International Journal of Global Engineering (IJGE) E- ISSN: 2456-3099

VOL 3 ISSUE 1 (2018) PAGES 66 - 72

Received: 17/07/2018. Published: 22/08/2018

UNDERSTANDING INSTANT MESSAGING TRAFFIC CHARACTERISTICS USING MSP 430

M.Rajkumar ,T.Karthik, M.Arun kumar

Department of Electronics and Communication Engineering

Dhanalakshmi Srinivasan College of Engineering, Coimbatore

Abstract-Instant messaging (IM using) has become increasingly popular due to its quick response time, its ease of use, and possibility of multitasking. It is estimated that there are several millions of instant messaging users who use IM for various purposes: simple requests and responses, scheduling face to face meetings, or just to check the availability of colleagues and friends.

Despite its popularity and user base, little has been done to characterize IM traffic. One reason might be its relatively small traffic volume, although this is changing as more users start using video or voice chats and file attachments. Moreover, all major instant messaging systems route text messages through central servers. While this facilitates firewall traversal and gives instant messaging companies more control, it creates a potential bottleneck at the instant messaging servers. This is especially so for large instant messaging operators with tens of millions of users and during flash crowd events. Another reason for the lack of previous studies is the difficulty in getting access to instant messaging traces due to privacy concerns.
In this paper, we analyze the traffic of two popular instant messaging systems, AOL Instant Messenger (AIM) and MSN/Windows Live Messenger, from thousands of employees in a large enterprise. We found that most instant messaging traffic is due to presence, hints, or other extraneous traffic. Chat messages constitute only a small percentage of the total IM traffic. This means, during overload, IM servers can protect the instantaneous nature of the communication by dropping extraneous traffic. We also found that the social network of IM users does not follow a power law distribution. It can be characterized by a Weibull distribution. Our analysis sheds light on instant messaging system design and optimization and provides a scientific basis for instant messaging workload generation

Introduction

Instant messaging (IM) has become increasingly popular. It is estimated that there are several millions of instant messaging users all over the world. Teenagers use instant messaging to keep

International Journal of Global Engineering (IJGE) E- ISSN: 2456-3099
VOL 3 ISSUE 1 (2018) PAGES 66 - 72
Received: 17/07/2018. Published: 22/08/2018

in touch with their friends and families, while corporate users exchange IMs to discuss work. Com- pared to other methods of communication, instant messaging offers several advantages: its almost synchronous nature makes it ideal for simple requests and responses. It provides presence and event notifications which make it easy to keep track of the availability of colleagues and friends ("buddies" in IM terminology). In addition, most IM systems today in- corporate support for voice or video chats as well as file transfer, making it an integrated environment for a wide va- riety of communication needs. The popularity of IM is expected to continue increasing in the foreseeable future.

Despite its huge popularity and user base, little work has been done to understand the workload generated by this important application. The limited existing work in this area focuses mostly on understanding the social behavior of instant messaging users. For example, the authors of [10] interviewed twenty IM users and found that IMs are often used to negotiate the availability of co-workers who may then switch to a different media (e.g., phone) for complex discussions. Work in [2] studies IM usage from sixteen teenagers and found different social behaviors between high school and college students. Since the bulk data in those studies came from surveys and interviews, their scopes are inevitably limited to small user samples (often fewer than fifty) and to the subjective descriptions of those users. The only work we are aware of that studies IM usage in a relatively large scale is [5] which collected IM logs from 437 users. However, they developed a specialized instant messenger, called "Hubbub".

Background on Instant Message protocols

In this section, we provide the basic background on IM protocols, focusing on two popular IM systems, AIM and MSN. Each IM user has a unique *screen name* in the system and a list of *buddies*. When a user is online, the IM sys- tem notifies the presence of her buddies to her as well as notifying her presence to her buddies. Most instant messaging systems are client-server based and route chat messages through central servers. Chat messages are typically small, and the IM system may limit the size of each message. Be- sides the chat messages, the IM client software also generates certain *hint* messages when a user is typing or editing a message. This feedback can be useful, for example, to prompt her buddy to wait for her response. IM users can also select a favorite picture as their *icons* which will be displayed when chatting with their buddies. In both AIM and MSN Messenger, a user can send messages to a buddy only when the buddy is online. In contrast, some IM systems such as QQ support offline messaging where messages sent to offline users are buffered at the servers and delivered when those users come online. Besides text-based messaging, most IM systems today support voice/video chat and file transfers in an integrated environment. While IM servers are involved in the initial establishment of a voice/video call or a file transfer, subsequent communications are often conducted between the two end users directly in a peer-to-peer fashion. However, when the caller and callee are both shielded by a firewall or NAT router, their communication must be relayed by IM servers. The focus of this study is the communication between IM clients and servers. We do not analyze peer to peer IM traffic.

