PPSP
Internet Draft
Intended status: Informational
Expires: January 2010

Y.Zhang
China Mobile
C.Li
Beijing Jiaotong University
July 9, 2009

Protocol Analysis and Comparison of PPlive,PPstream and UUSee by Internet Measurement
draft-zhang-ppsp-protocol-comparison-measurement-02.txt

Status of this Memo

Copyright Notice

Abstract

In this draft we introduce an Internet measurement work for pplive, ppstream and UUSee. First, we give a brief introduction about our motivation and target of this measurement. We then introduce the methodology, platform, data and modeling of our measurement. Finally we outline the p2p media streaming protocols by the measurement.

Table of Contents

1. Introduction

P2P media streaming is one of the most popular p2p system and is a new developing trend for modern video system. More and more users, vendors as well as researchers have been attracted to it. Currently, there are many such systems on Internet, such as PPLive, PPStream, UUSee, etc. Usually all of them can provide Live and VoD programs to users. They have a similar system structures according to our measurement. In good network conditions, all of them can bring good experiences to users; once the stable condition is broken, the p2p applications will present different performance.

2. Motivation and Target of the Measurement

Naturally, we care about how to evaluate system performance, what the performance limitations are under current system models, how to decrease the pressure on the network. In order to answer these questions, we study the following aspects of p2p streaming systems:

1. To extract the general p2p media streaming system structure;

2. To analyze and evaluate the signaling (protocol) messages (types, format, function, exchange sequence);

3. To study and model the core mechanism such as initial buffer lag policy, buffer cache schedule policy, data fetching policy, etc;

4. To optimize the system performance and improve network robustness;

5. To study its effect on physical layer network, transport lay protocol (TCP and UDP);

6. To study new p2p suitable transport protocol instead of application lay p2p transportation;

7. To study the availability of current p2p streaming systems on mobile ip;

8. To present related standard suggests.

3. Methodology of Our Measurement

Real network based measurement is important in studying p2p media streaming systems. However all current commercial p2p media streaming systems are propriety protocol systems without any public document about system' working mechanism available. Although some researchers report some useful finds, most of our concerns are not resolved yet. We have to firstly deeply understand how the system works. Usually we use certain reverse-engineering method to analyze its working principle. In the protocol cracking, we mainly focus on the top 3 of very popular p2p streaming systems - PPLive,PPStream and UUSee. Beside their popular degree, several other reasons make them as the starting: Each of them uses plain text in their protocol exchanging; both udp and tcp are adopted; each of them is the new generation data driven p2p streaming systems.

Our reverse engineering are conducted in following 3 stages:

Firstly, by tracing a standard client, we capture interactive packets between the local peer and others with ethereal/windump tool. Based on our experience on the p2p streaming system, some basic protocol types must be included such as buffer map, chunk request/response, shake hand packet, etc. For each udp packet, it is reasonable to assume that each packet is a completed protocol message, while for tcp packets we must extract messages from all application data stream. Usually we can get the rough protocol format by matching traced packets to the known basic protocol types. But there are still many unknown details left.

Secondly, the traced data is fed into our dumping tool, which can filter data into a text file with composed conditions, such as source ip/port, destination ip/port, VoD/Live protocol type. Through inspecting the text file of different channels, we find many regular changes that help us in parsing the protocol format in details. Of course, there is still one-third irregular data. In this case, we guess and try it based on our experiences in next step. At last, we parse about nearly 90% of PPLive and PPStream while 50% UUSee protocols(mostly because its signaling is encrypted). In PPLive system, there are about 15 message types, all of that haven't any clearly protocol marks in order to escape from restrictions ISPs may imposed on as we guess. Among these types, buffermap and peerlist messages are the most useful messages for us to measure and analyze the system.

At the third stage, we analyze the time sequences of the protocol messages. Some communication time sequence is obvious, such as chunk

request/response, while others are not. In later case, we guess and try differently composed events until a sequence makes right sound. Ultimately, all time sequences in communications are resulted successfully.

4. Measurement Platforms

For different analysis, we take different measurement methods. In general, our environment consists of 4 dell pc servers in a private ip space (LAN) behind a 6mbps ADSL NAT router. each of which has 2.8Hz Pentium CPU, 1MB memory, 80GB hard disk, 10/100Mbps Ethernet, windows OS, mysql database.

