Cause Analysis of Packet Loss in Underutilized Enterprise Network Links

Cause Analysis of Packet Loss in Underutilized Enterprise Network Links

by
Deepali Agrawal
Division of Electrical and Computer Engineering
(Computer Science and Engineering)
POSTECH

A thesis submitted to the faculty of POSTECH in partial fulfillment of the requirements for the degree of Master of Science in the Division of Electrical and Computer Engineering (Computer Science and Engineering)

Pohang, Korea
December 21, 2005
Approved by

_____

Major Advisor: James Won-ki Hong

Cause Analysis of Packet Loss in Underutilized Enterprise Network Links

디팔리 아그라왈

위 논문은 포항공과대학교 전자컴퓨터공학부 (컴퓨터공학) 석사 학위논문으로 학위논문 심사위원회를 통과하였음을 인정함.

2005 년 12 월 21 일

학위논문심사 위원회 위원장 홍원기 (인)

위 원 서영주 (인)

위 원 송황준 (인)

MECE 20042290 Deepali Agrawal, Cause Analysis of Packet Loss in Underutilized Enterprise Network Links, Division of Electrical and Computer Engineering (Computer Science and Engineering), 2006, 76P, Advisor: J. Won-Ki Hong, Text in English.

Abstract

ISPs and enterprises are equipping their networks with sufficiently large bandwidths to untangle the performance related problems such as packet loss, delay and jitter. Considering that these problems can exert a detrimental effect due to hampering of QoS of loss sensitive applications such as VoIP, streaming multimedia, videoconferencing and teleconferencing. However, overprovisioning does not have satisfying result on the loss sensitive applications. This situation raises a need to determine what is responsible for packet loss in such an environment. This study aims at analyzing the traffic characteristics and suspected causes of packet loss to reach the root cause in the underutilized enterprise networks. We collected packet loss and traffic information from a dormitory backbone switch deployed on POSTECH campus network. We define a clear and complete framework to approach the solution in a systematic, impeccably logical and goal-directed way. We generated independent and efficient tools to execute our methodology. Then by applying these tools on the collected data, we analyzed packet loss with respect to various traffic metrics at different time granularities. Analysis reveals that losses are strongly correlated with each other. Broadcast packets and non-IP packets do not affect losses. We also studied

various characteristics of the TCP flows like their counts, lifetimes, sizes, data rates and etc. We observed evidences showing correlation between the packet loss phenomenon and large TCP flows. However, it is crucial to draw factual conclusion about their relationship due to the SNMP constraints and hence, this work needs to be continued further. Nevertheless, the methodology and analysis tools that we have developed are flawless and are of their own value and they can be independently used for efficient and precise data analysis.

Table of Contents

Table of Figures

List of Tables

1 Introduction

Today, the core networks are evolving fast to fulfill the stringent time (e.g., QoS guarantees) and high performance network requirements whereas the edge routers/switches draw less attention and remain relatively low performance. In fact, more under-utilized links will appear near the edges of enterprise networks. Recently, it has come to notice that packet losses and delays are observed on such under-utilized links also. A study of packet loss or delay on such under-utilized links has lately got some attention.

ISPs started employing overprovisioning to mitigate these performance related problems such as packet loss, delay and jitter. Because, these losses can endeavor an adverse effect on QoS of loss sensitive applications like VoIP, streaming multimedia traffic, teleconferencing and peer-to-peer. However, overprovisioning does not cause adequate improvement in the performance of these loss sensitive network applications. These circumstances demand to determine what is responsible for packet losses in such an environment. In this study, we aim to analyze traffic characteristics and suspected reasons of packet loss to reach the root cause.

We collected packet loss and traffic information from the dormitory backbone switch deployed on POSTECH's campus network. To obtain the packet loss information, we fetched data from Cisco enterprise MIB variables supported by the monitored switch. Further, to obtain traffic information we implemented optical TAP on the monitored link and captured packet traces using DAG card that guarantees lossless performance in gigabit link.

We developed impeccably logical, well planned and complete methodology to approach the complex problem that we are dealing with in this work. We generated independent and competent tools that can efficiently execute the methodology. By

applying these tools on the data, obtained from our monitoring system, we analyze packet loss with respect to various traffic parameters like broadcast packets, various IP and non-IP protocols, TCP flows and their properties like flow count, lifetime, sizes, data rate, loss rate and etc. at various time granularities.

In this work, we determine that the losses are highly correlated with each other. Further, broadcast packets and non-IP packets are not affecting losses. We caught some clues showing that the packet loss events might be related to heavy tailed TCP flows that are mostly caused due to large file transfers. However, because of SNMP constraints it is difficult to reach factual conclusion. Nevertheless, our methodology and tools are impeccable and efficient.

The organization of this paper is as follows. Related work is presented in Section 2 and our packet loss and traffic monitor system is described in Section 3. Section 4 describes the traffic data collection and experimental environment. Our analysis tools are described in section 5. In Section 6, we give an analysis of packet loss and traffic. Finally, concluding remarks are given and possible future work is discussed in Section 7.

2 Related Work

Not much research work is done in cause analysis of packet loss area and the microcongestion study. Generally analysis of packet loss and microcongestion are just one of the by-products, albeit important one. The following summarizes some research work related to our study. Most of these studies focus on the packet delay issue rather than packet loss. Since packet loss is closely related to packet delay and network congestion, it is worthwhile to adapt and verify some of their claims along with our own analysis categories.

Papagiannaki et al. [1] examined causes of microcongestion episodes in an access router leading away from the core. They identified and discussed three root causes of congestion: 1) reduction in link bandwidth from core to access; 2) multiplexing across different input links; and 3) degree and nature of burstiness of input traffic streams. However, this study emphasizes only on delays, and packet loss analysis is not considered. Secondly, the links they scrutinized roughly carry 50% of the traffic each. However, we deal with highly underutilized links having approximately below 20% utilization. Thirdly, their analysis employs a generic queuing model that may not accurately reflect the architecture and behavior of modern switches.

Papagiannaki et al. [2] presented a characterization of congestion in the Sprint IP backbone network. They analyzed link utilization at various time scales (millisecond level) to measure the frequency and duration of microcongestion. While they detected traffic bursts, they did not mention the packet loss that occurred during these times. Their study did not provide various traffic parameters except burst, and no information of the packet loss. Hence, further work is needed on this topic. In our work, we will provide the packet loss characteristics with various traffic

parameters related to protocol, spatial characteristics and flows.

Hohn et al. [3] provided insights into system busy periods and showed how queues build up inside a router. They have monitored all input and output links of single IP router and reported actual magnitudes and temporal structure of congestion episodes. Although, their simple triangular shaped model can capture useful delay information, they have not related their measurements to the packet losses in the links.

Mochalski et al. [4] studied changes in traffic pattern relative to different points of observation using TAP in the network and investigated the contributing factors to the changes observed. They measured the delay across the router and firewall. They attributed high delays in the links to the congestion in the router and tried to relate the delay to packet loss, concentrating on analyzing packet loss using delay. We will try to analyze packet loss in finer time granularity using a range of parameters.

Chung et al. [5] has done an early work on detecting and analyzing packet loss in underutilized enterprise networks. They fetched data from private and standard SNMP MIB variables of the monitored routers and switches and analyzed data across three time scales: 5min, 5sec and 1sec. Analysis showed that traffic is bursty at small time scales like 1 second and 5 seconds. They proved that packet loss exists on underutilized links and losses are caused due to the traffic burst that occurs at time granularity smaller than 5 seconds. This analysis is done at course time granularity. Hence, further work needs to be done to determine the traffic characteristics in finer time granularity.

Chung et al. [6] has further extended his previous work [5]. He developed a passive traffic monitoring system, which can capture all packets going through a network link and analyzed the data that are representative of packet loss across

various time scales: 10 milliseconds, one second, 10 seconds and one minute. He analyzed packet loss with various traffic parameters like number of packets, packet size distribution and flows, etc., and indicated that only bursty packets affect the packet loss. Though the methodology used was very good, his conclusion is simple and so well known that he failed to bring in new findings. Hence, further work needs to be done in this field.

3 Traffic Monitoring and Loss detection

We monitored the traffic using two different modules; SNMP polling module and packet capture module. Figure 1 illustrates the overview of traffic monitoring modules. Our Linux monitoring system has two network cards; SNMP polling module uses the 100Mbits NIC card and packet capture module uses 1Gbits DAG card. Table 1 describes the function two modules. Next, subsections depict the working of these modules in detail.

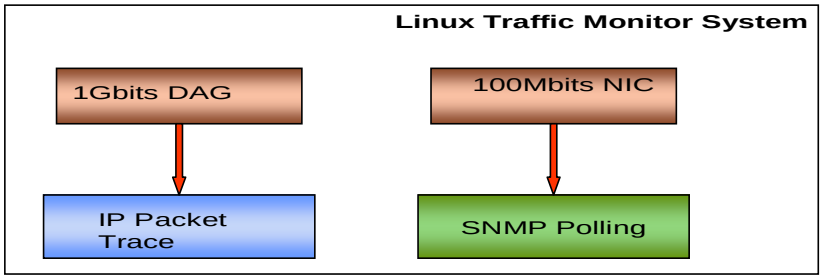


Figure 1. Overview of Traffic Monitoring Modules

Module Name	Description
SNMP Polling Module	This module polls Cisco standard MIB II and private MIB variables
Packet Capture Module	This module captures packet trace using DAG API

Table 1. Implemented Module List in Traffic Monitor System

3.1 SNMP Polling Module

Traditional way of monitoring the system is using SNMP [8]. In this module a SNMP agent is used to obtain traffic measures from SNMP enabled switches and routers. From the supported SNMP MIB II variables we selected and fetched data for four MIB variables; ifInUcastPkts, ifOutUcastPkts, ifInOctets, and ifOutOctets. The selected SNMP MIB II variables are described in Table 2.

Object	OID	Description
ifInUcastPkts	1.3.6.1.2.1.2.2.1.11	The number of subnetwork-unicast packet delivered to a higher-layer protocol
ifOutUcastPkts	1.3.6.1.2.1.2.2.1.17	The total number of packets that higher-level protocols requested be transmitted to a subnetwork-unicast address,including those that were discarded or not sent
ifInOctets	1.3.6.1.2.1.2.2.1.10	The total number of octets receive on the interface, including framing characters
ifOutOctets	1.3.6.1.2.1.2.2.1.16	The total number of octets transmitted out of the interface, including framing characters

Table 2. SNMP Standard MIB II Variables

In SNMP polling module these variables are used to get values for ingress/egress packet count and ingress/egress link utilization for each interface of the monitored switch, which helps understanding the overall switch status and traffic activities on each port. These variables are polled at the time granularity of one second.

3.1.1. SNMP Constraints

Table 3 exposes the sample of data file obtained by SNMP polling module. First column shows the time when the SNMP value is received and second column shows the ingress bits values for interface 2, polled at the granularity of 1 second. The

fourth row shows that 1.91 Giga bits are received at 7:40:56pm, which is practically not possible as the physical limit (bandwidth) of the link is 1Gbps. Moreover, during 7:40:50 to 7:40:56 no values are received from the switch.

Further, similar characteristics are observed for other interfaces also and for data files of other MIB variables as well. This data indicates that SNMP values are inaccurate at the granularity of 1 second which resulted from time lags in the devices' counter update interval.

Time	Ingress Bits
7:40:47 pm	1.30415E8
7:40:48 pm	1.25195E8
7:40:50 pm	1.27854E8
7:40:56 pm	1.91079E9
7:40:57 pm	1.23278E8
7:40:58 pm	1.34517E8

Table 3. SNMP Data from Dormitory Backbone Switch

Next, Table 4 illustrates the SNMP response loss rate in terms of percentage for various parameters that are we are polling. These rates are calculated over 30min interval near the end of the day when the maximum traffic activity and losses are observed across the switch and the switch is very busy. From Table we can see that the rate at which we do not receive back any response from switch is very high though the response loss rate is low (like 22%) during day time when the switch is not that busy.

