Proceedings of the 2007 IEEE Symposium on
Computational Intelligence and Data Mining (CIDM 2007)

Session recognition and Bandwidth Guarantee for Encrypted Internet Voice Traffic：Case Study of Skype

Jian-Hong Wang[1, 2] J.-Y. Pan[1] Yi-Chi Cheng[1]
[1] Department of Communication Engineering, National Chung Cheng University
[2] Department of Information Engineering, Chung Chou Institute of Technology
wanghong@dragon.ccut.edu.tw

Abstract--**Under limited bandwidth, how to improve quality of service in VoIP (Voice over Internet Protocol) is a significant problem of the network traffic today. However, some protocol of VoIP traffic such as Skype is encrypted; with this problem, the network administrator can't manage the VoIP bandwidth and result in time delay or loss of voice packets.**

The purpose of this study is to improve VoIP Bandwidth Guarantee by identifying encrypted Skype session. First, We use the Pcap (Packet Capture library) to capture session characteristics from the information about established connections in the network. Next, K-means is used to do clustering analysis and pattern recognition. After then, we can recognize the Skype session. Finally, we can feed the information of encrypted Skype session which has been identified into the Bandwidth management system. So that, we can improve quality of Bandwidth Guarantee in Skype.

Keywords: Encrypted VoIP, Skype session identification, K-means, pattern recognition.

I. INTRODUCTION

Voice over Internet Protocol (VoIP) is a popular implementation for transmitting real-time voice packet over the internet. But the heavy traffic of real-time voice packet transmitted through limited bandwidth often result in time delay or loss of voice packets and affect the quality of service.

Traditional VoIP bandwidth management system can understand only public protocols such as Session Initiation Protocol (SIP) and H.323 to obtain related bandwidth management information. If we can know the port number used by the VoIP traffic in advance, then we can assure the quality of service and manage the usage of bandwidth.

Private and encrypted protocols (such as Skype voice packets) will complicate the management of network bandwidth. Hence, the bandwidth cannot be guaranteed beforehand and quality of service cannot be provided.

The main reason that we choose Skype as a basis for this study is that it is the most used VoIP software. There are over 100 millions of Skype users in the world. And the number of Skype users in Taiwan is about 3.5 millions which is approximately 27% of the total VoIP users in Taiwan[14][5].

Due to that the protocol of Skype is encrypted, we are not able to apply traditional bandwidth management of VoIP system to achieve bandwidth administration over Skype. Therefore, this research focuses on the session recognition of encrypted Skype protocol. After P-Cap（Packet Capture library）gets the feature value of Skype session. Next, we use K-means (K-means is one of the simplest unsupervised learning algorithms that solve the well known clustering problem.[22])method to analyze clusters (Data clustering is a common technique for statistical data analysis, which is used in many fields, including machine learning, data mining, pattern recognition.[24])and identify patterns to assure Skype bandwidth and the quality of service of VoIP.

The remainder of this paper is organized as follows: Section II introduces the system framework of this study; Section III explains P-cap and K-Means Algorithm; Section IV explains experiment environment and presents system performance. Finally, we make a conclusion in Section V.

II. SYSTEM FRAMEWORK

There are five parts in this system architecture: session analysis module, Admin User Interface module (Admin UI), bandwidth allocation module, traffic control command module and quality of service module provided by Linux. The framework is illustrated in Fig. 1.

A. Session analysis module

The key work of this module is to detect packets in the system work. The session analyzer can capture the connection status via P-Cap. The P-Cap can get the information of each packet, including: protocol, source IP address, destination IP address, source port number, destination port number, packet size and transmission speed of packets per second. The information of each packet will be processed by cluster analyzer using data-grouping and classification methods. If a packet belongs to Skype session, bandwidth session module will be notified and the bandwidth will be guaranteed.

