PPSP
Internet Draft
Intended status: Standard track

Y. Zhang
China Mobile
N.Zong
HuaweiTech
G.Camarillo
Ericsson
J.seng
PPlive
R.Yang
Yale University

Expires: Feb 2011
July 12, 2010

Problem Statement of P2P Streaming Protocol (PPSP)
draft-zhang-ppsp-problem-statement-06.txt

Abstract

We propose to standardize the key signaling protocols among various P2P streaming system components including the tracker and the peers. These protocols, called PPSP, are a part of P2P streaming protocols. This document describes the terminologies, concepts, incentives, and scope of developing PPSP, as well as the use cases of PPSP.

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html

This Internet-Draft will expire on February 2,2011.

Table of Contents

1. Introduction

1.1. Background

Streaming traffic is among the fastest growing traffic on the Internet. In a recent white paper, Cisco predicts that by 2012, 90% of all Internet traffic will be video [Cisco].

There are two basic architectures for delivering streaming traffic on the global Internet: the client-server paradigm and the peer to peer (P2P) paradigm [P2PStreamingSurvey]. A particular advantage of the P2P paradigm over the client-server paradigm is its scalability. As an example, PPLive [PPLive], one of the largest P2P streaming vendors, is able to distribute large-scale, live streaming programs such as the CCTV Spring Festival Gala to more than 2 million users with only a handful of servers. CNN[CNN] reported that P2P streaming by Octoshape played a major role in its distribution of the historical inauguration address of President Obama. It is well demonstrated in practice that P2P streaming can deliver videos encoded at a rate of about 400 Kbps, in the presence of rapid user joins/leaves, with positive user experiences.

With the preceding technical advantages, P2P streaming is seeing rapid deployment. Large P2P streaming applications such as PPLive [PPLive], PPstream [PPstream] and UUSee [UUSee] each has a user base exceeding 100 millions. P2P streaming traffic is becoming a major type of Internet traffic in some Internet networks. For example, according to the statistics of a major Chinese ISP, the traffic generated by P2P streaming applications exceeded 50% of the total backbone traffic during peak time in 2008. In the beginning of 2010, CNTV, China National Network Television for CCTV launched its software named CBox supporting P2P live and VoD programs. To date, it has a rapid user increase. With the opening of FIFA world cup, CBox has increased 5 times in user number with 3 million online users a day and altogether 350 million times view. It is reported that CNTV can support 10 million simultaneous user visit[CNTV]. Similarly there were also reports that major video distributors such as Youtube [youtube] and tudou [tudou] are conducting trials of using P2P streaming as a component of their delivery infrastructures.

Given the increasing integration of P2P streaming into the global content delivery infrastructure, the lacking of an open, standard P2P streaming protocol becomes a major missing component in the Internet protocol stack. Multiple, similar but proprietary P2P streaming protocols result in repetitious development efforts and lock-in effects. More importantly, we notice that more participants beyond P2P streaming vendors have involved in P2P streaming industry, like

infrastructure vendors Akamai[Akamai], ChinaCache and ISP like ComCast[ComCast]. That is, P2P streaming has become an open industry with different participants from the source, infrastructure (in P2P mode although all the peers are super nodes)delivery and local P2P distribution to the terminals.

We argue that proprietary P2P streaming protocols lead to substantial difficulties when integrating P2P streaming as an integral component of a global content delivery infrastructure. For example, proprietary P2P streaming protocols do not integrate well with existing cache and other edge infrastructures.

1.2. Research or Engineering

As [P2PStreamSurvey] identifies, there exist multiple proprietary P2P streaming systems including PPLive, PPstream, UUsee, Pando, abacast, and Coolstreaming. A natural question to ask is whether the development of P2P streaming is mature and ready for standardization. We admit that P2P streaming will continue to improve and evolve. However, our investigation shows that existing P2P streaming systems are largely converging, sharing similar architecture and signaling protocols [draft-zhang-ppsp-protocol-comparison-measurement-00].

The aim of standardization in P2P streaming systems is to 1) decouple the information exchange with the following delivery so that the most common part in P2P streaming can use a generic and open protocol;

2) standardize the information exchange message so that equipments from different providers can interact with each other for a complete P2P streaming system.

1.3. Objective and outline

Multiple protocols such as streaming control, resource discovery, streaming data transport, etc. are needed to build a P2P streaming system [P2PStreamingSurvey]. We call those protocols P2P streaming protocols.