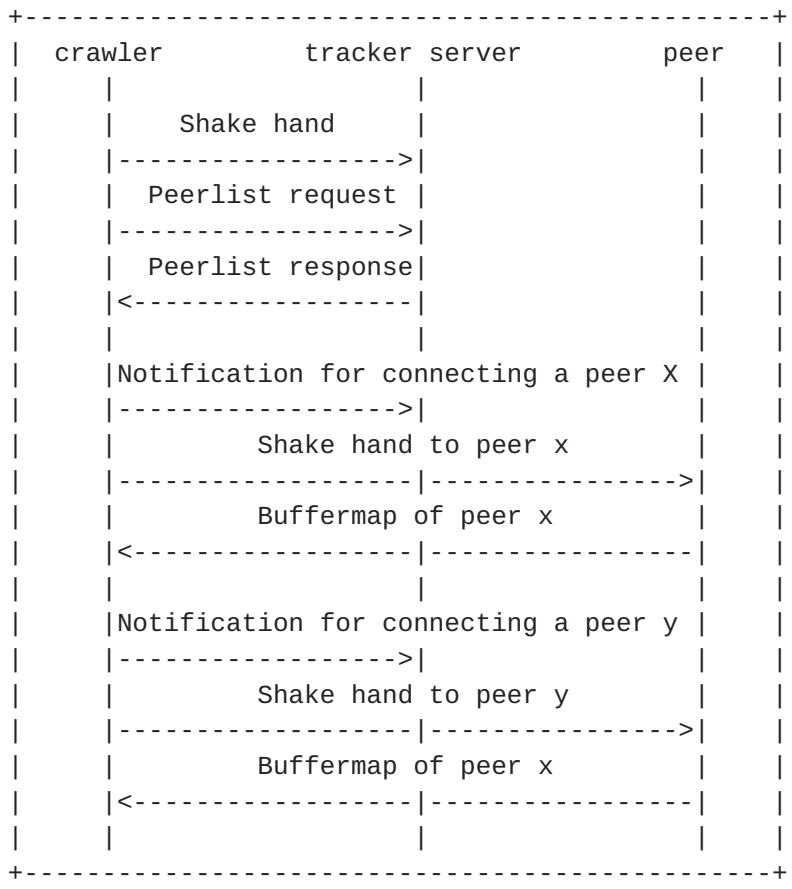

Figure 1 Protocol sequence for crawling

1. Official client trace: use tcpdump/windump/wireshark to capture exchanging packets between local peer and other peers.

2. System topology crawler: based on p2p streaming protocol, design legal measurement tool to detect the whole network in short time. We usually take some measures improve the probing efficiency such as multi threads, fast data searching in memory, multi-tables in

databases, master-slave distribution deployment, concurrent tcp session limitation enlargement.

3. Long term multi online peers probe: different from topology crawler, it will probe a certain online peers in a long term in order to collect the detail information of these peers. Usually more complete protocol set should be realized in it.

4. P2P streaming client measurement in mobile ip: A fix position PC is used as the mobile device, where three types of software have been installed: p2p streaming client software; packet capture software; our mobile ip simulation software.

5. Special client accessing to official network in order to evaluate the system robusticity and optimize the protocol: based on protocols,we build the legal p2p client to join in the official p2p network. By this client, we can test and evaluate new protocol design and new core system models.

5. Data Analysis and Modeling

As for data analysis and modeling, we have made some progress and published some papers [1]~[6]. Interested readers can refer to them for more details.

6. Study of P2P Media Streaming Protocols by the Measurement

We have measurement PPLive,PPstream and UUSee,and guessed the protocols both for live streaming and VoD streaming respectively. However due to some encryption(It's said to be Scrambling), VoD streaming data from PPStream hasn't been analyzed yet. We list the signaling transaction of pplive live streaming and VoD and ppstream live streaming as follows:

6.1. PPLive Live Streaming:

Messages with Tracker

0101 (peer registration)

0100 (tracker response)

0201 (peerlist request)

0200 (peerlist response)

0301 (tracker offset request)

0300 (tracker offset response)

Messages with Peer

0x4101 (peerlist request)

0x 4201 (peerlist response)

0x 5400 (peerlist response)

0x 4400 (Buffermap response)