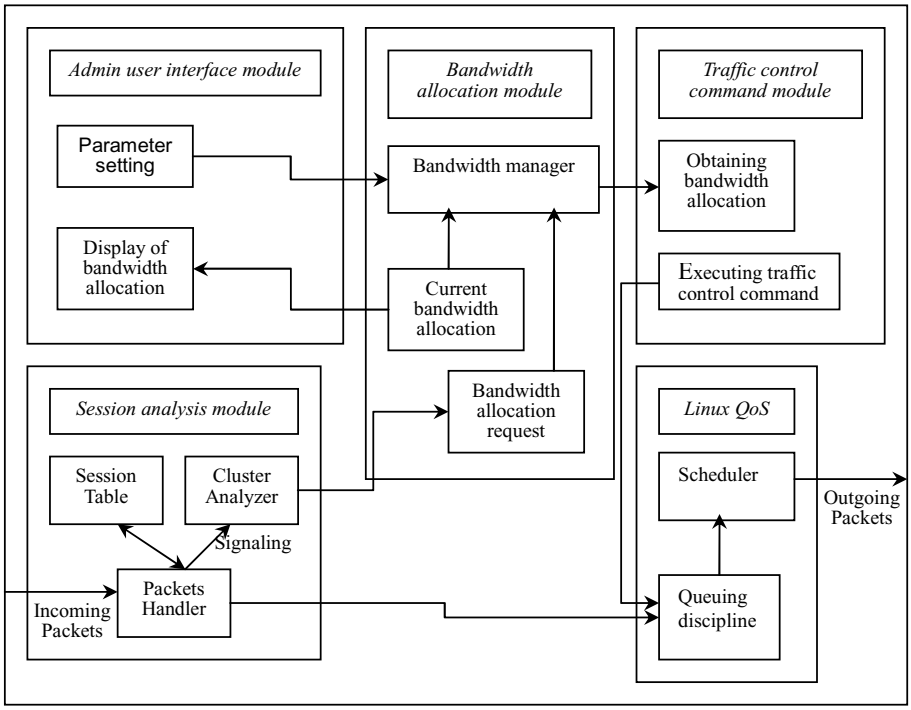

Fig. 1. System architecture[1]

B. Admin user interface module

Administrator can set up bandwidth allocation of Skype voice session through the Admin user interface. In addition, Admin user interface module can provide the display interface of bandwidth allocation for the system administrator to track the state of bandwidth in use that is saved in the bandwidth allocation module.

C. Bandwidth allocation module

Bandwidth allocation module is like the central nerves of the system. It develops inter-connection relationship with the other modules. It receives the bandwidth allocation parameters, such as bandwidth of each Skype voice session from the Admin UI module. Dynamic Bandwidth Allocation (DBA) is a technique by which traffic bandwidth in a shared telecommunications medium can be allocated on demand and fairly between different users of that bandwidth. Essentially, it is bandwidth management or is also sometimes known as statistical multiplexing. Where the sharing of a link adapts in some way to the instantaneous traffic demands of the nodes connected to the link.[29] When the connection of Skype voice session is established, we can decide whether the bandwidth of Skype voice session is allocated. When the connection of Skype voice session goes down, we can allocate the bandwidth of Skype voice session to other user. It computes the required bandwidth information for querying and maintenance in the future and allocates the dynamic bandwidth through the flow command module.

D. Traffic control command module

The main tasks of traffic control command module are obtaining bandwidth allocation status and executing traffic control command. The iproute2 tool is a collection of utilities for controlling TCP/IP networking and traffic control in Linux. Most network configuration systems make use of ifconfig and thus provide a limited feature set. The /etc/net project aims to support most modern network technologies, as it doesn't use ifconfig and allows a system administrator to make use of all iproute2 features, including traffic control.[28] The outcome of this model feed to the queuing discipline of Linux Quality of Service module.

E. Linux Quality of Service module

Linux Quality of Service module is provided by the Linux kernel. It supports bandwidth administration to meet the administrator's needs via different queuing rules, classification

and different scheduling mechanism.