The objective of the PPSP work is to standardize the key signaling protocols among various P2P streaming system components including the tracker and the peers. These protocols, called PPSP, are a part of P2P streaming protocols. Note that the complete set of standard P2P streaming protocols for a whole P2P streaming system could be developed following or parallel to the PPSP work.

- PPSP will serve as an enabling technology, building on the development experiences of existing P2P streaming systems. Its

design will allow it to integrate with IETF protocols on distributed resource location, traffic localization, and streaming control and data transfer mechanisms. Regarding to the components it involves, PPSP allows effective integration between the peer index server named tracker and different kinds of peers including edge infrastructure nodes such as cache, gateway and CDN nodes who can act as super peers and ordinary peers.

This document describes the terminologies, concepts and common architecture for P2P streaming systems, problems without standardized PPSP/incentives to standardize PPSP, scope of PPSP as well as its use cases. The rest of this document is organized as follows. In Section 2, we introduce some common terminologies and concepts. In Section 3, we introduce P2P streaming system architecture. In Section 4, we identify the problems without standardized protocols and incentives for developing PPSP protocols. In Section 5 and 6, we describe the software architecture and functional components of P2P streaming systems and therefore discuss the position and scope of PPSP. In Section 7, we list some PPSP use cases.

2. Terminology and concepts

Chunk: A chunk is a basic unit of partitioned streaming, which is used by a peer for the purpose of storage, advertisement and exchange among peers [Sigcomm:P2P streaming].

Content Distribution Network (CDN) node: A CDN node refers to a network entity that usually is deployed at the network edge to store content provided by the original servers, and serves content to the clients located nearby topologically.

Live streaming: The scenario where all clients receive streaming content for the same ongoing event. The lags between the play points of the clients and that of the streaming source are small.

P2P cache: A P2P cache refers to a network entity that caches P2P traffic in the network, and either transparently or explicitly as a peer distributes content to other peers.

P2P streaming protocols: P2P streaming protocols refer to multiple protocols such as streaming control, resource discovery, streaming data transport, etc. which are needed to build a P2P streaming system.

Peer/PPSP peer: A peer/PPSP peer refers to a participant in a P2P streaming system. The participant not only receives streaming content, but also stores and uploads streaming content to other participants.

PPSP: PPSP refer to the key signaling protocols among various P2P streaming system components including the tracker and peer. PPSP are a part of P2P streaming protocols.

Swarm: A swarm refers to a group of clients (i.e. peers) sharing the same content (e.g. video/audio program, digital file, etc) at a given time.

Tracker/PPSP tracker: A tracker/PPSP tracker refers to a directory service which maintains lists of peers/PPSP peers storing chunks for a specific channel or streaming file and answers queries from peers/PPSP peers for peer lists.

Video-on-demand (VoD): The scenario where different clients watch different parts of the media recorded and stored during past events.

3. Introduction of P2P streaming system

There are multiple available P2P streaming solutions. Some are deployed solutions, while others are still under active study. A survey of existing solutions can be found in [Survey].

In P2P streaming system, there are various swarms with each swarm containing a group of clients sharing same streaming content (e.g. channel, streaming file, etc) at a given time. These clients are called peers, as each client not only receives streaming content, but also stores and uploads streaming content to other clients. In a broad sense of global content delivery infrastructure, peers can include multiple types of entities such as end user applications, caches, CDN nodes, and/or other edge devices. Therefore, the basic functions of a P2P streaming system involve:

1) Maintaining information about which peers in which swarm in some directory service, a.k.a. tracker.

2) In each swarm, exchange information about content availability (e.g. which chunks stored by a peer) among peers, or between tracker and peer.

3) In each swarm, exchange the actual content among peers.

As shown in Figure 1, common information flows in a P2P streaming system include:

1) When a peer wants to receive streaming content:

1.1) Peer acquires a list of peers in the swarm from the tracker. A swarm can be indexed by a channel ID, streaming file ID, etc.

1.2) Peer exchanges its content availability with the peers on the obtained peer list.

1.3) Peer identifies the peers with desired content and requests for the content from the identified peers.