Parameter	Loss rate (%)
Ingress byte	40
Egress byte	37
Ingress packet	37
Egress packet	38
Ingress/egress drop	58

Table 4. SNMP Response Loss Rate

A router/switch's main priority is to forward the packets, which can cause switch to ignore the SNMP Get request that we send every second or delay the MIB counter updates, especially when traffic load is heavy. 10 second average values [5] of the MIB variables are considered as accurate enough for study. However, there are many blank spots in the SNMP measurements when we do not know what is actually happening in the switch and they are distributed across the time, which makes it difficult to use these measurements for sophisticated analysis.

3.2 Packet Capture Module

One second time granularity is not satisfactory enough to study the traffic characteristics in detail. To detect the causes of packet loss we need to look into the various traffic parameters in finer time granularity of milliseconds and microseconds. Because the suspected causes of packet loss such as microcongestion and traffic burstiness in smaller time can not be studied with the SNMP data polled every one seconds. Traffic monitoring using tap can solve these time interval problem. Using tap, traffic on the link can be captured in real time.

In our study we used optical tap to monitor traffic and implemented a packet capture module. We used an Endace's DAG 4.3GE card [9] to collect packet trace, which guarantees lossless capture performance in a gigabit Ethernet link. DAG

stores each packet in a typical record format called Extensible Record Format (ERF) that consist of fixed length header and non zero bytes of the packet captured. DAG's ERF trace allows us to obtain the high precision time-stamp (up to microsecond) of the packet.

Packet capture module uses C API of DAG [10] to capture packets, which provides the highest performance by using a zero-copy memory-mapped interface to the DAG. We capture 80 bytes of each packet which includes Ethernet header (14 byte), IP header (20 bytes), TCP/UDP header (20 bytes) and some part of payload.

3.3 Packet Loss Detection

Packet loss could be obtained by comparing the incoming and outgoing packet counters of standards MIB. However, it would not be accurate due to the difference in incoming and outgoing packet counts caused due to packets that are destined to the router, generated by the router and broadcast by the router. This problrm can not be avoided using standard MIB variables. Hence, to detect the packet loss we used Cisco enterprise MIB variables [11] that provide packet loss information; locIfInputQueueDrops, locIfOutputQueueDrops. We also poll enterprise MIB variable: cpuLoad that provides information about the CPU load. The selected enterprise MIB variables are described in Table 5.

Object	OID	Description
cpuLoad	1.3.6.1.4.1.9.2.1.56	CPU Utilization (5 sec avg.)
locIfInputQueueDrops	1.3.6.1.4.1.9.2.2.1.1.26	The number of packets dropped because the input queue was full
locIfOutputQueueDrops	1.3.6.1.4.1.9.2.2.1.1.27	The number of packets dropped because the output queue was full

Table 5. Cisco Enterprise MIB variables

Cisco document [12] specifies that: Each interface in the switch/router owns an input queue onto which incoming packets are placed to await processing by the Routing Processor (RP). Frequently, the rate of incoming packets placed on the input queue exceeds the rate at which the RP can process the packets.

The information of dropped packet number, because the interface input queue was full, is updated in locIfInputQueueDrops MIB variable in accumulated order, and the packet number, because the interface output queue was full, is updated in locIfOutputQueueDrops MIB variable in accumulated order. We calculate packet loss by following formula,

Packet Loss = locIfInputQueueDrops + locIfOutputQueueDrops

3.4 Packet Loss & Traffic Monitoring Method

In this work, we implemented a monitoring system that can collect traffic accurately in a small time granularity (upto microseconds). Modules in the monitoring system are shown in Table 1. We monitored traffic on the link using TAP and SNMP standard MIBs. At the same time we monitored packet losses by polling Cisco private MIBs using SNMP. Packet loss detection and cause analysis methods on underutilized links are described below.

- If there is a packet loss, we can obtain the magnitude and time the packet loss occurred in accuracy of 1 second.
- At the same time the traffic on the link is monitored by our implemented system.
- We analyze data sets obtained by the step above in offline to detect the causes of the packet loss in the underutilized links.

4 Root Cause Analysis Method

In this project we are dealing with a complex problem of determining the root cause of packet loss. We have defined a complete and clear framework that enables us to approach this problem in impeccably logical, systematic, and goal-directed way. Figure 2 illustrates this framework or methodology.

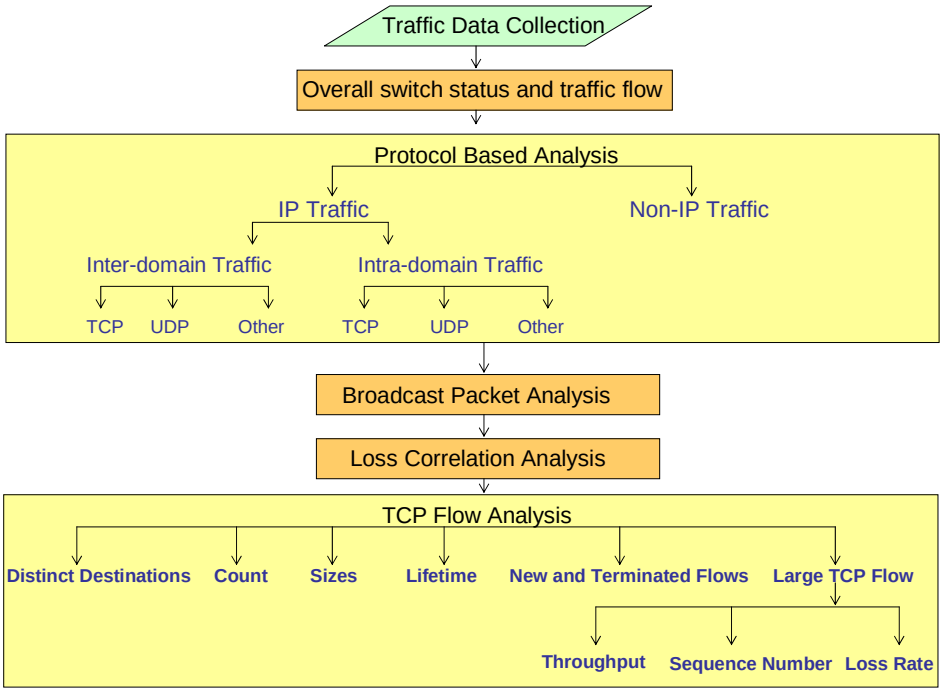


Figure 2. Overview of Cause Analysis Methodology

We first collect two data sets (using both SNMP and packet capture module) each a day long, one that has losses and the other with no losses. The data with losses is studied with respect to identifying why they may have occurred, and the data without losses is studied to understanding why losses have not occurred.

For each data set, we first calculate a precise, comprehensive but succinct birds-eye-view of the important variables of the traffic logs. We generate a Table that summarizes the average, standard deviation, max data rates, average and max utilization in both ingress/egress direction on each port based on the SNMP measurements. For packet loss we calculate total count per port in ingress/egress direction, corresponding packet loss rates (packets dropped / total packets per ingress/egress), and max. These data comprise a coarse, static weather map of the switch. Just by looking at these summaries for loss and no loss period, one can readily see if the rough characteristics have stayed invariant or changed. Secondly, we check the flow direction of the traffic. Does the traffic entering the switch from non-backbone ingress link flows to other non-backbone egress link or it goes to backbone link.

Next, using packet trace data we separate the traffic into IP and non-IP traffic. Non-IP traffic is used to check if any spurious or malicious packets are present in the link. This analysis also serves as sanity check for data. If too many spurious (non-IP) packets are present there could be some problem with the capture system or the traffic. Hence, we also check what type of non-IP packet is present in the link. Further, for IP packets we check (after exiting from the monitored switch) where do they go, by separating inter/intra domain traffic means the traffic that goes outside the POSTECH campus and the packets that are routed inside the campus. Then to understand the traffic composition that we are dealing with and too learn what is percentage of each protocol, is any particular protocol present in high percentage, does it have any correlation with the loss, we refine the traffic into TCP, UDP and other.

Next, we check the percentage of IP level broadcast packets and their distribution to see if they bear any relationship with loss event. Then we check if any correlation exists between the losses themselves and does congestion occur in our switch, if the losses are experienced by many interfaces simultaneously then we can say that the switch is congested.

Then for TCP only, we check how many inter domain TCP sessions (distinct 4 tuples) are present and how many intra domains. We concentrate only on TCP because it is present in highest percentage in our traffic and TCP flows are primary suspects of losses due to microcongestion on account of their linear increase/exponentially decrease congestion control mechanism.

Next, we analyze lifetime and sizes of TCP flows. Since TCP induced losses require large file transfer sessions, we need to know what the TCP workload composition structurally looks like. Further, we check how many flows are initiated and terminated every second. To confirm that most of the flows are short flows and almost equal number of flows are initiated and terminated every second.

Lastly, we select large TCP flows as they are the best candidates for causing microcongestion, and study their packet-level and byte-level time series. We expect to see the characteristic sawtooth pattern and compare the two during loss and no loss period. Then, we analyzed sequence numbers per flow and their loss rates to check their correlation with losses.

4.1 Traffic Data Collection

SNMP agents are running on various network devices deployed in the campus network. POSTECH's campus network is comprised of a gigabit Ethernet backbone, which, in turn, is composed of two Cisco IP routers, two core backbone switches,

dozens of gigabit building backbone switches, and hundreds of 100Mbps switches and hubs that are deployed inside the buildings, as shown in Figure 3. For our study we were interested in the link that is continuously underutilized and convey the traffic that is composed of many internet applications. Our campus internet access links are not underutilized and we monitored various links across various switches/routers to find the most suitable place to observe packet losses on underutilized link. We found the link that conciliated above conditions, located on campus dormitory network. This link is the edge of 1 Gbps links in POSTECH's campus network and is continuously underutilized. Many students are using this link and connecting to internet through it because this link is established in the dormitory network. The traffic generated by the students is representative of traffic that ISPs handle from their users.

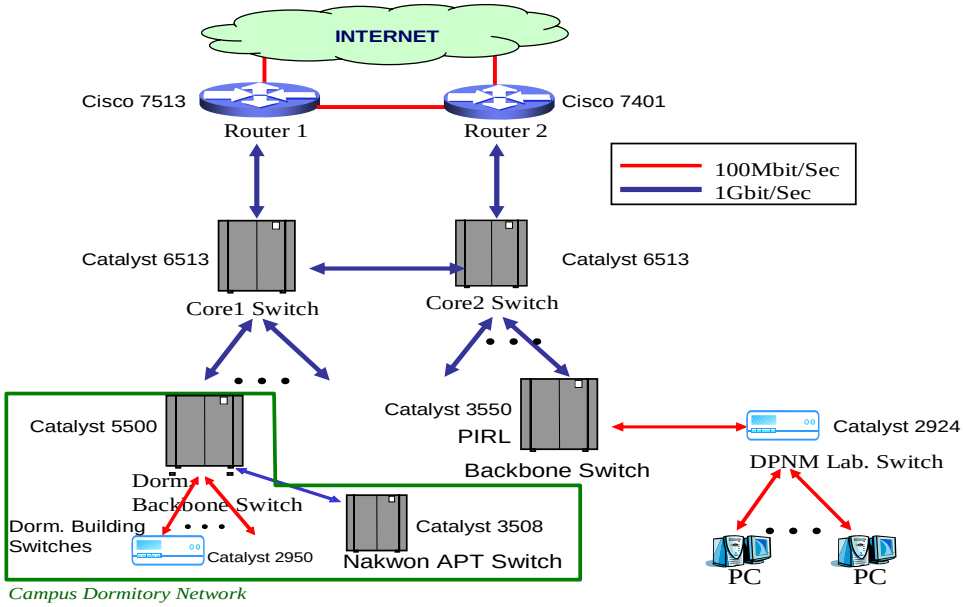


Figure 3. POSTECH's Campus Network Overview

We selected dormitory backbone switch (Catalyst 5500), to monitor traffic and packet loss, that is placed next to the Core switches (Catalyst 6513) that are placed next to internet access routers (Cisco 7513 and 7401). This dormitory backbone switch is connected to many sub-dormitory switches that are connected to different dormitories and Nakwon apartments as shown in Figure 4. We monitored all the links between the dormitory backbone switch and the sub-dormitory switches to find the most suitable link to install the TAP for our study. We found that the link between dormitory backbone switch (Catalyst 5500) and sub-dormitory switch (Catalyst 3508) which is connected to "Nakwon APT" can satisfy our requirements. This link was a good choice because it was underutilized and it showed the occurrence of steady packet losses.