III. THE P-CAP AND K-MEANS ALGORITHM

Packet Capture library (P-cap) is the industry-standard tool for link-layer network access in Windows environments: it allows applications to capture and transmit network packets bypassing the protocol stack, and has additional useful features, including kernel-level packet filtering, a network statistics engine and support for remote packet capture. P-Cap consists of a driver, that extends the operating system to provide low-level network access, and a library that is used to easily access the low-level network layers. This library also contains the Windows version of the well known libpcap Unix API.[21]

The training traffic data of this research is captured from the Electrical Engineering department of Chung-Cheng University. Firstly, we use the commands "TCPdump" and "TCPreplay" to record and play the flow, and extract the feature value of the packet to determine whether they are Skype voice sessions. At step two, we use clustering analysis tool, K-Means, to classify the feature values. If the feature values of session are identified as Skype session, the bandwidth of this session will be guaranteed.

K-means is one of the simplest unsupervised learning algorithms that solve the well known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed a priori. The main idea is to define k centroids, one for each cluster. These centroids should be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest centroid. When no point is pending, the first step is completed and an early groupage is done. At this point we need to re-calculate k new centroids as barycenters of the clusters resulting from the previous step. After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new centroid. A loop has been generated. As a result of this loop we may notice that the k centroids change their location step by step until no more changes are done.[22]. The main purpose of K-means is to find out the significant feature values in numerous variables to form clustering centers. We must first determine the number of clusters needed for analysis. A large number of clusters do not guarantee good results. Hence, we need to choose the most appropriate size of clusters for it is a factor influencing the outcome. The clustering centers can be selected randomly or obtained by training.

At step 3, K-Means Algorithm calculates the distance between each information point and the clustering center and computes the sum to be the error function. K-Means Algorithm looks for the clustering centers repeatedly and computes the error function until the smallest error value is obtained. The computing method of K-Means Algorithm is Distance Square Error. Assuming that there are c groups and there are n_k data in e_k $\{x_1, x_2, \ldots, x_{nk}\}$, if the clustering center is y_k, then the Square Error of e_k can be defined as following equation.

$$e_k = \sum_{i=1}^{n} |x_i - y_k|^2 \qquad (1)$$

x_i is the information point of cluster k. And the total square error E equals the total of square error of each cluster. It is also called the Error Function or Distortion of individual cluster.

$$E = \sum_{k=1}^{n} e_k \qquad (2)$$

Combining equation 1 and 2, we get the E value：

$$E = \sum_{k=1}^{n} \sum_{i=1}^{n} |x_i - y_k|^2 \qquad (3)$$

We get the most suitable clustering centers from collected training data and determine the most appropriate clustering number to get the minimal E value. After obtaining the suitable clusters, in conjunction with previous cluster analysis, we can determine the closest clusters. If the K-means algorithms find the set of Skype clusters, we can understand the features of Skype sessions. If the features of packet size and number of transmitted packets per second belong to Skype sessions', we treat this session as Skype session.

The main advantages of this algorithm are its simplicity and speed which allows it to run on large datasets. Its disadvantage is that it does not yield the same result with each run, since the resulting clusters depend on the initial random assignments. It maximizes inter-cluster (or minimizes intra-cluster) variance, but does not ensure that the result has a global minimum of variance.[27]

IV. EXPERIMENT ENVIRONMENT AND EFFECTIVENESS ANALYSIS

A. Testing platforms and equipment

This research uses ADI Coyote Gateway Reference Design Based on the Intel® IXP425TM Network Processor and Monta Vista® Linux® Professional Edition 3.1 to be the development platform. The testing environment architecture is as Fig. 2. Intel® IXP425™ provides one WAN port and four LAN ports for experiment. We set LAN as NAT (Network address translation, a system for reusing IP addresses. [23]) mode to be used by households and place one 10 Mbps hub before connecting to Internet. This

is for emulating the bandwidth of typical ADSL used in the household. Meanwhile, we use Skype version 2.0 in our experiment.