Server Architecture

Both AIM and MSN use an asymmetric architecture. In AIM, an authentication server processes user logins and redirects users to messaging servers, called BOS .

With the MSN messaging system, a user contacts a dispatch server (DP server) first, which redirects the user to a notification server (NS server). The NS server is responsible for notifying the user of various events, such as buddy presence, instant message calls, and email arrivals. Similar to a user's connection to her BOS server in AIM, a user's connection to her NS server is persistent throughout her en- tire online time. When one user wants to chat with another, it sends a request to the NS server, which then redirects the user to a switchboard server (SB server). Correspondingly, the callee of an IM chat receives a notification from her NS server, and then is redirected to the same SB server. All conversations between the two users are then relayed by the SB server. The advantage of this architecture is that an instant message is relayed by a single server (i.e., the SB server). In contrast, an instant message in AIM is relayed by two servers: the BOS server of the caller and that of the callee. Another difference is: an AIM user sends and receives all chat messages over a single connection to her BOS server, while a MSN user communicates with different buddies over separate connections to SB servers. When the two MSN users become silent for a long time without sending any message, they are automatically logs off from the SB server, but not from their NS servers. Group chats are conducted in the same way as two-user chats in MSN open source implementation of IM clients exist, they typically only implement a particular (often out of date) version of the protocol and only a subset of the functionality. Although an IM client can talk to the server using the protocol it chooses, our IM sniffer in the middle of the network must be able to understand various versions of IM protocols and be able to parse their messages correctly. We developed sniffer software to collect IM packets online without violating user privacy. Written with C and pcap libraries, the sniffer monitors packets that are passed from the OS kernel, identifies IM packets and hashes privacy sensitive information, such as user screen names and chat messages, then dumps the hashed IM packets in pcap format.

Experimental Setup

We have installed a sniffer machine at the Internet gate- way of a large enterprise network with more than four thousand employees. The sniffer machine is equipped with an additional network card connected to the network switch of the gateway, which forwards packets it sends to or receives from the Internet to the sniffer network card by port mirroring. We have reverse engineered two popular instant messenger protocols, AIM and MSN Messenger. While some In August 2006, AOL released its new instant messenger software, AIM Triton, with new protocols (correspondingly, the traditional AIM software and protocol are called AIM Classic). We have not reverse- engineered this new protocol. However, the traffic of AIM Triton only since in this case the IM port is not used as a server port (all IM port numbers in our study are greater than 1024).

International Journal of Global Engineering (IJGE) E- ISSN: 2456-3099

VOL 3 ISSUE 1 (2018) PAGES 66 - 72

Received: 17/07/2018. Published: 22/08/2018

All dumped packets in pcap format include timestamps when they are captured. Due to the online packet reconstruction, the dumped packets may be out of order. When a dumped file is closed, post-processing is performed to sort the packets by their timestamps.

Workload Description

We have collected the AIM and MSN Messenger packets with our IM sniffer software for nearly one month. The workload covers a 26-day period from 2006-10-14 20:34 to 2006-11-06 20:43. Our workload includes 469 AIM users and 408 MSN users, with more than 20,000 users conversations and millions of IM messages. We have also collected the IP and TCP headers of three other popular instant messenger traffic: Yahoo Messenger, GTalk/Jabber, and IRC Chat. To our best knowledge, this is the largest scale instant messaging measurement to date.

Table 1 shows an overview of IM traffic in this workload based on the TCP/IP header analysis. To avoid false positive

