streaming is ok for the user. For high quality video streaming, more bandwidth is required.

These days all other resources like memory, storage, cache and processor are sufficient enough for video streaming capacity, bandwidth is the only cause if there is any problem in video streaming. So, bandwidth acts as bottleneck in video streaming. That's why bandwidth efficient frameworks are really necessary to remove the bandwidth fluctuation issues which leads to bandwidth bottleneck[5].

So, it is a responsibility of end-user to access a minimum required bandwidth from ISP as per the need of video quality. However, user can implement some additional steps so that the noteworthy video quality is accessible to the user.

Decrease no. of active devices: -

If quantity of gadgets is more on the network, then the bandwidth used by video streaming will be less and hence less quality.

Update your router for optimize streaming: -

Putting resources into gadgets can give you streaming rich media services. So updating the router can be on one of the choice or only upgrading firmware of the router to latest version may reduce latency and delay issues.

3.2. Copyright Issues

Some of the video streamed on cloud platforms may be subjected to copyright, means these videos should only be viewed live (video on demand) and should not be available offline. No matter how much security is implemented, after all content of the video will reach the end-user only through his system memory, and there are ample no of programs available which can cache the content of system memory and save it for offline use.

For example, some of the paid courses online are only subjected to accessed via live streaming, but still these courses are downloadable and available offline at cheaper price. So, piracy of digital content is one of the most tedious problem in cloud computing. Proper licensing and legal rationale should be followed for the fair use of data[6].

There is no way to 100% protect your content from stealing, however some of the methods can make it harder for people to steak your content.

Disabling copy and right-click

This would take some lines of code to embed it into page and it can prevent 70-80% non-technical people from stealing your content.

Implement DRM

Digital rights management is a set of access control technologies for restricting the use of copyrights works. DRM controls the reading if the entire data. Some of the DRM systems uses display drivers and encrypt the data until it is displayed on the screen.

3.3. Scalability

Some of the older devices doesn't support video streaming from YouTube, Netflix and Amazon prime. Because these services are protected by digital rights management, or DRM services. This is one of the irritating problem in cloud computing. If someone has a access to high quality video streaming and sometimes he is able to watch only low quality video streaming, this is also because of scalability issue. To discard this scalability issue, scalable video coding can be used[7]. More we head on to security, lesser we are able to maintain availability.

The scalability problem can be solved by: -

Different levels of security: - Cloud services can implement different levels of security for different android devices.

Using Widevine: - Widevine is the mostly used DRM solution which supports most of the platforms and is most effective when it comes to protecting digital content.

3.4. Adaptability: -

Cloud video streaming requires an adaptive environment to run completely. In some of the environment it won't be accessible because of environment incompatibility. For example, Streaming Video in a browser without flash player support is not possible. In some of the browsers user need to manually allow the flash player to run.Now-a-days, mobile cloud computing is more prone to adaptability issue in video streaming. That's why, various adaptability video streaming methods are used in mobile computing[1].

Another problem can be cache memory of the browser may be full and the player will not be able to cache the content.

In some cases, hardware acceleration issues might create a problem. Sometimes multiple firewalls, antivirus may not allow the video to stream. These issues can be resolved by: -

Toggling browser: - As some browser may not have in-built support for flash player, so changing the browser might be the good choice to tackle the problem of adaptability.

Clearing Cache: - User should try to clear cache memory of browser as there may be no space for video content to be cached.

H/W acceleration setting: - By adjusting the hardware acceleration setting, this problem can be solved.

Update Display Drivers: - An outdated or corrupted display driver may cause video streaming issues. So, updating display drivers may solve the problem.

3.5. Quality of service: -

A single cloud service provider may be providing more than one cloud services at a time. For example, Video on demand, Online IDE access. So, the cloud provider maintains a QOS queue for different types of data being sent to the client. Sometimes video streaming data may have more priority than any other data, in that case more bandwidth should be given to video packets.

It may be the case that video streaming having less priority than other types of data, for example online IDE tool having more priority and hence provided more bandwidth. In such case, video streaming may be of low quality or it may lag (buffering). UDP connection may be used for such situations because if some packets of video streaming are skipped, there may not be any problem and such minimal loss can be tolerated. So, deciding Quality of service for different types of data is always a headache for cloud service provider in case of video streaming.

Quality of service for video streaming is measured in terms of following: -

Available Bandwidth: - Out of provided bandwidth, how much bandwidth is used by video streaming. More the bandwidth, more priority is given to video data packets.

End-to-End Delay: - Total time taken for transmission of data from server to client. Compression, packetization, queuing, serialization, propagation, processing, and decompression all contribute to the total delay in video transmission.

Delay Variation (Jitter): - Jitter is based on concurrent traffic and activity and network condition. It is basically variation in delay.

Packet Loss: - If volume of incoming traffic is more than the capacity of interface or link, packets may be dropped in that case.

Quality of service can be improved by: -

- Increasing Bandwidth.

- Identifying traffic and defining policies for each kind of traffic.