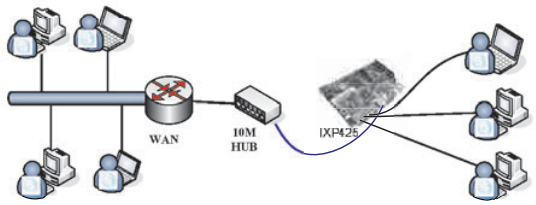

Fig. 2. Testing environment architecture

B. Recognition rate analysis

In this research, we use the internet traffic of the Electrical Engineering department of Chung-Cheng University to be the training data. Training data is used to configure our system and makes it more accurate before using the testing data of the internet. During the experiment process, we find that the simpler the environment, the fewer the clusters are needed. We can have a good recognition rate with 2 to 3 clusters. The more complex the network environment, the more clusters there should be to allow internet traffic be allocated in different clusters and increase identification rate. Hence, it has to go through the training process to obtain the suitable number of clusters. The suitable number of clusters can result in higher recognition rate.

We experiment simple Skype environment in comparison with complex environment to test the recognition rate of this system. In a pure Skype environment, there are only Skype voice packets. On the other hand, in a complex environment there are Peer-to-Peer traffic and other voice communication software for tests. Table I and Table II represent two respective experiment environments. Pure Skype environment is located in a clean sub-network that only Skype traffic is transmitted. To observe whether the system will be affected by the Network background traffic, we use the parameters below:

1) Non-Skype sessions (N_S): In a complex environment, the detected sessions will be identified if the packet number per second is smaller than the threadshold value.

2) Skype sessions (S): Skype sessions created by Skype API program.

3) Suspected Skype sessions (M): When the system captures session features, sessions will be classified as Skype session.

4) Non-detected Skype session number (N): Sessions with features but not classified as Skype session.

5) Non-reported rate of session (FN): The number of Skype sessions that system takes as non-Skype session.

6) Faulty-reported rate of session (FP): The number of non-Skype sessions that system takes as Skype session.

TABLE I
Table of FN and FP in Skype environment

N_S	S	M	N	FN	FP
0	20	19	1	0.05	0
0	40	39	1	0.025	0
0	60	59	1	0.0167	0
0	80	78	2	0.025	0
0	100	97	3	0.03	0

TABLE II
Table of FN and FP in complex environment

N_S	S	M	N	FN	FP
20	20	21	1	0.0250	0.0500
40	40	41	3	0.0375	0.0500
60	60	59	5	0.0417	0.0333
80	80	79	6	0.0375	0.0313
100	100	101	7	0.0350	0.0400

The results show that in the Skype environment, the worst identification rate for FN and FP is 95% while it is 90% in the complex environment. The reason for a lower identification rate is that the internet traffic in a complex environment is unpredictable. As a result, it is difficult to estimate the number of clusters and determine the cluster centers. To obtain a better outcome, we will need to repeat the experiment several times and improve the results of cluster.

C. Network quality analysis

We understand that the Skype users' satisfaction is related to the network quality [11]. In this study, we find that the percentages of influence in jitter, packet loss and internet delay are 46%, 53% and 1% respectively. For their high percentage, we choose jitter and packet loss for discussion.

The testing environment architecture is as Fig. 2. Two computers communicated with Skype are in the same sub-network. One of them uploads through FTP in full speed. We observe the performance in bandwidth guaranteed Skype voice communications and those without bandwidth guarantee under heavy background traffic circumstances. The jitter and packet loss rate are presented in Fig. 3. and Fig. 4. We know the jitter and packet loss rate are lower when using our system which has bandwidth system assurance compared to the situation that don't have bandwidth system assurance via the results of Fig. 3. and Fig. 4. These are the two most important elements that influence voice quality and users' opinion.