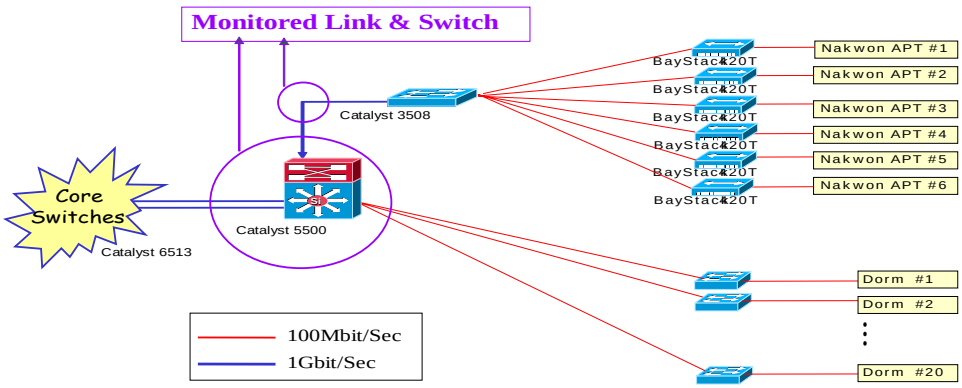


Figure 4. POSTECH intranet infrastructure for Dormitories

Figure 5 demonstrates our experimental setup. We installed optical tap on the link that connects dormitory backbone switch (Catalyst 5500) to the sub-dormitory switch (Catalyst 3508). The dormitory backbone switch supports Cisco private MIBs, so we could obtain packet loss information using SNMP polling module which was implemented in the traffic monitoring system.

We monitored one-direction (ingress) traffic that flows from the sub-dormitory switch to the dormitory switch. Because almost all the time the output queue drops (locIfOutputQueueDrops) values were zero and packet loss was mainly caused due to packet drop in input queue of the interface. However, near the end of the semester we obtained some data that showed packet drop in the output queue of the interface. Hence, in future both the uplink and the downlink will be needed to monitor for packet loss study.

We monitored only one link of the dormitory backbone switch using tap. Because monitoring one link is enough to study the causes of packet loss when packet loss occurs on that particular port. Traffic status on the other interfaces of the switch is obtained using SNMP polling.

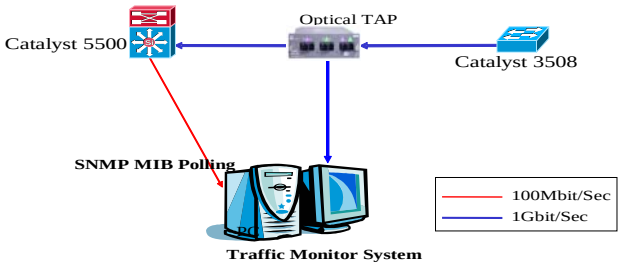


Figure 5. Experimental Environment Overview

Dormitory Backbone switch (Catalyst 5500) that we are monitoring has 11 interfaces and all of them are active. Interface 11 is the one that is connected to the Nakwon apartment sub-dormitory switch and a tap is installed on this link. Interface two is connected to the core1 switch (Catalyst 6513.) Both of these interfaces are Giga ports. However, rests of the interfaces are fast Ethernet (100Mbits) ports and are connected to dormitories. Virtual LANs (VLAN) are installed across these 100Mbit ports combining two ports into one VLAN as disclosed by Figure 6. Hence,

total bandwidth of these interfaces is 200Mbits as two separate physical links are connected into each VLAN setup.

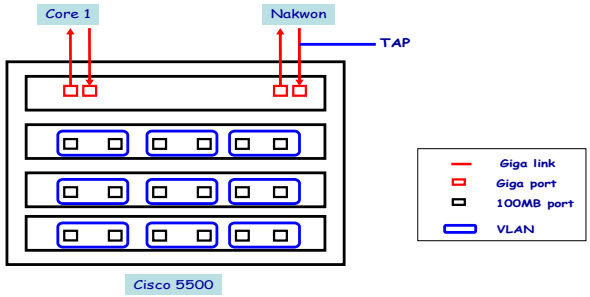


Figure 6. VLAN Setup for Monitored Switch

4.2 Analysis Tools

We have developed competent analysis tools to execute the methodology that we have defined to approach the problem. These tools can efficiently analyze our SNMP and DAG logs that are captured in a raw format.

From SNMP polling module we obtain SNMP data files for each MIB variable that we are polling for all the interfaces of the switch. Table 6 below list some tools developed using perl [13] to process the SNMP data to reveal information in useful format. Appendix A describes them in further detail.

Tool Name (Perl Script)	Input	Output	Function
Interface Count	SNMP Log file	Text file	Count of interfaces those experience losses simultaneously
Interface Index	SNMP Log file	Text file	Index of interfaces those experience losses simultaneously
Response Loss Rate	SNMP Log file	Text file	Time in Unix seconds when the SNMP response is not received
Average	SNMP Log file	Text file	Averaged values for specified seconds
Total	SNMP Log file	Text file	Total count across all ports

Table 6. SNMP Logs Processing Tools Description

Table 7 describes tools that are developed to process the DAG logs (packet trace files) obtained from packet capture module. These tools provide measurements for various traffic parameters like number of broadcast packets, non-IP packets, etc. Further, they process the data to generate flows and analyze their properties like flow counts, sizes, lifetimes, data rates, and etc. Appendix B describes them in further detail.

Tool Name (C programs)	Input	Function
Utilization measure	DAG Logs	Inter domain and intra domain bit counts (at 1s, 1ms and 1us scale)
Protocol Analysis	DAG Logs	Inter & intra domain Bit Counts : TCP, UDP and Other (1s, 1ms & 1us)
Flow generator	Binary and ascii flow file	Generate 4 (src/dst IP, src/dst port) tuple based TCP flows
Flow count	Binary Flow file	Inter/intra domain flow counts per second
Flow lifetime and size	Binary Flow file	Inter/intra domain flow lifetime and sizes
New and Exit Flow	Binary Flow file	New and terminated flow counts and sum of their sizes per second
Distinct destination	Binary Flow file	Count of distinct destinations to which each source IP connects
Top_n_flow	Binary Flow file	flows of size in the specified range
Data Rate	Binary Flow file & DAG logs	pps and BPS of the selected flow (at 1s, 1ms and 1us scales)
Run length and loss rate	Binary Flow file & DAG logs	run-length magnitudes and interval between two run-lengths in microsecond
Broadcast packets	DAG logs	Count of IP level broadcast packets (at 1s, 1ms and 1us scales)
Non-IP packets	DAG logs	Count of non-IP packets and ARP packets (at 1s, 1ms and 1us scales)

Table 7. DAG Log Processing Tools Description

Diagnosis Tool

This tool combines various tools listed above (1-4 from SNMP tools and 1-9 from DAG tools) to simultaneously generate data files for all types of different analysis categories by taking SNMP and DAG logs as input. Then using GNUPLOT [14] scripts we can generate gnuplot executable file (gnuplot.config) for any number of input files for a specified time period. After generating plots using gunplot, we use latex [15] script to arrange all the plots, which are present in a specified folder,

in one latex document. By following this procedure we can generate a complete document that has plots arranged according to various categories that we have specifies in our logical framework. This document helps to study all basic plots in systematic way to have understanding of the system and traffic that we are monitoring. Figure 7 shows overview of diagnosis tool.

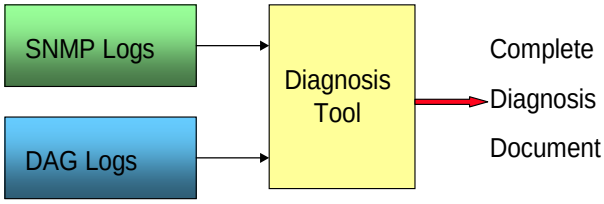


Figure 7. Diagnosis Tool Overview

5 Cause Analysis of Packet Loss

This section exhibits the cause analysis of packet loss based on various traffic metrics in various time scales. We collect packet traces passively using TAP and poll MIB variable at 1 second granularity and then aggregate to obtain 5 second and 10 second average values. We have collected data over 4 months. Data collection and analysis were interleaved together; after collecting 2~3 days' data (packet traces and SNMP data), we analyze it; then again collect new data set, analyze it further and so on. These results are obtained by applying our analysis tools on the collected data.

5.1 Switch Entire Perspective

In this section we study the overall switch status by looking at the traffic statistics across all the interfaces of the switch.

5.1.1. Underutilized Links

Tables 8 to 13 demonstrate the statistics of Ingress, Egress Bits and the Packet loss when losses exist on Nakwon link and losses do not exist on Nakwon link. Unit of all the quantities in the Tables 8, 9, 11 and 12 is kept as Mbps. These tables help us understand the basic characteristics of the traffic and monitored switch status at the coarse level, in the presence and absence of loss.

Average utilization column shows that all the links are well underutilized. Specially, the Nakwon link which is 1.7% utilized even when packet loss is detected. However, the ratio of maximum utilization to the bandwidth is very high at 1 second time granularity. For few links the ratio is even higher than 1 due to the SNMP granularity and inaccuracy issues that are discussed in section 3. Hence, we

calculated the maximum/bandwidth ratio at 5 and 10 second granularity. We can observe that ratio for all the interfaces go below 1 at the granularity of 10 seconds.

Traffic statistics during loss and no loss period are not overly different. However, largely traffic rate and utilization across all interfaces of switch is higher during loss period compared to no loss period. For interface 2 (backbone link), ratio of ingress traffic to egress traffic (1.6) is also higher for loss period than the no loss period (0.9). Table 8 and 10 depict that the traffic during loss period has heavy skew towards ingress that leads to ingress packet drops. Table 13 illustrates that no packet loss exists on Nakwon link but other links do experience loss during this time. However, these packet drops are significant on egress side with very few ingress packet drops which is opposite to what Table 10 illustrates.