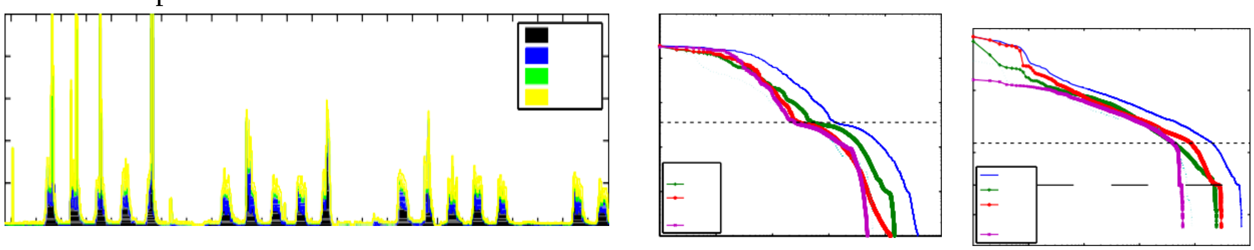
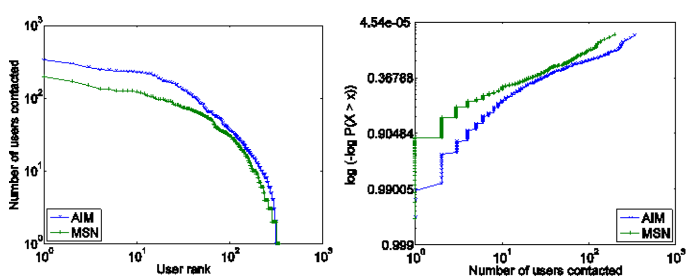


Figure 12. Duration of each conversation



(a) Log-log plot (b) Weibull plot

Figure 13. Number of buddies for IM users

Instead, the CCDF follows a Weibull distribution. Figure 11 shows the CDF distribution of chat message intervals for AIM and MSN.

Figure 12 shows the distribution of IM conversation du- ration for AIM and MSN users. The conversation duration is the duration from the first IM chat message to the last IM chat message in the conversation (conversations less than 10 seconds are excluded). The distributions of conversation duration for AIM and MSN look different.

International Journal of Global Engineering (IJGE) E- ISSN: 2456-3099
VOL 3 ISSUE 1 (2018) PAGES 66 - 72
Received: 17/07/2018. Published: 22/08/2018

User level analysis

Buddy lists reflect the social network of IM users. Al- though we cannot rebuild the contact network of IM systems with only a small subset of IM users, the rank distribute not characterize the social network of IM users. Although AIM has an upper bound of the number of buddies for a user, this upper bound (300) is not reached for AIM users in our study.
Since the node degree distribution in an IM network is not power law, the “super nodes” in such a network are not as important as those in peer-to-peer networks. Actually, although worms and viruses in IM networks have been re- ported for a long time, they have not been a big threat to the Internet, possibly because the propagation of virus on IM networks is slow due to the network topology.

Figures 14 shows the number of buddies a user chats with in our workload. For AIM systems, a user chats with only 1.9 users on average, about 7% buddies in its buddy list. For MSN systems, a user chats with 5.5 users on average, about 25% buddies in its buddy list. In our network, AIM users seem less active than MSN users, possibly due to different user population. need to investigate In either case, a user only contacts a small percentage of users in her buddy list during the one month workload. Thus, the social network of IM users has a small working set. For exam- ple, the 406 MSN users chat with only 2053 out of the 8456 buddies in their buddy lists.
Figure 15 shows the number of conversations for AIM and MSN users. The number of conversations for AIM users is much smaller than that for MSN users: as shown in the CDF figure, 10% AIM users have more than 30 conversations, while 18% MSN users have more than 100 conversations. In both AIM and MSN systems, a small fraction of users contribute most conversations, and consume most server resources of the IM system

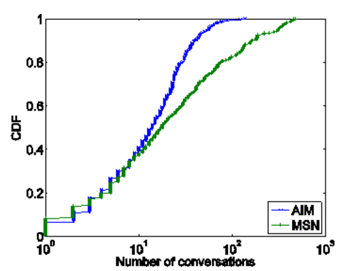


Figure 15. Number of conversations of IM users.

Other related work

In addition to instant messaging systems, work [13] presents a measurement study on the Short Message Ser- vice (SMS) in cellular networks. Work [1] presents a measurement study on an Internet chat system, which can be regarded as an early instant messaging system and has many similarities with modern IM systems.

International Journal of Global Engineering (IJGE) E- ISSN: 2456-3099
VOL 3 ISSUE 1 (2018) PAGES 66 - 72
Received: 17/07/2018. Published: 22/08/2018

More remotely related are workload characterization studies on peer to peer systems. Gummadi et al. characterized the active session length, download sizes, and evolution of object popularity in KaZaa traffic collected from a university campus [3]. More recently, Klemm et al. pro- posed a synthetic workload model for the query behaviors in large scale peer to peer systems distributed across three continents [6]. It observed that the distribution of query popu- larity has a "flattened head" when aggregated over multiple days.