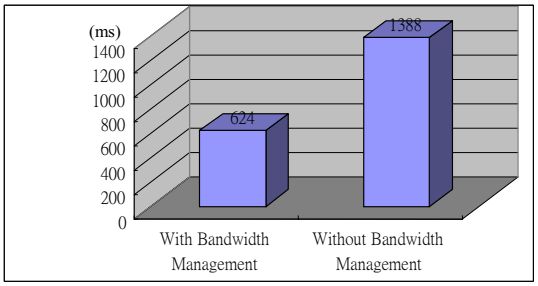

Fig. 3. Jitter comparison

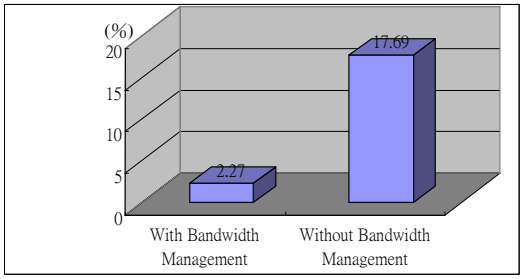

Fig. 4. packets loss comparison

D. Voice quality analysis

This research adopts Mean Opinion Scores (MOS) and E-Model index to be the basis for voice quality evaluation. In multimedia (audio, voice telephony, or video) especially when codecs are used to compress the bandwidth requirement (for example, a digitized voice connection from the standard 64 kilobit/second PCM modulation), the Mean Opinion Score (MOS) provides a numerical indication of the perceived quality of received media after compression and/or transmission. The MOS is expressed as a single number in the range 1 to 5, where 1 is lowest perceived quality, and 5 is the highest perceived quality. MOS tests for voice are specified by ITU-T recommendation P.800. The MOS is generated by averaging the results of a set of standards, subjective tests where a number of listeners rate the heard audio quality of test sentences read aloud by both male and female speakers over the communications medium being tested. A listener is required to give each sentence a rating. The mean score ranges from 1 to 5, including decimal fractions. The numbers 1 to 5 represent very poor, poor, average, good and excellent respectively.[25]

E-Model (ITU recommendation G.107) is a powerful and repeatable way of assessing the voice quality within VoIP networks. Traditional methods of measuring voice quality required an intrusive approach, where a reference signal is passed through the network and the degraded signal is received at the other end of the network. These signals are then compared using complex algorithms and a resultant Mean Opinion Score (MOS) is obtained. The E-Model's approach is non-intrusive; an intrusive reference signal is not required, nor is capture of the corresponding degraded signal. MOS measurements are possible on live traffic anywhere in the call path. The E-Model's output is referred to as the "R factor", which is a numeric value that ranges from 0 to 100. The R factor is then correlated to a MOS, based on a few network variables.[26] The higher the value of R, the better the voice quality is.

There are 130 voting samples collected from the webpage for mean opinion score analysis. And three voice categories are evaluated: original Skype voice sample, bandwidth-management voice sample and non-bandwidth-management voice sample. The results of MOS for original skype voice sample, bandwidth-assured voice sample and non-bandwidth-assured voice sample are 3.80, 3.56 and 2.88 respectively. In another word, users are in general satisfied with skype's quality of serve and only a number of users are satified with the bandwidth-assured system. Most of users, however, are dissatisfied with the non-bandwidth assured system. We can see the sample scoring distribution in Fig. 5.

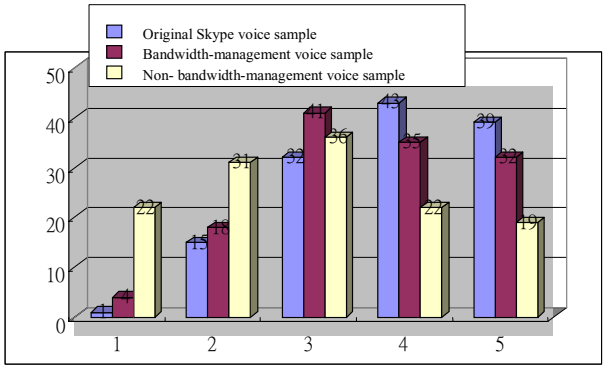

Fig. 5. Scoring chart of voice samples

We know the voice quality is better when using our system which has bandwidth system assurance compared to the situation that don't have bandwidth system assurance via the results of MOS and E-model in Table III.

TABLE III
SCORING COMPARISON OF MOS AND E-Model

Sample / Method	Original voice sample	Bandwidth-assured voice sample	Non-Bandwidth-assured voice sample
MOS	4.30	3.56	3.12
E-Model	X	3.82	2.41

V. CONCLUSION

When the sufficient training data is required for clustering analysis of voice packets, the algorithm of recognition classification used in this study can be applied to other VoIP software. The objective of this experiment is to identify Skype session in order to guarantee the bandwidth and assure the quality of service when using VoIP. Our system provides powerful and effective functionality for network processor when processing packet traffic. Although this study focuses on Skype only, we believe that the developed system can be applied on other VoIP software as well.