0x 5200 (chunk request)

0x 5300 (chunk response)

0x 4601 (chunk response)

0x 6101 (chunk request)

0x6201 (chunk response)

0x 4000 (disconnect)

0x4500 (chunk request)

0x4901 (udp handshake)

0x4400: update/5s

6.2. PPLive VoD

0x04 (peers_data_request)

0x05 (peers_data_response_1)

0x06 (peers_data_response_2)

0x11 (peers_peerlist_request)

0x12 (peers_peerlist_response)

0x13 (peers_shakehand)

0x14 (peers_bitfield)

0x16 (trk shakehand)

0x17 (trk notification of a peer)

0x18 (peers_keepalive)

0x43 (host_ip)

0x44 (public_ip)

0x51 (host_ip_format2)

0x52 (public_ip_format2)

6.3. PPStream Live (early version, TCP)

0x02 (buffermap)

0x22 (buffermap)

0x03 (chunk request)

0x04 (chunk response)

0x05 (nochunk response)

0x08 (peerlist request)

0x09 (peerlist response)

0x40 (media information)

0x1d (client version)

0x10 (protocol error information)

0x1b (remote live time)

0x1 (peer id)

0x19 (remote refuse)

0xA0 (vod bitmap)

6.4. UUSee Live

0x01080909 (UUSEE_MSG_HANDSHAKE_REQ)

0x014203e9 (UUSEE_MSG_HANDSHAKE_PEERLIST)

0x014403e9 (UUSEE_MSG_HANDSHAKE_BUFFERMAP)

0x014503e9 (UUSEE_MSG_HANDSHAKE_CHUNKREQ)

0x18070514 (UUSEE_MSG_HANDSHAKE_ACK)

6.5. Conclusion

PPLive, PPStream and UUSee all support Live and VoD programs, and have similar system structures. As for the signaling protocol, both

systems have similar process. We have depicted the interactions between peers and peers as well as trackers in the problem statement draft. Interested readers can refer to draft-zhang-ppsp-problem-statement for details.

However, there are still some difference between pplive,UUsee and ppstream which belongs to their high secrets. For example, in the buffer aspect, pplive and UUSee have a large buffer, which can smooth time delay jitters but contributes to relative large startup delay. The chunk fetch policy is seqential fetcing and rarest first at same time. The schema of the buffer is a variable length in chunks but fixed length in playback time. The buffer map schedule is simple based on chunksid. On the other hand, PPStream is a small buffer system (early version), which has small startup delay but leads to video freeze with large jitters. The chunk fetch is randomly in each buffer window. The schema of the buffer is of multi playable media blocks with fixed playback time (mulit buffers) for each of them. Hence its buffer schedule is based on multi buffer windows, where each chunk is uniquely marks indicated by the buffer window offset and the internal chunk id. According to our understanding, the mechanism of PPStream is a litter complex.

7. Security Considerations

We don't involve security measurement till now.

8. References

8.1. Normative References

[1] Y. Chen, C. Chen and C. Li, "A Measurement Study of Cache Rejection in P2P Live Streaming System", Workshop on Multimedia Network Systems and Applications in conjunction with ICDCS-2008, June, 2008, Beijing, China

[2] Y. Chen, C. Chen and C. Li, "Measure and Model P2P Streaming System by Buffer Bitmap", HPCC '08, Sept. 2008, Dalian, China

[3] Cx.Li,Cj.Chen "Fetching Strategy in the Startup Stage of p2p Live Streaming", http://arxiv.org/abs/0810.2134

[4] Cx.Li,Cj.Chen "Initial Offset Placement in p2p Live Streaming Systems" http://arxiv.org/abs/0810.2063

[5] Cx.Li,Cj.Chen "Inferring Playback Rate and Rate Resets of p2p Video Streaming Transmissions by Piecewise Line Envelope Approximation" The Journal of China Universities of Posts and Telecommunications March.2009

[6] Cx.Li,Cj.Chen "Real P2P System Measurement under Mobile IP Environment" accepted by icnds2009

8.2. Informative References

Author's Addresses

Yunfei Zhang
China Mobile

Phone: 86 13601032119
Email: zhangyunfei@chinamobile.com

Chunxi Li
Beijing Jiaotong University

Phone: 86 10 51684757
Email: chunxili@yahoo.cn