2) When a peer wants to share streaming content with others:

2.1) Peer sends information to the tracker about the swarms it belongs to, plus streaming status and/or content availability.

```
+-------------------------------------------------------+
|                  +-------------------+                |
|                  |   Tracker         |                |
|                  +-------------------+                |
|                   ^  |             ^                  |
|                   |  |             |swarms,           |
|             query |  | peer list   |streaming status  |
|                   |  |             |and/or content    |
|                   |  |             |availability      |
|                   |  V             |                  |
|        +-------------+         +------------+         |
|        |    Peer1    |<------->|   Peer 2   |         |
|        +-------------+ content +------------+         |
|                 ^  ^    availability                  |
|                 *  | content                          |
|         content *  |availability                      |
|                 *  V                                  |
|             +------------+                            |
|             |   Peer 3   |                            |
|             +------------+                            |
+-------------------------------------------------------+
```

Figure 1 Common information flows in P2P streaming system

4. Problem of proprietary protocols and incentives for developing standard PPSP

We start by considering the success of the Web. It is the standard HTTP protocol that makes it possible to deploy the global content distribution eco-system that consists of not only end devices such as Web servers and Web clients, but also infrastructure devices such as Web caches and CDN nodes. All of these devices communicate through standard protocols and provide substantial benefits to the consumers, the content publishers, and the network infrastructure.

As we discussed in Section 1, given the increasing integration of P2P streaming into the global content delivery infrastructure, proprietary P2P streaming protocols not only result in repetitious development efforts and lock-in effect, but also leads to substantial difficulties when integrating P2P streaming as an integral component of a global content delivery infrastructure. The explicit incentives to get rid of the proprietary protocols can be seen from the talks of Johan Pouwelse, scientific director of P2P Next: "?broadcasters from the BBC to Germany's ARD just seem to love the idea of ditching their proprietary platforms[Johan Pouwelse]."

Let's take a look of several cases for further problem identification.

4.1. Proprietary signaling leads to difficult interactions in case of multiple parties involved in the delivery

Let's first see a simplest case.In an open P2P streaming industrial environment, it's a common thing for different streaming vendors(esp. spread in different regions) cooperatively take the broadcasting. Suppose PPLive broadcasts live Chinese spring festival gala for American Chinese by Pando networks. At a first sight, this seems reasonable because there are relatively few American Chinese PPLive users. Therefore it's hard to realize efficient P2P delivery by PPLive network only. Borrowing peer resources from an ally like Pando may help the efficient distribution. However different messages and interactions in the two systems cause the difficulty in interoperability among PPLive peers and Pando peers.

Consider a more complex case where P2P streaming vendors cooperate with CDN providers. People can refer to UUSee, RayV and Forthtech practice for this use case. In the context of P2P streaming, infrastructure devices such as edge caches and CDN nodes have been shown as promising means to both improve the performance of P2P streaming (e.g., lower latency) by providing more stable "super peers" and reduce traffic in ISP network [CDN+P2P] [RFC 5693].

However, there can be substantial obstacles in deploying infrastructure edge devices supporting proprietary P2P streaming protocols [HTPT]. Unlike the Web with the standard HTTP protocol, the current P2P streaming landscape consists of multiple, proprietary P2P streaming protocols mostly differing in signaling transactions. Consequently, in order to support P2P streaming, the infrastructure devices need to understand and keep updated with various proprietary P2P streaming protocols. This introduces complexity and deployment cost of infrastructure devices.

Things get worse if there are M P2P streaming vendors and N CDN providers for possible cooperative combination. How does a specific CDN node identify different private systems and report to different trackers with proprietary protocols? It seems there are no good ways to address this. The CDN node has to update its protocol through case-by-case negotiations.

With standard PPSP, edge caches and CDN nodes can be designed to inter-operate with only the standard protocols, reducing the complexity and cost to support streaming involving P2P.

4.2. Proprietary signaling leads to multiple client software in a terminal

Because of the private protocols, although there are quite much in common, application developers cannot reuse the common parts, wasting a lot of repeated work.

This makes a terminal installing multiple different software for different purpose. For example a user installs CBox for CCTV program watching and PPLive for Japanese and Korean movies. This brings two problems:

1) Terminal limitation may don't allow for many clients in one machine, esp. for mobile terminals. The limited CPU,storage and cache often limit the concurrent threads and processes.

2) Because different software are independent, it may lead to vicious competitions. We even see the competitors to delete each other's software when automatically running the software. If there are standard protocols and some common part to co-use, such things are hard to occur.