Interface Number	Mean	Standard deviation	Max			Utilization (AVG/BW)	Max/BW		
1	0	0	0 (1s)	0 (5s)	0 (10s)	0	0 (1s)	0 (5s)	0 (10s)
2 (backbone)	265	399.4	2836	1240.34	829.61	0.265	2.836	1.24	0.829
3	25.85	38.2	346	155.94	87.10	0.129	1.73	0.779	0.435
4	12.49	18.3	163	54.59	33.24	0.062	0.815	0.272	0.166
5	65.86	94.2	797	268.98	196.07	0.329	3.985	1.344	0.980
6	30.60	44.5	438	160.79	96.46	0.153	2.19	0.803	0.482
7	54.07	10.1	213	48.15	37.77	0.270	1.065	0.240	0.188
8	18.84	29.9	381	92.27	58.31	0.094	1.905	0.461	0.291
9	4.13	13.0	198	68.03	48.51	0.020	0.99	0.340	0.242
10	11.69	16.7	150	43.91	28.58	0.058	0.75	0.219	0.142
11 (Nakwon)	17.44	27.4	279	102.51	61.82	0.017	0.279	0.102	0.061

Table 8. Ingress Bits Statistics with Loss

Interface Number	Mean	Standard deviation	Max			Utilization (AVG/BW)	Max/BW		
1	0	0	0 (1s)	0 (5s)	0 (10s)	0	0 (1s)	0 (5s)	0 (10s)
2 (backbone)	160.7	236.8	2513	718.90	561.47	0.16	2.513	0.718	0.561
3	38.36	65.2	778	260.20	230.89	0.191	3.89	1.301	1.154
4	30.13	50.6	608	202.91	175.72	0.150	3.04	1.014	0.878
5	41.99	69.2	979	265.04	173.77	0.209	4.895	1.325	0.868
6	51.80	77.4	927	392.27	210.93	0.259	4.635	1.961	1.054
7	22.45	41.0	605	254.19	154.79	0.112	3.025	1.270	0.773
8	31.03	52.1	544	197.58	179.45	0.155	2.72	0.987	0.897
9	9.33	18.4	357	81.19	58.45	0.046	1.785	0.405	0.292
10	35.23	62.0	1150	430.45	274.35	0.176	5.75	2.152	1.371
11 (Nakwon)	24.08	46.4	699	206.85	127.41	0.024	0.699	0.206	0.127

Table 9. Egress Bits Statistics with Loss

Interface Number	Total Ingress Drop	Ingress Max	Ingress Rate	Total Egress Drop	Egress Max	Egress Rate
1	0	0	0	0	0	0
2 (backbone)	0	0	0	0	0	0
3	0	0	0	0	0	0
4	134	34	8.81E-06	0	0	0
5	10	10	2.77E-07	0	0	0
6	44	44	1.87E-06	0	0	0
7	0	0	0	0	0	0
8	14958	1691	8.50E-04	0	0	0
9	78	54	1.76E-05	0	0	0
10	185	66	1.24E-05	0	0	0
11 (Nakwon)	464	135	8.81E-06	0	0	0

Table 10. Packet Loss Statistics

Interface Number	Mean	Standard deviation	Max			Utilization (AVG/BW)	Max/BW		
1	0	0	0 (1s)	0 (5s)	0 (10s)	0	0 (1s)	0 (5s)	0 (10s)
2 (backbone)	146.99	174.49	2753	672.84	475.39	0.146	2.753	0.67	0.47
3	15.79	18.79	296	71.85	43.38	0.078	1.48	0.359	0.216
4	10.05	12.75	187	47.78	33.39	0.050	0.935	0.238	0.166
5	45.79	54.61	743	194.92	154.75	0.228	3.715	0.974	0.773
6	27.19	32.58	405	113.48	84.37	0.135	2.025	0.567	0.421
7	4.67	6.34	108	34.45	19.83	0.023	0.54	0.172	0.099
8	14.61	18.00	279	67.73	55.01	0.073	1.395	0.338	0.275
9	22.30	29.13	414	109.38	81.27	0.111	2.07	0.546	0.406
10	6.82	8.67	169	40.30	25.02	0.034	0.845	0.201	0.125
11 (Nakwon)	18.33	23.20	245	90.33	73.34	0.018	0.245	0.09	0.73

Table 11. Ingress Bits Statistics without Loss

Interface Number	Mean	Standard deviation	Max			Utilization (AVG/BW)	Max/BW		
1	0	0	0 (1s)	0 (5s)	0 (10s)	0	0 (1s)	0 (5s)	0 (10s)
2 (backbone)	147.05	167.37	3178.56	752.90	451.82	0.147	3.178	0.75	0.45
3	31.27	48.88	1269.15	307.79	207.68	0.156	6.345	1.538	1.038
4	18.63	26.34	464.611	130.78	124.92	0.093	2.323	0.653	0.624
5	21.36	34.08	533.176	185.62	112.87	0.106	2.665	0.928	0.564
6	30.07	40.89	605.363	224.21	136.36	0.150	3.026	1.121	0.681
7	17.83	27.02	484.82	121.16	90.36	0.089	2.424	0.605	0.451
8	20.68	28.55	752.159	175.14	104.58	0.103	3.760	0.875	0.522
9	3.79	6.44	180.046	51.02	37.45	0.018	0.900	0.255	0.187
10	11.49	14.92	223.923	55.65	39.17	0.057	1.119	0.278	0.195
11 (Nakwon)	8.98	12.42	273.487	67.59	42.30	0.008	0.273	0.067	0.042

Table 12. Egress Bits Statistics without Loss

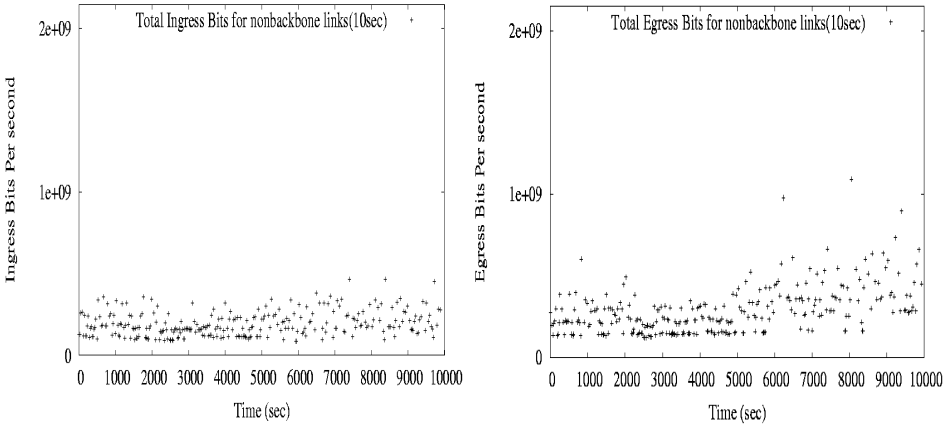
Interface Number	Total Ingress Drop	Ingress Max	Ingress Rate	Total Egress Drop	Egress Max	Egress Rate
1	0	0	0	0	0	0
2 (backbone)	0	0	0	16042	347	11.9E-05
3	0	0	0	0	0	0
4	38	20	2.88E-06	1703	56	11.6E-05
5	2	2	6.20E-08	0	0	0
6	13	13	5.35E-07	2470	44	9.58E-05
7	0	0	0	0	0	0
8	0	0	0	1694	44	10.1E-05
9	228	142	1.45E-05	0	0	0
10	0	0	0	914	33	10.4E-05
11 (Nakwon)	0	0	0	0	0	0

Table 13. Packet Loss Statistics

5.1.2. Non-backbone Vs Backbone Link Traffic

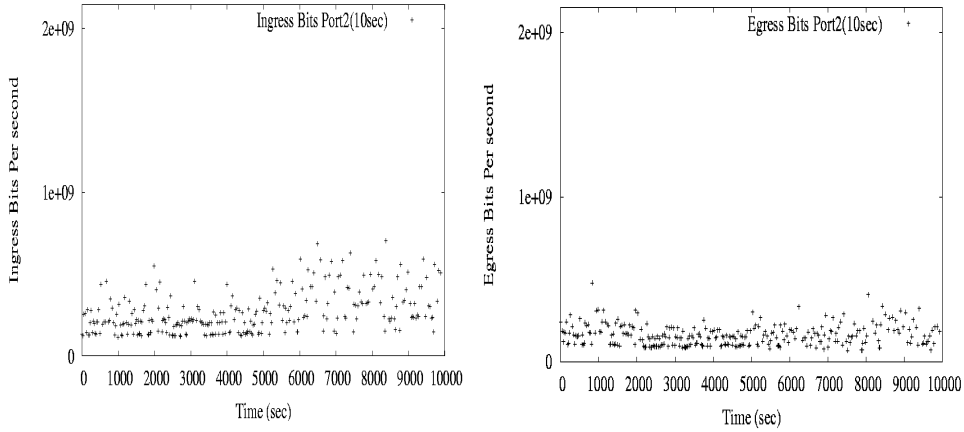
Figures 9 and 11 illustrate the distribution of ingress/egress bits across interface 2 (backbone link) which is connected to core 1 switch during the time when loss is detected on Nakwon link and when no loss is detected. Whereas, Figures 8 and 10 imitate sum of traffic across all non-backbone links during loss and no loss time respectively. Since, only one backbone link is present in the switch, we expect the sum of all non-backbone link traffics to be fairly identical with the backbone link traffic in both ingress and egress direction. Figures below roughly manifest our expectation. Further, it discloses that not much traffic goes from one ingress port to another non-backbone egress link. Ratio of total non-backbone ingress traffic to backbone egress traffic and vice-versa are little higher during loss period than the period when loss does not exist. Next, from the Figures we can observe the dual line pattern; which combined are from the same traffic and not a

separate session. However, the dual line implies that traffic fluctuations exhibit measure of continuity.



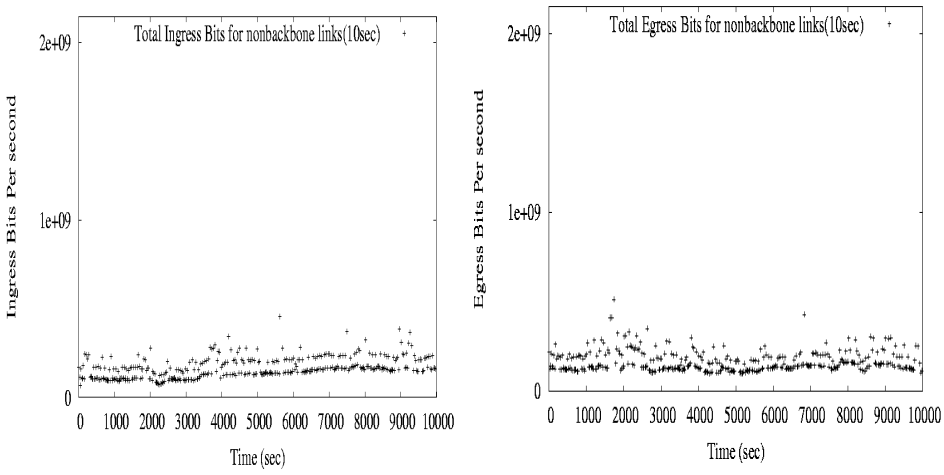
(a) Ingress bits **(a) Egress bits**

Figure 8. Total Non-backbone Ingress and Egress Bits Distribution with Loss



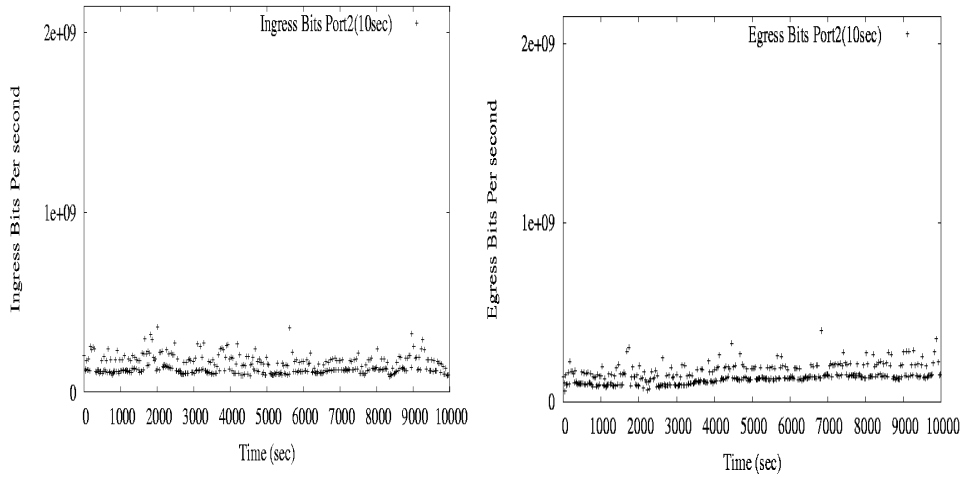
(a) Ingress bits **(a) Egress bits**

Figure 9. Backbone Ingress and Egress Bits Distribution with Loss



(a) Ingress bits **(a) Egress bits**

Figure 10. Total Non-backbone Ingress and Egress Bits Distribution without Loss



(a) Ingress bits **(a) Egress bits**

Figure 11. Backbone Ingress and Egress Bits Distribution without Loss

5.2 Protocol Based Analysis

We collected packet trace by installing TAP on the Nakwon link. From the collected packet trace we separated inter domain traffic; traffic that arrives at the monitored dormitory backbone switch from the Nakwon apartments and then further goes outside of POSTECH campus, and the intra domain traffic; traffic that arrives from Nakwon apartment and is routed inside the campus.