- Implementing QOS monitors such as (PRTG Network Monitor).

3.6. Video Compression: -

Video compression is process of encoding a video in such a way that it takes less storage and is easy to transmit over a network. In video compression redundant and non-functional data is eliminated from the video. Hence less storage and bandwidth are consumed[8].

There are broadly two types of compression methods: -

Lossless Compression: - It is used to reduce the amount of information to be transmitted in such a way that when compressed information is decompressed, there is not any loss of information.

Lossy Compression: - In this method, some of the information is lost when a compressed information is decompressed. It cannot produce exact copy of information as it was before compressing. There are different lossless compression methods: -

a. **Run Length Encoding: -** In this method, first of all, file is read, and then redundant strings are identified. After that Redundant strings are removed and only one copy of string is preserved followed by escape character and no of times that string matched in original file.
b. **Huffman Coding: -** It is a bottom-up approach of compression in which first of all nodes are created, and then assigned a number according to their position in the tree, and then compressed according to the path from root to that node.
c. **LZW (Lempel-Ziv Welch) method: -** It will read information and give a code to each character, it will assign the same code to each character in a file until null is encountered.

So, the problem that can arise during video compression is because of lossy compression. However lossy compression is fast, but It may create problems during decompression as it cannot create the exact copy of information. So, the solution is to:

Use lossless compression: - In this case user must compromise with bandwidth and storage as it will require more space than in lossy compression.

Use suitable methods for different types of data: - Different methods can be used for different data for example, run length encoding method is best for images having solid black pixels.

3.7. Latency and Reliability Issues: -

With higher resolutions and bitrates, a large amount of information is conveyed, and this volume of movement can bring about inertness and dependability issues. Sometimes, it is noticed that streaming video totally drops out at an important minute. Low latency video streaming frameworks are existing to provide better results[9].

These types of problems can be reduced by lowering the latency rate: -

HTTP Adaptive Streaming: -

Conventional HTTP streaming (using HLS or DASH) can be have 30-60 seconds of delay. It can be decreased by diminishing fragment length up to 10 seconds.

Chunked Transfer Encoding: -

This method diminishes the division delay in video streaming. It can make chunks of more smaller size (less than 2 seconds) and hence if packet loss happens, less bandwidth will be

consumed to re-transmit.

Signalling: -

Instead of TCP overhead, UDP connection may be used where loss of some chunks is tolerable.

3.8. Synchronization: -

Before transferring video data to client, image sequence and audio and compressed separately using different compression technique. It is easier to transmit different types of data separately over a network. At client-side same data is decompressed using different techniques and merged afterwards before sending it to drivers.

In such situation, synchronization of data is of vital importance. There may be the case that audio data may be lagging or leading. Media synchronization frameworks are existing to provide better results[10]. These synchronization problems may be caused due to following reasons: -

QOS: - Quality of service may cause the problem because video data may be prioritized than audio data or vice versa. In such case audio data might not be received completely by network interface card and video data may be already in processing because of priority.

Compression anomaly: - On server side there can be problem while compressing the information. For example, synchronization counter for both data may vary slightly and it will cause lagging during play.

Video player Synchronization: - Video player might cause problem while decompressing both kind of data. Although it can be fixed by adjusting synchronization counter for audio or video data.

4. CONCLUSION

This paper discuss about the various challenges with video streaming in cloud computing. Numerous advance architectures and frameworks are existing , but users are still facing challenges in the video streaming. Challenges like bandwidth bottleneck, security, scalability, copyright issues etc. are mention in this paper. To improve these challenges more advance architectures can be propose in future.

References

[1] S. P. Tamizhselvi and V. Muthuswamy, “Adaptive video streaming in mobile cloud computing,” *2014 IEEE Int. Conf. Comput. Intell. Comput. Res. IEEE ICCIC 2014*, pp. 467–471, 2015.

[2] S. S. Panhalkar and H. P. Khandagale, “Video Streaming Using Cloud Computing Based on Android Application,” *Proc. - 2018 4th Int. Conf. Comput. Commun. Control Autom. ICCUBEA 2018*, pp. 1–6, 2018.

[3] S. F. Hassan and R. Fareed, “Video streaming processing using fog computing,” *ICOASE 2018 - Int. Conf. Adv. Sci. Eng.*, pp. 140–144, 2018.

[4] A. T. Nasrabadi, A. Mahzari, J. D. Beshay, and R. Prakash, "Adaptive 360-degree video streaming using scalable video coding," *MM 2017 - Proc. 2017 ACM Multimed. Conf.*, pp. 1689–1697, 2017.

[5] C. Zhou, Z. Li, and Y. Liu, "A measurement study of oculus 360 degree video streaming," *Proc. 8th ACM Multimed. Syst. Conf. MMSys 2017*, pp. 27–37, 2017.

[6] T. jun Jung and K. deok Seo, "A client-driven media synchronization mechanism for RTP packet-based video streaming," *J. Real-Time Image Process.*, vol. 12, no. 2, pp. 455–464, 2016.