4.3. Proprietary signaling leads to low network resource utilization

From the network resource utilization perspective, if we have no standard protocols in designating the resource availability(which is the imagined PPSP task)and every application uses proprietary protocol for storage and bandwidth usage, then for a same content, many on-the-way data in different applications have to be cached/stored and transferred repeatedly, which wastes storage and causes possible congestion in the network.

4.4. Proprietary signaling doesn't handle well with mobile and wireless environment

Mobility and wireless are becoming increasingly important features to support in future Internet deployments [GENI], [FIND]. Currently there are more and more mobile and wireless Internet users. By the end of 2009, there are 233 million mobile users in China[CNNIC].

Along with the introduction of mobile and wireless capabilities into the Internet, mobile streaming is becoming a key offered service [MobileTV]. In Korea the number of mobile TV subscriber has reached seventeen millions, accounting for one third of the mobile subscribers. In Italy, there are one million mobile TV users. During the 2008 Beijing Olympic Games, more than one million users utilize China mobile's mobile TV service.

Considering the mobile and wireless nodes have better CPU, memory and storage and the mobile network has better network bandwidth (esp. there are more uplink bandwidth which is wasted for transferring little data in current practice) than before, there are much more possibility for the mobile and wireless node to be peers supporting P2P streaming.

However, mobile peers may face bigger challenges for supporting P2P streaming with unsteady network connections, less steady power and different media coding for mobile devices. Current proprietary protocols are mainly designed in fixed Internet environment and don't address these challenges. We may therefore raise such a question: Shall we let these private protocols to fit in mobile environment system-by-system independently or solve these problems in the design of an open PPSP protocol suite?

The answer is obviously clear. It is worth mentioning that the development of PPSP should consider the specific requirements of mobile Internet. For example, the overhead of PPSP be small in low bandwidth mobile Internet. Also, information exchange in PPSP should support mobility, low battery and heterogeneous capabilities of mobile terminals. Systematic requirements on the development of PPSP will be addressed in the requirements documents.

5. Components of P2P streaming system

```
+-----------------------------------------------------+
|                  Application Layer                  |
|-----------------------------------------------------|
|                 Play-out Layer                      |
| +----------+  +------------+   +-----------+        |
| |start/stop|  |pause/resume|   | FF/rewind |        |
| +----------+  +------------+   +-----------+        |
|---------------------------------------------------- |
|                Information Layer                    |
| +------------+   +------+   +-----------+           |
| |registration|   |report|   | statistics|           |
| +------------+   +------+   +-----------+           |
|-----------------------------------------------------|
|                Communication Layer                  |
| +--------------------+   +------------------+       |
| |tracker communication|   |peer communication|     |
| +--------------------+   +------------------+       |
|           +-------------+                           |
|           | bootstrap   |                           |
|           +-------------+                           |
|-----------------------------------------------------|
|                  Transport Layer                    |
+-----------------------------------------------------+
```

Figure 2 Components of P2P streaming system

To organize our efforts, we show the components of a complete P2P streaming system in Figure 2.

1) Transport Layer is responsible for data transmission between peers. UDP, TCP or other protocols can be used.

2) Communication layer includes three components:

2.1) Tracker communication is a component that enables each peer to get peer list from the tracker and/or provide content availability to the tracker.

2.2) Peer communication is a component that enables each peer to exchange content availability and request other peers for content.

2.3) Bootstrap is a component that enables newly joined nodes to obtain tracker information.

3) Information layer is responsible for peer and content information collection and management.

3.1) Registration is a component that enables nodes to register to the system, and publish the content information. The information may include but is not limited to: content description, content type, creation time, node information such as physical location, IP address.

3.2) Report is a component that enables peers to report streaming status to the tracker. The information may include peer inbound/outbound traffic, amount of neighbor peers, peer health degree and other streaming parameters.

3.3) Statistics is a component that enables trackers to manage the aggregated system information for global control in upload bandwidth consumption, overhead consumption and other regards.

4) Play-out layer is responsible for controlling the action of media play (e.g. start, pause, resume, stop, fast-forward, and rewind).

5) Application layer is the top layer for streaming applications.

6. Scope of PPSP

6.1. Protocols to be standardized

We propose to standardize protocols in PPSP which enable the tracker communication and the peer communication components in the communication layer, as well as the report component in the information layer. These protocols, called PPSP, are key mechanisms involving two important roles - tracker and peer in P2P streaming processes, as addressed in Section 3. These signaling protocols,in essence, aim at standardizing the content information exchange mechanisms among different devices in P2P streaming systems. Note that PPSP are only a part of P2P streaming protocols. The complete set of standard P2P streaming protocols for a whole P2P streaming system could be developed following or parallel to the PPSP work.