Figures 13 and 14 respectively illustrate the distribution of inter and intra domain bits per second over the entire day period. The marked regions are the periods when packet loss is detected. Figures portray that most of the traffic is routed outside the POSTECH campus and the intra domain traffic is miniscule.

Figure 15 depicts the distribution of packet loss over the entire day time period at 1 second measurement granularity obtained by polling Cisco enterprise MIB variables. The graph shows that the packet loss occurred on a port that is connected to the sub-dormitory switch for “Nakwon APT.” The link connected to this port is underutilized. From Figure 15 we can observe two distinct regions of losses. Clustered losses are observed near the end of the day.

It is evident from Figure 14 that the intra domain traffic is trivial amount and injects small peaks in traffic ones in a while, which may have generated due to worms. However, these peaks do not seem to have significant effect on loss. The intra domain traffic is caused mainly due to the ftp servers that are located inside campus. The number of ftp servers is large and they are distributed across the campus. Hence, this trivial amount of intra domain traffic can not create bottleneck at some particular ftp server.

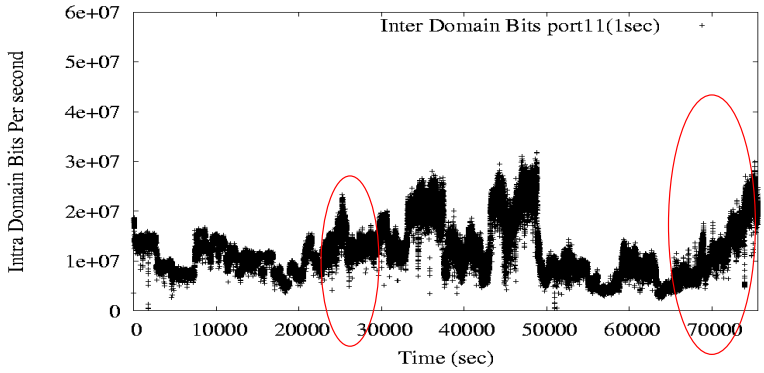


Figure 12. Inter Domain Bits Distribution

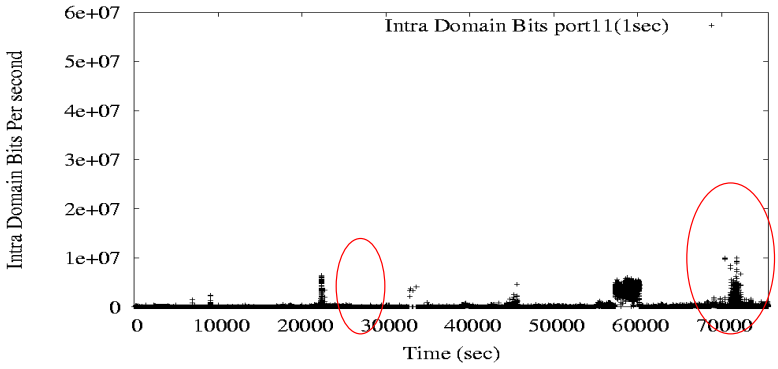


Figure 13. Inter Domain Bits Distribution

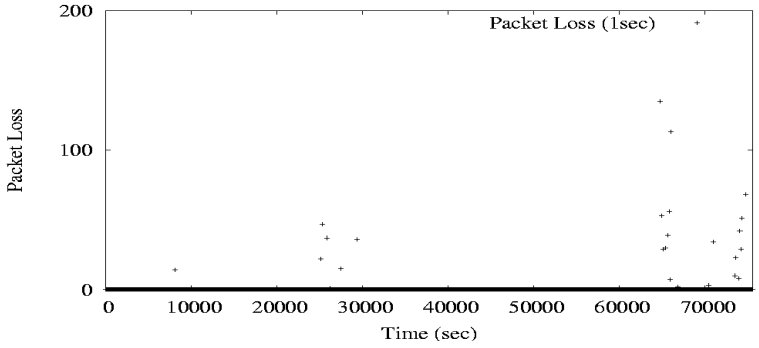


Figure 14. Packet Loss Distribution

Table 14 demonstrates the statistics of inter domain and intra domain bits in the unit of Mbps. From the Table inter domain traffic is 36 times higher than the intra domain traffic. Further, the standard deviation in the inter domain traffic is much higher than the intra domain traffic.

Nak-won link	Mean	Standard deviation	Max	Utilization (AVG/BW)	Max/BW
Intra Domain	0.351	1.681	0.926	0.0003	0.0009
Inter Domain	11.837	5.228	31.785	0.0118	0.0317

Table 14. Inter/Intra Domain Bits statistics at 1 sec granularity with Loss

While analyzing traffic it is important to understand the composition of traffic that we are studying and be ware of the different protocols that are present. In further sub-sections we present our traffic composition in terms of various protocols and analyze them to check whether any particular protocol is related to the packet loss phenomenon. We have divided the traffic as IP and non-IP traffic. Next, the IP (inter/intra domain) traffic is categorized into TCP, UDP and other; non-TCP and non-UDP but IP traffic.

5.2.1. IP Traffic Analysis

Figures 15 proclaim the distribution of inter/intra domain TCP bits at 1 second granularity, obtained by analyzing the packet trace collected from Nakwon link. Big chunk (96%) of traffic in the monitored link is composed of TCP. Hence, the volume of this protocol traffic is high and many peaks can be observed. Some of these peaks that are marked in the Figure do match with the packet loss time. The fact that the burst in the ingress packets is one of the reasons of the packet loss is confirmed here. TCP protocol is used for the long file transfers and TCP uses slow

start algorithm. Because of these properties TCP packets and their flows can be suspected as one of the reasons of the loss. Hence we have analyzed them in detail in later sections.

Figures 16 disclose the distribution of inter/intra domain UDP bits at 1 second granularity over same 1 day time period. UDP packets are the second highest number of packets in the traffic and mostly used for multimedia application traffic. In inter-UDP plot we can observe that the ingress bits are little higher in number during loss period (near end of the day) than rest of the day. However, compared to TCP traffic UDP traffic is really small. Moreover, UDP is not greedy protocol as TCP (TCP connections generally try to capture as much as bandwidth is available making UDP transfers to starve). Hence, we concentrate on TCP traffic more.

Figures 17 illustrate inter/intra domain other traffic, which is non-TCP and non-UDP but IP traffic. This traffic is mainly composed of ICMP protocol. However, ICMP bits are really few in number and showed no special characteristics.

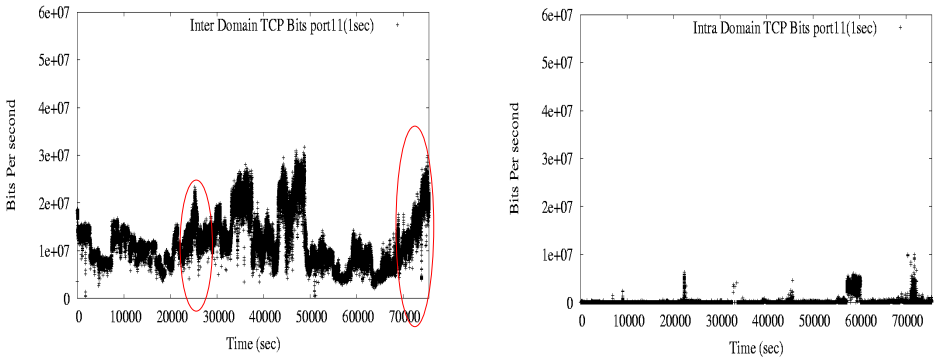


Figure 15. Inter and Intra Domain TCP Bits Distribution

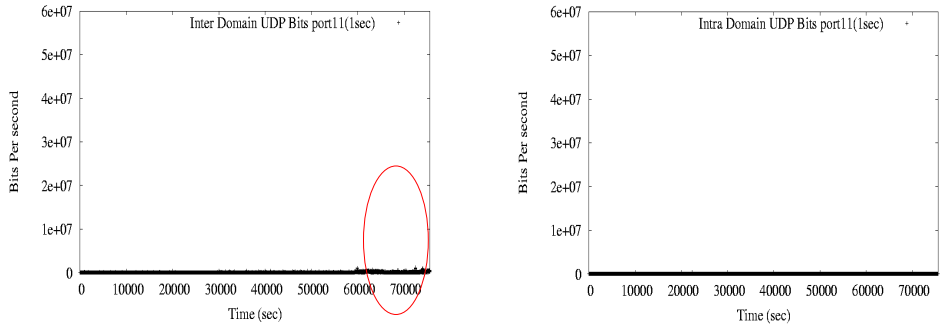


Figure 16. Intra and Intra Domain UDP Bits Distribution

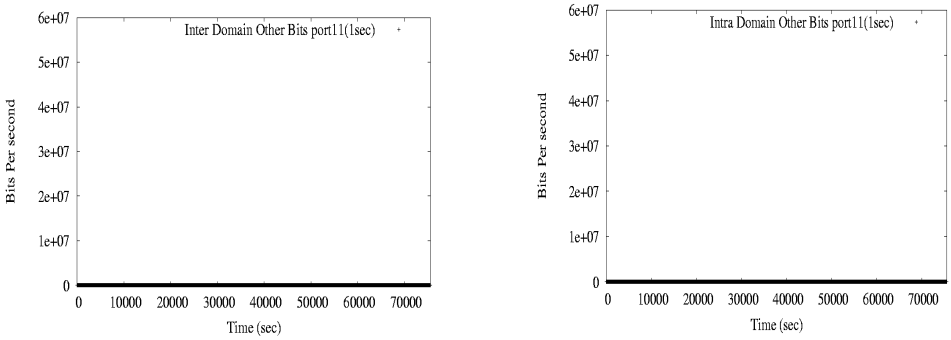


Figure 17. Inter and Intra Domain Other Bits Distribution

Table 15 demonstrates the IP traffic composition in percentage and the numbers indicate the sum of Megabits of particular protocol received during entire day. From the Table we get the same conclusion as the Figures above. TCP traffic is present is highest percentage, next is UDP and other is miniscule.

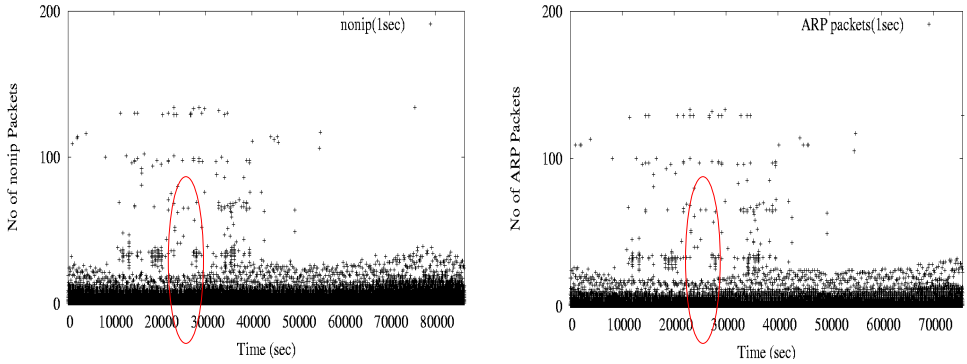
Nak-won link	TCP	UDP	Other
Intra Domain	26348 (2.8%)	221 (0.02%)	28 (0.003%)
Inter Domain	891082 (96.7%)	3665 (0.39%)	111 (0.01%)

Table 15. IP Traffic Composition

5.2.2. Non-IP Traffic Analysis

This section presents analysis of Non-IP traffic. Burst of spurious or malicious packets that are generally non-IP packets could be present in our network which leads to losses. Hence, it is imperative to verify what percent of non-IP packets exist in the network and what type of packets are these. Figure 18 (a) illustrates distribution of non-IP packets over entire day time period at granularity of 1 second. First half of the Figure shows bursty and clustered non-IP packets and it matches with one of the loss regions. However, these packets are too few in number.