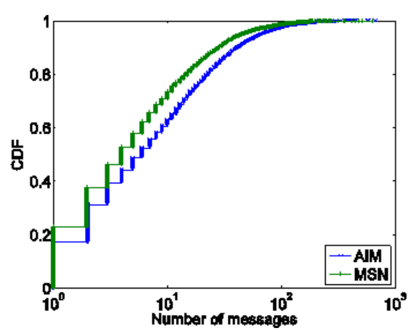
Lindemann et al. proposed a simulation model for disseminating presence information in mobile ad hoc networks [7], which can be employed for workload generation of IM traffic as well.

- Service control messages including those for log in and log out, server redirection, application level keep alive,etc.
- Other management messages

Figure 8 shows the number of messages and traffic volume for AIM and MSN. For both protocols, chat messages account for only a small percentage of IM messages, either by number or by volume. The number of hint messages is greater than the number of chat messages, and the number of presence messages is greater than that of hint messages. For AIM, the volume of hint messages is smaller than that of chat messages probably due to the binary nature of the protocol. In contrast, the protocol for MSN Messenger is text- based, and the message headers account for the majority of a hint message. The volume of binary messages accounts for the majority of MSN Messenger traffic, because it con- tains file transfer and voice/video chat traffic. However, we cannot distinguish them from the binary messages for user icons because they have the same message type. The num- ber of those messages is not very large, but the packet sizes are much larger than other messages.

We also observe that the chat message size distributions of AIM and MSN Messenger are similar, as shown in Figure

Chat messages in AIM tend to be larger because most



Conclusion

Understanding the characterization of instant messaging traffic is essential to its system design and workload generation. This paper analyzes the traffic of the two most popular

Received: 17/07/2018. Published: 22/08/2018

instant messaging systems from thousands of employees in a large enterprise. In the future, we plan to extend the scope of our study by analyzing traces from other user population.

1. Acknowledgments

We are very grateful to William Cornejo, Erich Nahum, Charles Wright, Arup Acharya, and Xiping Wang for manyinsightful discussions on this work. We would like to thank the anonymous reviewers for comments on an early draft of the work.

2. References

[1] C. Dewes, A. Wichmann, and A. Feldmann. An analysis of internet chat systems. In Proc. of ACM SIGCOMM IMC, Oct. 2003.
[2] R. E. Grinter and L. Palen. Instant messaging in teen life. In Proc. of ACM Computer Supported Cooperative Work (CSCW), Nov. 2002.
[3] K. P. Gummadi, R. J. Dunn, S. Saroiu, S. D. Gribble, H. M. Levy, and J. Zahorjan. Measurement, modeling, and analysis of a peer-to-peer file-sharing workload. In Proc. of ACM Symposium on Operating Systems Principles (SOSP), Oct. 2003.
[4] E. Isaacs, A. Walendowski, S. Whittaker, D. J. Schiano, and C. Kamm. The character, functions, and styles of instant messaging in the workplace. I n Proc. of ACM Computer Supported Cooperative Work (CSCW), Nov. 2002.
[5] Klemm, C. Lindemann, M. K. Vernon, and O. P. Wald- horst. haracterizing the query behavior in peer-to-peer file sharing system. In Proc. of ACM SIGCOMM IMC, Oct. 2004.
[6] C. Lindemann and O. P. Waldhorst. Effective dissemination of presence information in highly partitioned mobile ad hoc networks. In Proc. IEEE SECON, 2006.
[7] M. Mannan and P. C. van Oorschot. On instant messag- ing worms, analysis and countermeasures. In Proc. of ACM WORM, Nov. 2005.
[8] C. D. Morse and H. Wang. The structure of an instant mes- senger network and its vulnerability to malicious codes. In Proc. of ACM SIGCOMM, Aug.2005.
[9] B. A. Nardi, S. Whittaker, and E. Bradner. Interaction and outeraction: Instant messaging in action. In Proc. of ACM Computer Supported Cooperative Work (CSCW), Dec. 2000.
[10] D. Stutzbach and R. Rejaie. Understanding churn in peer- to-peer networks. In Proc. of ACM SIGCOMM IMC, Oct. 2006.
[11] M. M. Williamson, A. Parry, and A. Byde. Virus throttling for instant messaging. Technical Report HPL-2004-81, HP Laboratories, May 2004.
[12] P. Zerfos, X. Meng, and S. H. Wong. A study of the short message service of a nationwide cellular network. In Proc. of ACM SIGCOMM IMC, Oct. 2006.