Because bootstrap, registration and statistics components are out-of-band mechanisms for streaming processes, they are not in current scope of PPSP. Both transport, play-out and application layers in P2P streaming system are also beyond the current scope of PPSP.

Therefore, PPSP include the PPSP tracker protocol - a signaling protocol between PPSP trackers and PPSP peers, and the PPSP peer protocol - a signaling protocol among PPSP peers.

1) PPSP tracker protocol

This protocol will define:

1.1) Standard format/encoding of information between PPSP peers and PPSP trackers, such as peer list, content availability, streaming status including online time, link status, node capability and other streaming parameters.

1.2) Standard messages between PPSP peers and PPSP trackers defining how PPSP peers report streaming status and request to PPSP trackers, as well as how PPSP trackers reply to the requests.

Note that existing protocols should be investigated and evaluated for being reused or extended as the messages between tracker and peer. Possible candidates include the use of the HTTP, where the GET method could be used to obtain peer lists, the POST method used for streaming status reports, etc.

2) PPSP peer protocol

This protocol will define:

2.1) Standard format/encoding of information among PPSP peers, such as chunk description.

2.2) Standard messages among PPSP peers defining how PPSP peers advertise chunk availability to each other, as well as the signaling for requesting the chunks among PPSP peers.

Again, existing protocols should be investigated and evaluated for being reused or extended as the messages among peers. Possible candidates include the use of the HTTP, where the GET method could be used to obtain chunk availability, etc. Considering the potential large number of peers, some lightweight (possibly binary) protocols could also be good candidates.

6.2. Service types to be considered

As stated in Section 1, PPSP will serve as enabling technology and tools for building various P2P streaming systems. We are not standardizing certain streaming services. The reason why we list service types here is to show we would consider the properties of these services as the requirements for PPSP design.

Common service types supported by current P2P streaming systems include live streaming(including time-shift), video-on-demand (VoD).

In live streaming, all PPSP peers are interested in the media coming from an ongoing event, which means that all PPSP peers share nearly the same streaming content at a given point of time. In live

streaming, some PPSP peers may store the live media for further distribution, which is known as TSTV (time-shift TV) where the stored media are separated into chunks and distributed in a VoD-like manner.

In VoD, different PPSP peers watch different parts of the media recorded and stored during a past event. In this case, each PPSP peer keeps asking other PPSP peers which media chunks are stored in which PPSP peers, and then pulls the required media from some selected PPSP peers.

7. Use cases of PPSP

7.1. Worldwide Provision by cooperative P2P Streaming vendors with PPSP

As stated in section 4.1,the cooperation of P2P Streaming vendors can easily expand the broadcasting scale with standard PPSP. The interactions between cooperative P2P streaming provider A's tracker server and P2P streaming provider B and C's SuperNodes is shown in Figure 3.

```
   +-------------------------------------------------------------------+
   |                                                                   |
   |                          +------------------+                     |
   |            +------------>| A's     Tracker  |<----------+         |
   |            |             +------------------+           |         |
   |     Tracker|                ^              ^            |         |
   |    Protocol|         Tracker|              |Tracker     |Tracker  |
   |            |        Protocol|              |Protocol    |Protocol |
   |            |                |              |            |         |
   |            |                |              |            |         |
   |            v                v              v            v         |
   |      +------+ Peer    +------+            +------+    +------+    |
   |      | B's  |<------->| B's  |            | C's  |    | C's  |    |
   |      | SN1  |Protocol | SN2  |            | SN1  |    | SN2  |    |
   |      +------+        +------+            +------+    +------+    |
   |         ^  ^                                           ^ ^        |
   |         |  |                                           | |        |
   |         |  | Peer Protocol                Peer Protocol| |        |
   | Peer    |  +-------------+               +-------------+ |Peer    |
   | Procotol|                |              |                |protocol|
   |         |                |              |                |        |
   |         |                |              |                |        |
   |         |                |              |                |        |
   |         v                v              v                v        |
   |      +------+ Peer    +------+    +---------+  Peer   +---------+ |
   |      | A's  |<------> | B's  |    |A's      |<------> |C's      | |
   |      | User1|Protocol | User2|    | User1   |Protocol | User2   | |
   |      +------+        +------+    +---------+        +---------+ |
   |                                                                   |
   +-------------------------------------------------------------------+
```