By analyzing further we discovered that most of the non-IP packets in our network are ARP packets. Distribution of ARP packet is shown in Figure 18 (b). ARP request packets are the MAC layer broadcast packets. They induce processing overhead whenever received at the switch. Hence, we separated and plotted ARP request and reply messages which are demonstrated in Figure 19. We can observe that ARP request packets are much more in number and they are bustier in nature than the ARP reply packets. However, they probably do not interfere with the stable switch operation and cause packet loss as ARP packets are really few in number.



(a)NON-IP (b)ARP

Figure 18. Non-IP and ARP Packet Distribution

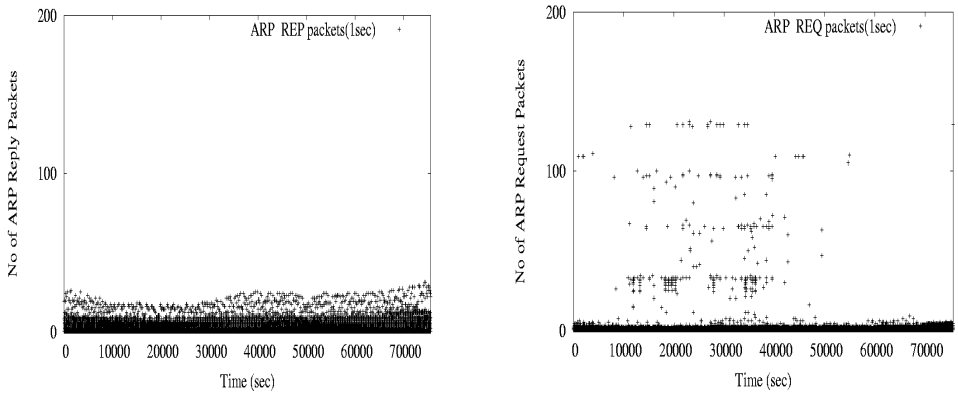


Figure 19. ARP Request and Reply Packet Distribution

5.3 Broadcast Packet Analysis

Purpose of this analysis was to determine if any relationship exists between IP level broadcast packets and packet loss phenomenon. Figure 20 illustrates distribution of broadcast packets. Many peaks in the broadcast packets distribution can be observed over the entire day time period. However, the highest burst of packet can be seen near the end of the day and this peak timing roughly matches with the packet loss time.

All these broadcasts tie up system resources as well as consume network bandwidth [16]. Every node (Switch, PC, server, printer, or any other network-attached device) in a given broadcast domain must process each broadcast frame it receives. When a node receives a broadcast, it generates an interrupt. In turn, each interrupt consumes some amount of processing time by the node. Excessive amounts of broadcast traffic not only waste bandwidth, but also degrade the performance of every device attached to the network. Hence, it may be the case that this moderate amount of broadcast packets observed during the loss period, degrade the switch

performance to some extent for that time period. However, the broadcast packets are not present in excess amount so probably they are handled well by the switch.

Figure 21 demonstrates the distribution of CPU utilization over entire day, which is obtained by polling Cisco enterprise MIB variable. Average CPU utilization of this switch is little higher than expected. Moreover, it is higher at the end of the day when packet losses are detected. High CPU utilization during loss time could be due to the burst in the broadcast packets or due to burst in incoming packets or bytes in smaller time granularity or due to congestion, which makes switch highly busy for that time period. However, some losses are observed even when the CPU utilization is not very high and is around 45 percent.

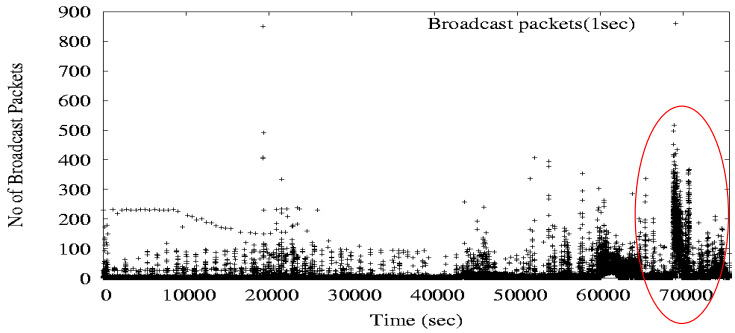


Figure 20. Broadcast Packets Distribution

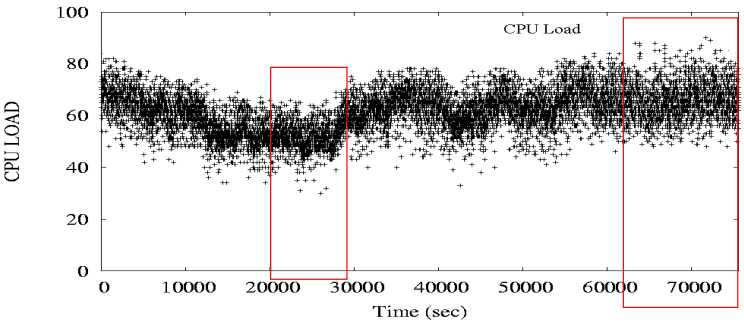
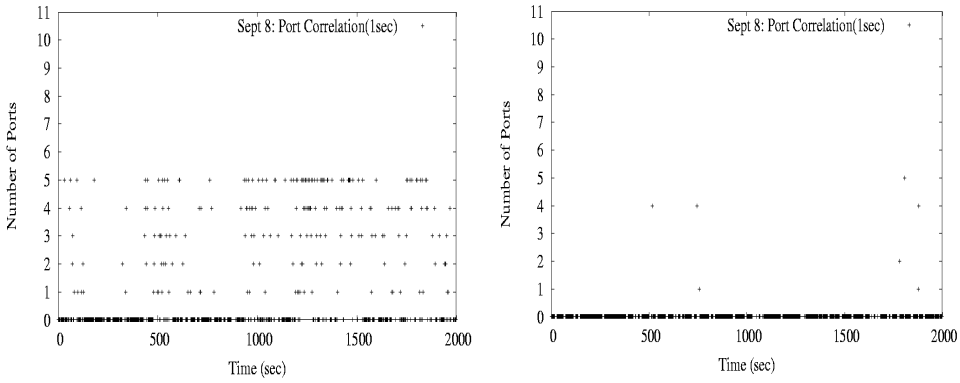


Figure 21. CPU Utilization

5.4 Loss Correlation

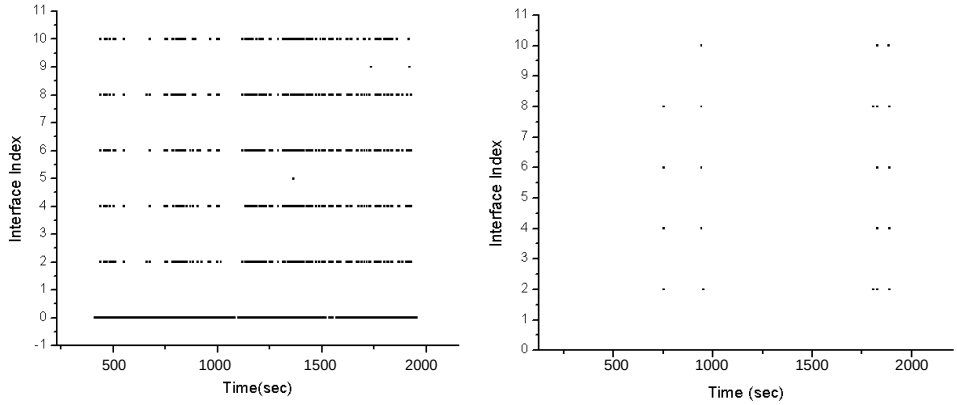
Using SNMP we collect packet loss counts for all the interfaces of the monitored switch. Figure 22 (a) and (b) illustrates the number of interfaces those experience the packet loss event simultaneously. Figure 22(a) is plotted for the 30 minutes time interval when heavy losses are experienced by many (upto 5) interfaces concurrently. This Figure clearly indicates that the switch is congested during this time and losses, experienced by different interfaces, are strongly correlated to each other. Whereas, Figure 22(b) depicts the 30 minutes time period when rare losses are experienced by few interfaces. Though, the losses are few in this time period still they are correlated to each other. This shows that the losses are always correlated in our measurement environment.

Each dot in Figure 23 (a) and (b) represents when packet loss was experiences by a particular interface. Both the Figures show that the correlated losses are mainly experienced by interface 2, 4, 6, 8 and 10. During these both times Nakwon link does not encounter any loss.



(a) Heavy Losses **(b) Few Losses**

Figure 22. Number of Interfaces Experiencing Loss Simultaneously



(a) Heavy Losses **(b) Few Losses**

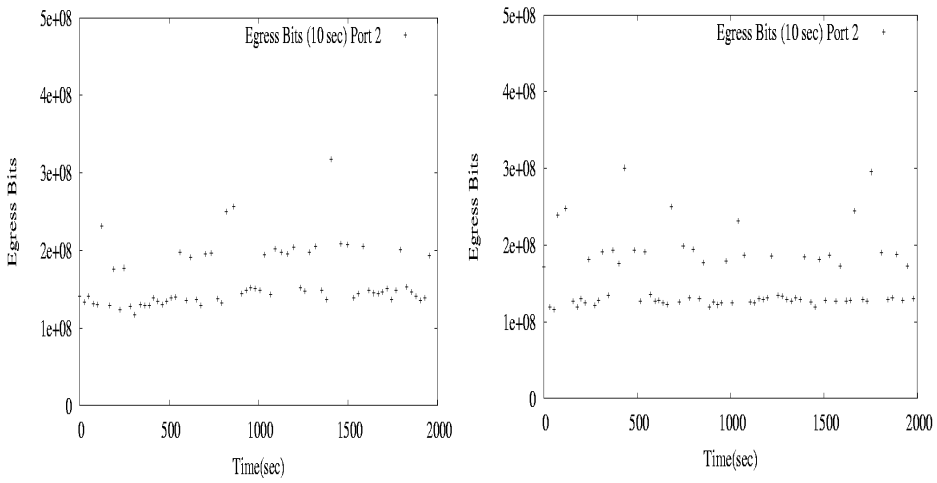
Figure 23. Index of Interfaces Experiencing Loss Simultaneously

5.5 Flow Analysis

We have seen in section 6.2 that the TCP traffic occupies the highest (96) percentage of the overall traffic. Hence, in this section we concentrate only on TCP flow analysis to determine the root cause of packet loss. We select two time period as described in section 6.4; one which has strongly correlated losses along with heavy losses on backbone link (interface 2) and other period when really few losses are detected on backbone link. During both these time periods losses on the other links are mainly caused due to egress queue drop and no loss is detected on the Nakwon link. Figures 24 and 25 illustrate the egress bits distribution at 10 seconds time scale and egress packet drop distribution at 1 second time scale for both heavy loss and rare loss time periods. Egress traffic is bit higher during the heavy loss period (mean 161Mbps) than the rare loss period (mean 155Mbps). The losses are distributed over entire time period though they are more clustered in the second half of the Figure 25(a).

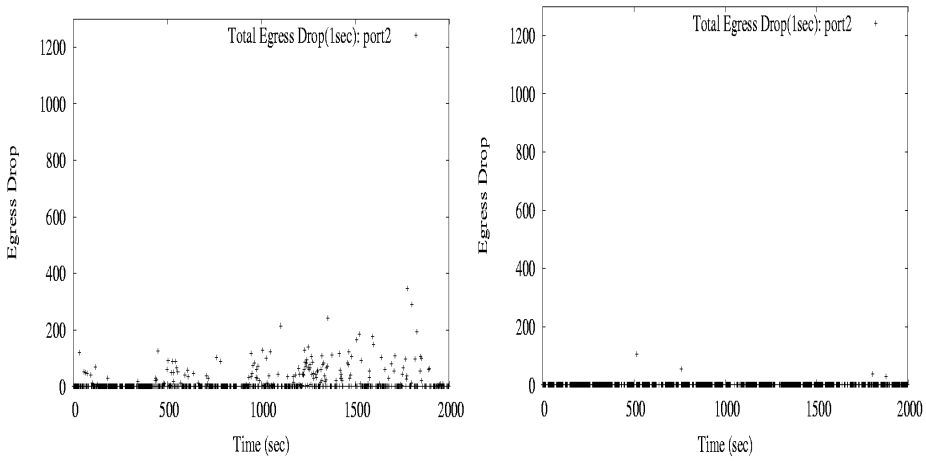
In the monitored switch, traffic comes from Nakwon apartments to the interface 11 (connected to Nakwon link) and then it travels to interface 2 (connected to backbone link) which routes most of these packets outside POSTECH campus. Hence, when interface 2 experiences losses, it should be reflected from the TCP flow behavior of the Nakwon link. Because TCP flows travel from the Nakwon link to backbone link and they will experience the similar losses as encountered by interface 2.