ACKNOWLEDGEMENT

This paper was supported by the National Science Council under contract number NSC- 95-2219-E-194-005-. In addition, the author is very grateful to the relevant researchers' supports and the anonymous reviewers for their suggestions and comments.

REFERENCES

[1] C.W. Chiu , J.Y. Pan , "The design and implementation of Internet phone dynamic bandwidth management",A Thesis Submitted to the Degree of Master in Department of Communication Engineering, National Chung Cheng University,July 2005.

[2] Jyh-Shing Roger Jang , "Data Clustering and Pattern Recognition",[online].Available: http://neural.cs.nthu.edu.tw/jang/books/dcpr/.

[3] S. L. Garfinkel, "VoIP and Skype Security", Skype Security Overview, January 2005.

[4] D. Bergström, "An analysis of Skype VoIP application for use in a corporate environment", October 2004, [online]. Available： http://www.geocities.com/bergstromdennis/.

[5] G. Prentice "Hello, we're Skype – we really like big new ideas", Skype Hardware Development & Business Opportunity Seminar, June 2006.

[6] HTB Home, [online]. Available: http://luxik.cdi.cz/~devik/qos/htb/.

[7] ITU-T Recommendation P.800.1, "Mean opinion score （MOS）Terminology", February 2003.

[8] ITU-T, "G.107 The E-model, a computational model for use in transmission planning", ITU-T, March 2005.

[9] J. Stephens, Iptables – How does it work? , [online]. Available： http://www.sns.ias.edu/~jns/security/iptables/iptables_conntrack.html.

[10] K. Suh, D. R. Figueriredo, J. Kurose, and D. Towsley, "Characterizing and detecting Skype relayed traffic: A case study using Skype", In Proceedings of IEEE INFOCOM 2006, Barcelona, Spain, April 2006.

[11] K. T. Chen, C. Y. Huang, P. Huang and C. L. Lei, "Quantifying Skype User Satisfaction", In Proceedings of ACM SIGCOMM 2006, Pisa, Italy, September 2006.

[12] O. Andreasson, Iptables Tutorial 1.1.19, 2003, [online]. Available： http://www.faqs.org/docs/iptables/index.html.

[13] S. A. Baset and H. G. Schulzrinne, "An Analysis of the Skype Peer-to-Peer Internet Telephony Protocol", In Proceedings of IEEE INFOCOM 2006, Barcelona, Spain, April 2006.

[14] Skype, [online]. Available：http://www.skype.com/.

[15] Y. Gong, "Identifying P2P users using traffic analysis", July 2005, [online]. Available：http://www.securityfocus.com/infocus/1843/.

[16] L Deri, "Open Source VoIP Traffic Monitoring", SANE 2006, May 2006.

[17] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley and E. Schooler, "SIP: Session Initiation Protocol", IETF RFC 3261, June 2002.

[18] F. Andreasen and B. Foster, "Media Gateway Control Protocol （MGCP）Version 1.0", IETF RFC 3435, January 2003.

[19] ITU-T Recommendation H.323, "Packet-based multimedia communications systems", 2003.

[20] F. Cuervo, N. Greene, A. Rayhan, C. Huitema, B. Rosen and J. Segers, "Megaco Protocol Version 1.0", IETF RFC 3015, November 2000.

[21] http://www.winpcap.org/

[22] http://www.elet.polimi.it/upload/matteucc/Clustering/tutorial_html/kmeans.html

[23] http://en.wikipedia.org/wiki/NAT

[24] http://en.wikipedia.org/wiki/Data_clustering

[25] http://en.wikipedia.org/wiki/Mean_Opinion_Score

[26] http://www.gl.com/packetscanletter.html

[27] http://en.wikipedia.org/wiki/Data_clustering#K-means_clustering

[28] http://en.wikipedia.org/wiki/Iproute2

[29] http://en.wikipedia.org/wiki/Dynamic_bandwidth_allocation