Figure 3 Cooperative Vendors Interactions

7.2. Three Screen P2P streaming in heterogeneous environment using PPSP

This is a use case where PC, Setbox/TV and mobile terminals from fixed Internet, mobile Internet constitute the peer overlay for streaming content sharing. Using PPSP protocols, peers from different kinds of networks can share and download what they have from each other to form a 3 screen streaming system.

```
+--------------------------------------------------------------------+
|                                                                    |
|       Tracker Protocol  +---------+   Tracker Protocol            |
|         +-------------> | Tracker |<------------------+            |
|        |              +---------+                   |            |
|        |                   ^                         |            |
|        |                   |                         |            |
|        |                   |                         |            |
|        V                   |                         V            |
|    +------+                |                 +------------+       |
|    |  STB |          Tracker Protocol       |Mobile Phone|       |
|    +------+                |                 +------------+       |
|        ^                   |                         ^            |
|        |                   |                         |            |
|        |                   |                         |            |
|        |                   V                         |            |
|        |Peer Protocol  +---------+    Peer Protocol  |            |
|        +-------------> |    PC   |<------------------+            |
|                       +---------+                                 |
|                                                                    |
+--------------------------------------------------------------------+
```

Figure 4 Heterogeneous P2P Streaming Interactions with PPSP

7.3. CDN supporting streaming

This scenario is similar to use case 1 except that this is more like a M to N mapping while use case 1 is more often to be a case by case mapping. This reduces the case by case negotiation between the original provider and multiple CDN providers if otherwise proprietary protocols are used makes it easier for both sides to interoperate.

The interactions between the P2P streaming provider's tracker server and CDN surrogates as well as interactions between CDN surrogates are the same as a normal peer as shown in Figure 4.

PPSP can be used in:

1) Interface between CDN nodes and tracker. This is very useful for a small streaming provider who has no its own CDN surrogates and much money to distribute its stream worldwide.

2) New construction of CDN systems by PPSP. This can often occur for an operator or CDN vendor to build a P2P CDN system supporting streaming or file sharing applications with low cost.

```
   +-------------------------------------------------------------------+
   |                                                                   |
   |                          +------------------+                     |
   |            +------------>| Original Tracker |<----------+         |
   |            |             +------------------+           |         |
   |     Tracker|                ^              ^            |         |
   |    Protocol|         Tracker|              |Tracker     |Tracker  |
   |            |        Protocol|              |Protocol    |Protocol |
   |            |                |              |            |         |
   |            |                |              |            |         |
   |            v                v              v            v         |
   |      +------+ Peer    +------+            +------+    +------+    |
   |      | CDN1 |<------->| CDN1 |            | CDN2 |    | CND2 |    |
   |      | POP2 |Protocol | POP1 |            | POP1 |    | POP2 |    |
   |      +------+         +------+            +------+    +------+    |
   |         ^  ^                                           ^ ^        |
   |         |  |                                           | |        |
   |         |  | Peer Protocol                Peer Protocol| |        |
   | Peer    |  +-------------+              +--------------+ |Peer    |
   | Procotol|                |              |                |protocol|
   |         |                |              |                |        |
   |         |                |              |                |        |
   |         |                |              |                |        |
   |         v                v              v                v        |
   |      +------+ Peer    +------+    +---------+  Peer   +---------+ |
   |      | USA  |<------> | USA  |    |Caribbean|<------> |Caribbean| |
   |      | User1|Protocol | User2|    | User1   |Protocol | User2   | |
   |      +------+         +------+    +---------+         +---------+ |
   |                                                                   |
   +-------------------------------------------------------------------+
```

Figure 5 CDN Supporting P2P Streaming with PPSP

7.4. Hierarchical P2P Streaming Distribution with PPSP

The hierarchical P2P streaming distribution have many advantages over non-hierarchical conterparters such as better QoS(start-up latency and service interruption reduction[P2broadcast], higher throughput and lower packets drop ratio[Hybrid], topology-mismatch reduction and better management[AHLSS].

PPSP is useful for clustering the peers because there are abundant node information and content information exchange fetched in the message.