This section presents the detail analysis of TCP flows, obtained by analyzing the packet traces collected from the Nakwon link, with respect to various parameters like flow count, lifetime, size and etc.



(a) Heavy Losses **(a) Rare Losses**

Figure 24. Egress Bits Distribution at Backbone Link



(a) Heavy Losses **(a) Rare Losses**

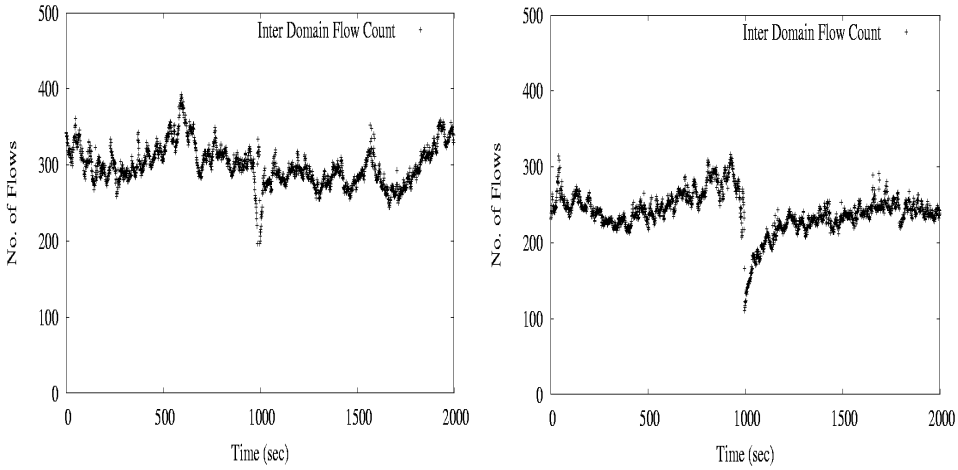
Figure 25. Egress Packet Drop Distribution at Backbone Link

5.5.1. Flow Count Distribution

We generated the 4 tuples: source address, destination address, source port, destination port, based TCP flows separately for intra domain and inter domain traffic. Figures 26 and 27 illustrate the distribution of inter/intra domain flows during the heavy loss period and rare loss period. From the Figures we can observe that the inter domain flows are higher in number than the intra domain flows. Which is obvious as the inter domain traffic is much higher than the intra domain traffic.

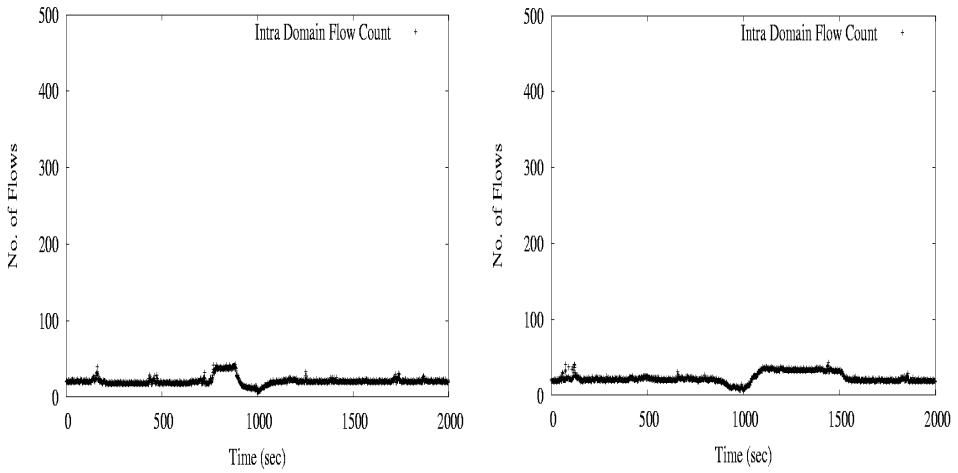
We were interested in intra domain flow analysis because the end-to-end delay of TCP on an internal connection is bound to be nontrivially smaller than inter-domain connections which then impacts TCP congestion control and its associated losses: (a) long feedback delays in inter-domain flows slow down reactiveness (oscillatory period is longer), (b) depending on the inter-domain end-to-end bandwidth, one suspects that its overall more constricted than POSTECH's internal bandwidth. However, intra domain flows are too few in number compared

to inter domain flows to make any difference. Next, from graphs we can observe that the number of inter domain flows are higher during heavy loss period than the rare loss period.



(a) Heavy Losses **(a) Rare Losses**

Figure 26. Inter Domain TCP Flow Distribution



(a) Heavy Losses **(a) Rare Losses**

Figure 27. Intra Domain TCP Flow Distribution

Tables 16 and 17 respectively, demonstrate the TCP counts for heavy loss and rare loss period. We can observe that the total numbers of flows are higher during heavy loss period. Tables 18 and 19 respectively, illustrate statistics for inter/intra domain flows during heavy loss and few loss period. Tables give the same conclusion that the mean and maximum value of inter domain flows are higher during the heavy loss period than the rare loss period. However, intra domain flow statistics are comparable during both time periods.

Interface	Total flows	Total Intra Domain flows	Total Inter Domain flows
Nakwon Link	15257	1268	13989

Table 16. TCP Flow Counts with Heavy Loss

Interface	Total flows	Total Intra Domain flows	Total Inter Domain flows
Nakwon Link	14131	1261	12870

Table 17. TCP Flow Counts with Rare Loss

Nakwon Link	Mean	Max	Min	Standard deviation
Intra Domain Flow	20.699	43	7	5.33
Inter Domain Flow	300.11	393	197	26.277

Table 18. Intra/Inter Domain TCP Flow Statistics with Heavy Loss

Nakwon Link	Mean	Max	Min	Standard deviation
Intra Domain Flow	22.838	43	7	6.428
Inter Domain Flow	240.490	316	111	26.669

Table 19. Intra/Inter Domain TCP Flow Counts with Rare Loss

5.5.2. Flow Size Distribution

Figures 28 and 29 respectively, illustrate the flow size distribution for inter and intra domain traffic during the period when heavy losses are detected on backbone link and when few losses are detected on backbone link. In the inter-domain log-log plot of flow size, we approximately see the characteristic linear shape (with little curviness) which indicates a power-law flow size distribution stemming from a power-law file size distribution.

Internet file sizes have been discovered in the 1990s to be heavy-tailed, means if x is file size and f(x) its frequency then roughly, empirically the following functional form holds:

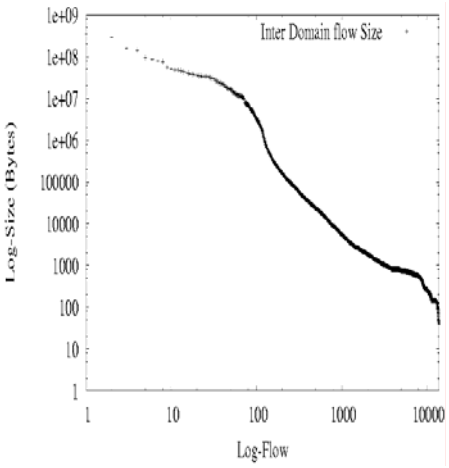
$$Pr[x = a] \sim= c\ x^{\{-alpha\}}$$

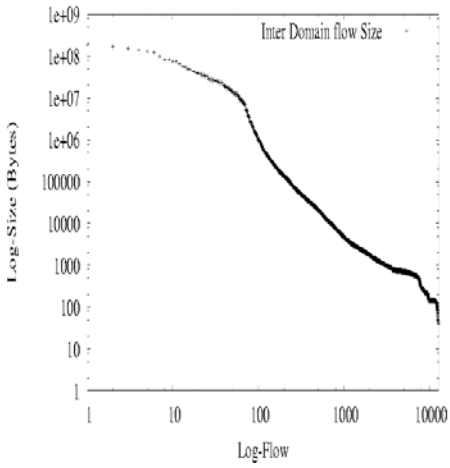
Where, alpha > 0 is a parameter in the range 1.1-1.3. Hence, by taking log on both sides we get,

$$Log\ Pr[x = a] \sim= Log\ c - alpha\ Log\ x$$

Which is a straight line (by viewing y = Log Pr[x = a] and z = log x as composite variables). Thus, to capture this characteristic linear shape we have plotted the flow sizes on the log scale. Further, we have ranked the flows with 1 being the flow with largest size and so on. Even after ranking the flows, the linear relation continues to hold, as can be seen from Figure 28.

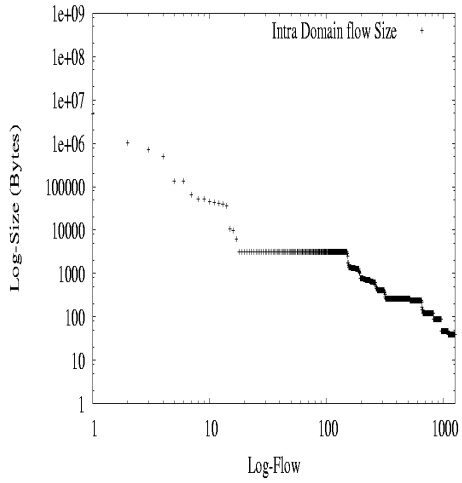
Next, Figure 28 (a) and (b) show that number of large (above 10MB) flows that are present during heavy loss period is little higher than the rare loss period.

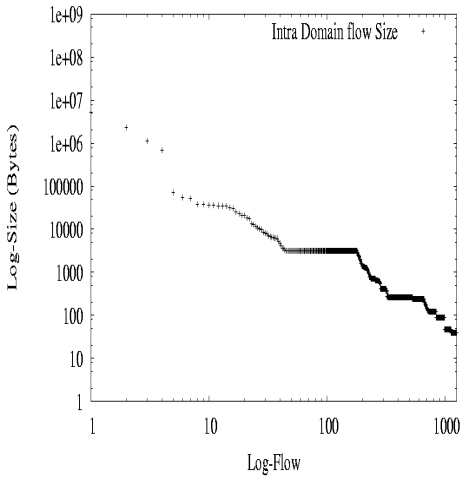




(a) Heavy Losses **(a) Rare Losses**

Figure 28. Inter Domain TCP Flow Size Distribution





(a) Heavy Losses **(a) Rare Losses**

Figure 29. Intra Domain TCP Flow Size Distribution

Tables 20 and 21 respectively, demonstrate the flow size statistics for inter domain and intra domain flows during heavy loss and rare loss period. During both the time intervals, mean flow size for the inter domain traffic is higher than the intra domain traffic, means larger files are transferred from internet than the routers inside the campus. Further, the variation in the flow sizes is larger for inter domain flows than the intra domain flows. Minimum flow size is 40 bytes; these flows are formed due to single ACK TCP packet.

Average and max flow sizes are higher during the heavy loss period than the no loss period; means flows are larger during loss period. Variation in the flow sizes is also higher during loss interval.

