A Measurement-Based Study of Xunlei

Min Zhang
Beijing Jiaotong University
mia.minzhang@gmail.com

Changjia Chen
Beijing Jiaotong University
changjiachen@sina.com

Nevil Brownlee
CAIDA, UC San Diego
nevil@caida.org

1. INTRODUCTION

Peer-to-Peer applications may well account for a majority of modern Internet traffic[5][6]; meanwhile, new P2P applications emerge rapidly on the Internet. In order to gain better understanding of the current Internet, this paper studies Xunlei, an upcoming P2P application, currently mainly used in China.

Nowadays, Xunlei is one of the most popular file-sharing applications in China, integrating other open P2P file-sharing systems (BitTorrent, eDonkey, Kad) and even IPTV. According to records on wikipeida[1], over eighty million users installed Xunlei and its web site attract over fifty million hits per day; 1,880 million are reported on Xunlei's website[3]. We believe that Xunlei deserves international research interest, even though it does not provide an official English version (yet) and only runs on Windows machines. It was shown that Xunlei gained significant popularity in regions outside China as well, mainly due to Chinese migrants [6]. Furthermore, translated and ported versions are likely to be available soon, eventually allowing Xunlei to spread worldwide.

To our best knowledge there is no published work on Xunlei. This might partly be due to limited international attention, but also due to Xunlei's proprietary nature and its use of data encryption, which makes analysis cumbersome. In this paper, we take a first step to study Xunlei. First, some experiments are designed for exploring Xunlei's architecture. By analyzing raw data on packet-level with Wireshark[2], we not only learn about Xunlei's server structure and how its client works, but also gain some preliminary empirical results for its downloading process. These analysis results are expected to provide new insights to researchers and application designers, and also to give a better understanding of the Chinese Internet.

2. EXPERIMENTS

We carried out two kinds of experiments. One is focused on Xunlei's architecture. Since there are no formal documents on its protocol and mechanism, we start by investigating how the system operates. For clarity purposes, we divide the client-server communication into three phases: login, idling and downloading. Login procedure happens when a client is launched, and idling procedure means certain things a client does without file searches or file downloads. Downloading includes request sources and whole file transmission.
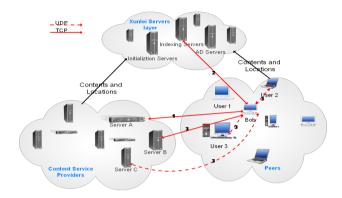
Figure 1: Diagram of Xunlei System

For each phase, we sniff all packets using Wireshark. After careful packet level analysis, we intend to figure out the client-server communication and the infrastructure of Xunlei server system.

The other experiment is for exploring Xunlei's behavior in downloading file from web sites. First, we choose some files listed in the downloading rankings and portal sites as our experiment subjects. Next, we download files from sites respectively; and each downloading progress is recorded by Wireshark. Our analysis is based on information extracting from source and destination IP addresses, protocol, data bytes on packet level.

3. RESULTS

We consider that the Xunlei system consists of three parts: Xunlei servers, Content Server Providers, and Peers. Xunlei servers collect resource information from Content Server Providers and Peers. In Fig.1, a general example for downloading is taken. If a user named "Bob" launched a download request to Server "A", Xunlei's indexing servers could tell "Bob" resources (other locations where he also can download the file), and then "Bob" would connect to them respectively for downloading data with multi-source technology. In general, Xunlei supports several protocols, including HTTP, FTP, BitTorrent, and its propriety ones. Data is transferred among peers via UDP. Both UDP and TCP are used in communicating with content server providers.

3.1 Server

Though both consist of servers and clients, Xunlei servers do differ from eDonkey's client server architecture[4]. First,

Table 1: Interactions During Idling

Interactive server	Packet type	Interval (seconds)	Packet content
keep-alive server	ICMP	10	40bytes (meaningless)
nodes server	UDP	45	MAC address, net type
hub5pc.sandai.net	UDP	120	MAC address, client IP and port
hub5u.sandai.net	UDP	300	version,MAC address,net type,etc.

Figure 2: Numbers of sources according to different types in downloading

the number and addresses of eDonkey servers change frequently, while those of Xunlei servers are steady according our observation. According to our study, forty-six IP addresses are observed as Xunlei servers, which are mainly located in three provinces of China. Moreover, Xunlei servers are more complicated and systematical than those of eDonkey. In the eDonkey network, each server is mainly in charge of indexing files for clients; but xunlei servers cooperate effectively with clear-cut divisions of work, such as client initialization, advertisement, file index, virus scan, pictures store, etc. For example, on startup a Xunlei client connects to one specific server. A number of further servers are then hosting and presenting advertisements. Finally, yet another server server is responsible for finding target files.

3.2 Client

We will now describe the analysis results for Xunlei's client-server communication:

- **Login:** When establishing connection to Xunlei network, the client usually has a login sequence with Xunlei servers. However, there is no space for us to show the login procedure in detail.

- **Idling:** During the idling state, Xunlei client keeps four kinds of regular interactions (Table 1) with servers besides updating advertisements. We consider these are heartbeat packets for updating and reporting client status.

- **Downloading:** When a user clicks a link to download files from web site, the client launches TCP connections with a specified server for querying the target files. After receiving its reply, the client downloads files from numerous sources using multi-source technology.

3.3 Evaluation of Downloading

With understanding of the Xunlei server and client, we move on to evaluate its downloading to investigate the cause of its popularity. Here we only give analysis results of downloading no-media files from web sites.

- Besides the clicked web site, we observe that Xunlei client downloads data from many sources. Servers of clicked web site are defined as clicked servers. We classify sources into three groups: clicked server, unclicked server, peer. Fig.2 shows different sources distribution based on forty experiments from two portal sites; furthermore, according our observation, all three types of sources provide corresponding data to clients.
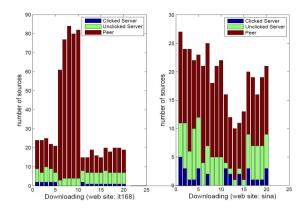
- Since a Xunlei client downloads files from serval sources, we doubt whether it downloads too much reduplicate data. In this way, by comparing our downdata and the target file size, we calculate data redundancy for each download. The results (no sufficient space for numbers here) show that xunlei's multi-sources manager works effectively for downloading from numerous sources.

- The behavior of downloading files from unclicked servers is inappropriate, as it exploits their advertisement profits, page view hits, etc. However, we consider it as a strategy for utilizing Internet resources, which also balances the loads among servers.

4. FUTURE WORKS

So far, our preliminary results give a description of Xunlei's architecture and its client-server communication. Based on our empirical data, Xunlei client downloads data from numerous sources including servers and peers with multi-source technology. It also provides us some hints for designing a business-driven information integration for making good use of the Internet, which will be extended and advanced formally; meanwhile, considering the other aspects of Xunlei (such as IPTV, bulk-file sharing, compatibility with uTorrent and eMule), we will continue to explore it, making a full-scale study of Xunlei.

5. REFERENCES

[1] http://en.wikipedia.org/wiki/xunlei.
[2] http://www.wireshark.org/.
[3] http://www.xunlei.com/hr.html.
[4] O. Heckmann and A. Bock. The edonkey 2000 protocol. *Technical Report KOM-TR-08-2002, Multimedia Communications Lab, Darmstadt University of Technology*, December 2002.
[5] W. John, S. Tafvelin, and T. Olovsson. In *Trends and Differences in Connections Behavior within Classes of Internet Backbone Traffic*. PAM, 2008.
[6] H. Schulze and K. Mochalski. Internet study 2008/2009. *Technical Report, IPOQUE*.