7.5. Serving Gateway/GGSN acting as Super Nodes assisting P2P streaming delivery in Cellular mobile environment

In cellular mobile environment, with the increase in bandwidth and mobile terminal capabilities, P2P streaming is better to be realized than before. Note that we don't have compulsory mobile peers. The network peers and WIFI peers are easier selected. GGSN, as the gateway for the cellular network to Internet, is more and more viewed as a promising place to add the cache functionality assisting P2P streaming services. Because it's deployed by the operators, the stability and storage size are better guaranteed than ordinary PC. It's more likely to select GGSN as the super nodes assisting the delivery. The interactions between serving gateway/GGSN and tracker, between different serving gateways/GGSNs and between serving gateway/GGSN and mobile terminal is explicated in Fig6.We name these kinds of serving gateway/GGSN as Mobile Supporting Super Nodes(MSSN).Note that if mobile terminals are not eligible to be a peer, it can use client/server mode for streaming service simply taking serving gateway/GGSN a source.

There are basically two scenarios in cellular networks:

1) Self operational P2P streaming services for mobile operators: PPSP is the suitable protocol for tracker-serving gateway/GGSN and serving gateway/GGSN-mobile nodes interaction. The serving gateway/GGSN can be both a super node or a complete proxy for different mobile terminals with different capabilities.

2) The 3rd party P2P streaming service with optimized localization by GGSN. When introducing a popular P2P streaming application like PPLive in the mobile network, serving gateway/GGSN can coordinate with the 3rd party trackers to cache the content without needing continuous update of the 3rd party protocols.

```
   +--------------------------------------------------------------------+
   |                                                                    |
   |                            Peer Protocol                           |
   |                         +--------------------+                     |
   |                         |                    |                     |
   |            ,--...-      |      ,--...-       |     ,--...-         |
   |          .'       '.    |    .'        '.    |   .'        '.      |
   |        /            \   V   /            \   V  /             \    |
   |        '  Cellular  +------+  Internet  +------+  Cellular     |   |
   |       |    Access   | MSSN |            | MSSN |   Access      /   |
   |       \   Network   +------+            +------+  Networks     /   |
   |       \              /^  ^ \             /     \              .'   |
   |        `.           / |  |  \           /       \            .'    |
   |         '.        .'  |  |   '+-------+'        `.         ./      |
   |           '------'    |  |    |Tracker|           `-------'        |
   |         Peer Protocol |  |    +-------+                            |
   |  +------+ /HTTP       |  | Tracker ^ Protocol                      |
   |  |Mobile|<------------+  +---------+                               |
   |  |Phone |<-------------------------+                               |
   |  +------+    (Tracker Protocol)                                    |
   +--------------------------------------------------------------------+
```

Figure 6 GGSN assisting P2P streaming delivery

8. Security Considerations

PPSP has a similar assumption in peer privacy as P2PSIP[ID.ietf-p2psip-base], i.e., all participants in the system are issued unique identities and credentials through some mechanism not in the scope of PPSP, such as a centralized server. Hence PPSP will not attempt a solution to these issues for P2P streaming networks in general. However PPSP have some unique security issues:

1) The content published by peers may not be checked by centralized certificating server. Therefore P2P streaming network may have the danger of malicious content distribution.

2) Content pollution is another common problem faced by P2P streaming.

3) Because there is a tracker who is critical to the P2P streaming systems. There is an increased probability that threats may involve launching attacks against the tracker.

PPSP may include some mechanisms to prevent malicious nodes from polluting the streaming content or launch attacks to the tracker. The protocol documents will contain a complete description of the security/privacy concerns of PPSP.

9. Acknowledgments

We would like to acknowledge the following who provided feedback or suggestions for this document: D. Bryan from Cogent Force; E. Marocco from Telecom Italia; V. Gurbani from AT&T; R. Even from Huawei; H. Zhang from NEC Labs, USA; C. Schmidt and L. Xiao from NSN; C. Williams from ZTE; V. Pasual from Tekelec; X. Zhang from PPlive; H. Deng from China Mobile; and J. Lei from Univ. of Goettingen.

10. Informative References

[Cisco] Approaching the Zettabyte Era by Cisco.

[PPLive] www.pplive.com

[PPStream] www.ppstream.com

[UUSee] www.uusee.com

[youtube] www.youtube.com

[tudou] www.tudou.com

[CNN] www.cnn.com

[Octoshape] www.octoshape.com

[ATT]http://mobile.sooyuu.com/Article/content/200905/2173150946 29281_1.shtml

[Sigcomm:P2P streaming]Challenges, Design and Analysis of a Large-scale P2P-VoD System,Yan Huang et al, Sigcomm08.

[RFC 5693] Application-Layer Traffic Optimization (ALTO) Problem Statement, E. Marocco et al, draft-ietf-alto-problem-statement-04

[Pando]www.pando.com

[CoolStreaming] CoolStreaming/DONet: A Data-Driven Overlay Network for Efficient Live Media Streaming, Xinyan Zhang et al,

[HPTP] HPTP: Relieving the Tension between ISPs and P2P, Guobin Shen et al,

[draft-zhang-ppsp-protocol-comparison-measurement-00]www.ietf.org/internet-drafts/draft-zhang-ppsp-protocol-comparison-measurement-00.txt

[GENI] www.geni.net

[FIND]www.nets-find.net

[draft-zhang-ppsp-dsn-introduction-00]www.ietf.org/internet-draft/draft-zhang-ppsp-dsn-introduction-00.txt

[MobileTV] MobileTV,Turning in or switching off, Arthur D. Little

[Computer Networks: Traffic] Traffic analysis of peer-to-peer IPTV communities, Thomas Silverston et al, Computer Networks, 53 (2009) 470-484

[Survey]A survey on peer-to-peer video streaming systems Yong Liu et al, Peer-to-Peer Netw Appl (2008) 1:18-28,Springer.

[draft-zhang-alto-traceroute-00] www.ietf.org/internet-draft/draft-zhang-alto-traceroute-00.txt

[P2PStreamingSurvey] Zong, N. and X. Jiang, "Survey of P2P Streaming", IETF PPSP BoF, November 2008.

[Challenge] Peer-to-Peer Live Video Streaming on the Internet: Issues, Existing Approaches, and Challenges, Bo Li et al, IEEE Communications Magazine, June 2007(94-99).

[CDN+P2P]Efficient Large-scale Content Distribution with Combination of CDN and P2P Networks,Hai Jiang et al,International Journal of Hybrid Information Technology, Vol.2, No.2, April, 2009.

[Peering CDN] A Case for Peering of Content Delivery Networks, Rajkumar Buyya1 et al, http://dsonline.computer.org/portal/site/dsonline/menuitem.9ed3d9924aeb0dcd82ccc6716bbe36ec/index.jsp?&pName=dso_level1&path=dsonline/2006/10&file=o10003.xml&xsl=article.xsl&.

[P2broadcast] P2broadcast: a hierarchical clustering live video streaming system for P2P networks, De-kai Liu et al,International Journal of Communication Systems,Volume 19,Issue 6.

[Hybrid]Hybrid Overlay Networks Management for Real-Time Multimedia Streaming over P2P Networks, Mubashar Mushtaq et al, Lecture Notes in Computer Science, Volume 4787/2007.

[AHLSS]AHLSS: A Hierarchical, Adaptive, Extendable P2P Live Streaming System, Runzhi Li et al, International Journal of Distributed Sensor Networks, Volume 5, Issue 1 January 2009.

[ComCast]http://www.afterdawn.com/news/article.cfm/2008/05/20/comcast_invests_in_p2p_streaming_startup

[Johan Pouwelse]http://newteevee.com/2008/07/24/open-source-p2p-streaming-getting-ready-to-disrupt-cdn-business-models/

[CNTV] news.xinhuanet.com/2010-06/30/c_12281703.htm

[CNNIC] http://it.sohu.com/s2010/cnnic25/

[ID.ietf-p2psip-base] Jennings, C., Lowekamp, B., Rescorla, E., Baset, S., and H. Schulzrinne, "REsource LOcation And Discovery (RELOAD)Base Protocol", draft-ietf-p2psip-base-08.

[Akamai] Cheng Huang , Angela Wang , Jin Li , Keith W. Ross, Understanding hybrid CDN-P2P: why limelight needs its own Red Swoosh, Proceedings of the 18th International Workshop on Network and Operating Systems Support for Digital Audio and Video, May 28-30, 2008, Braunschweig, Germany .

Author's Addresses

Yunfei Zhang

China Mobile Communication Corporation

zhangyunfei@chinamobile.com

Ning Zong

Huawei Technologies Co., Ltd.

zongning@huawei.com

Gonzalo Camarillo
Ericsson
Gonzalo.Camarillo@ericsson.com

James Seng
PPLive
james.seng@pplive.com

Richard Yang
Yale University
yry@cs.yale.edu