Nakwon Link	Mean	Standard deviation	Max	Min(Bytes)
Intra Domain Flow Size	0.0135	0.28	8.86	40
Inter Domain Flow Size	0.24	6.06	436.61	40

Table 20. Intra/Inter Domain TCP Flow size Statistics with Heavy Loss

Nakwon Link	Mean	Standard deviation	Max	Min(Bytes)
Intra Domain Flow Size	0.013	0.231	5.862	40
Inter Domain Flow Size	0.233	4.583	241.228	40

Table 21. Intra/Inter Domain TCP Flow size Statistics with Rare Loss

5.5.3. Flow Lifetime Distribution

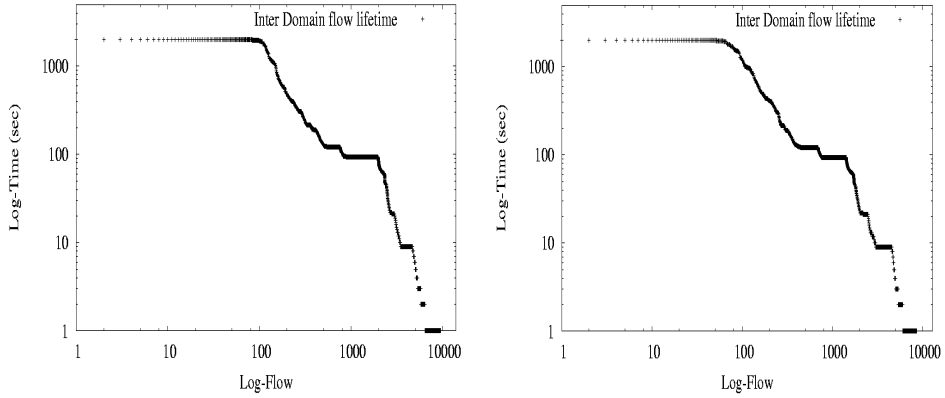
Figures 30 and 31 respectively, illustrate the TCP flow lifetime distribution for inter domain and intra domain traffic across the log scale for heavy loss and rare loss interval. The flows are ranked with flow 1 being the longest flow and so on.

Again, the log-log plots of flow lifetime closely depict the characteristic linear shape which indicates a power-law file size distribution.

Next, it appear from Figure 30(a) and (b) that flow lifetime plot is flat across first nearly 100 flows even though their sizes are different as seen from the flow size distribution graphs in previous sub-section. By carefully observing the data we determine that, different number of packets are received by different flows during the equal time interval, which makes the difference in the sizes of comparable lifetime flows.

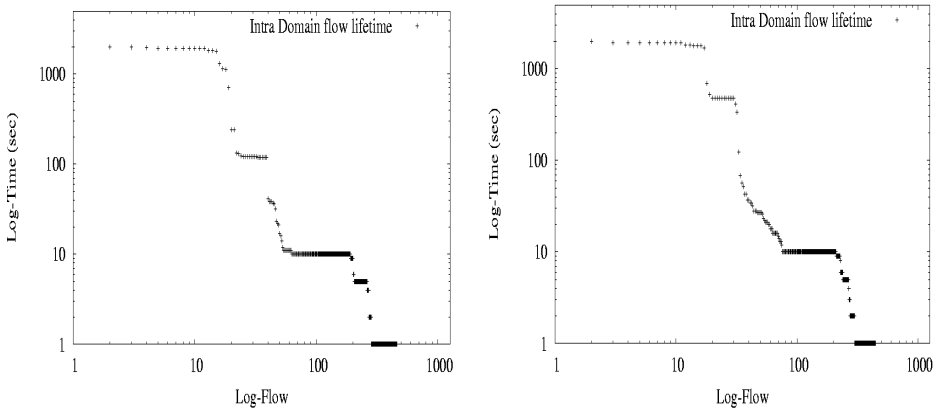
By comparing Figure 30 and 31 we can observe that number of long (100 second above) flows that are present in the inter domain traffic is higher than intra domain traffic. This is because inter domain flows are larger in size and the end-to-end delay of TCP is larger for inter domain traffic than intra domain traffic.

Next, Figures below depict that little bit higher number of long flows are present during heavy loss time as compared to rare loss period. Further, from data we observed that flows are longer during heavy loss period compared to rare loss period.



(a) Heavy Losses **(a) Rare Losses**

Figure 30. Inter Domain TCP Flow Lifetime Distribution



(a) Heavy Losses **(a) Rare Losses**

Figure 31. Intra Domain TCP Flow Lifetime Distribution

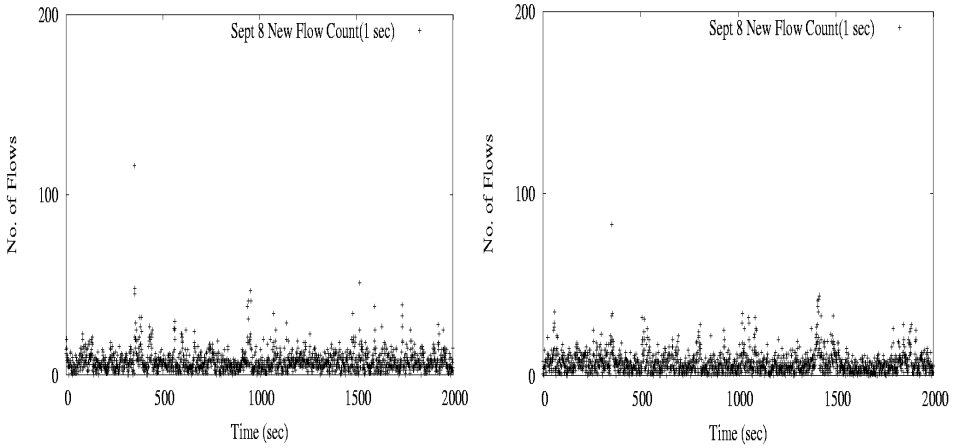
5.5.4. New and terminated Flow

Figures 32(a) and (b) demonstrate the new flows distribution when heavy packet loss is detected and when rare packet loss is detected. Each dot in the plot represents the number of new flows initiated in that particular second. Figures 33 (a) and (b) demonstrate the terminated flows distribution when heavy packet loss is detected and when rare packet loss is detected. Each dot in the plot represents the number of terminated flows in that particular second.

Plots for new flow count and terminated flow count are comparable in nature, which implies that almost similar number of flows are generated and terminated every second, means the ones that comes in that leaves. These flows are mostly the short flows. Similarity in the nature of these graphs confirms that the data behaves logically and data is consistent with what is expected.

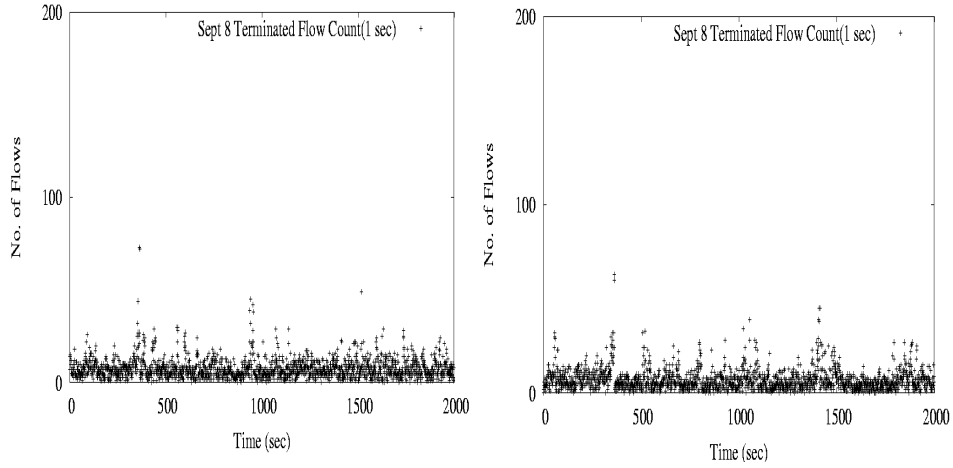
Further, the numbers of new and terminated flows are higher during the period when heavy loss is observed than the period when rare loss is observed.

Number of flows generated during some particular period shows the traffic diversity during that time. During loss period having more number of flows means having more versatile traffic which, in turn means having more unique combinations of the source address, destination address, source port and destination port. This could be caused due to various applications used during that time.



(a) Heavy Losses **(a) Rare Losses**

Figure 32. TCP New Flow Distribution



(a) Heavy Losses **(a) Rare Losses**

Figure 33. TCP Terminated Flow Distribution

Tables 22 and 23 respectively, show the statistics of new and terminated flows during the heavy loss and rare loss period. Tables indicate similar conclusion as the plots above, mean values for new and terminated flows are exactly same during both time periods.

Nakwon Link	Mean	Max	Min	Standard deviation
New Flow	7.660	116	0	6.039
Terminated Flow	7.660	73	0	5.624

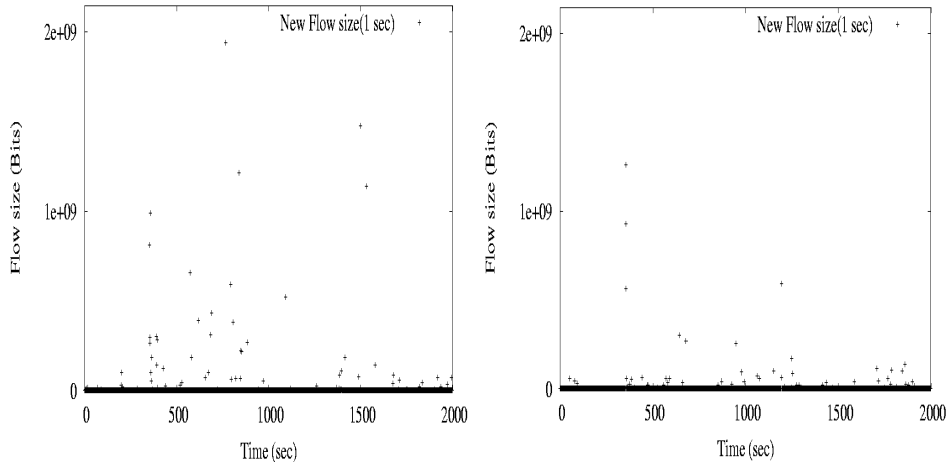
Table 22. TCP New/Terminated Flow Count Statistics with Heavy Loss

Nakwon Link	Mean	Max	Min	Standard deviation
New flow	7.081	83	0	5.753
Terminated Flow	7.081	63	0	5.733

Table 23. TCP New/Terminated Flow Count Statistics with Rare Loss

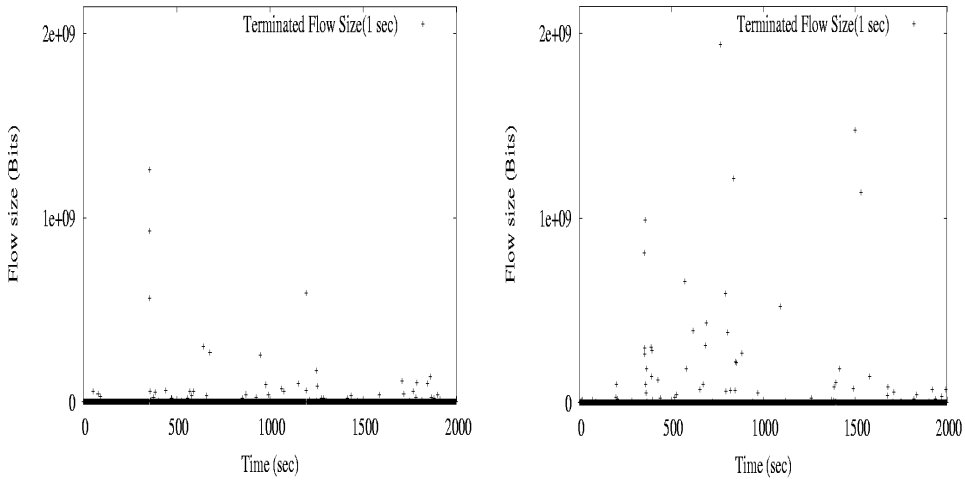
Figures 34 and 35 respectively, illustrate the distribution of sum of new and terminated flow sizes when heavy packet loss is observed and when rare loss is observed. Each dot in the plot represents the sum of flow sizes those are initiated in that particular second and same is with the terminated flow size plots. From the graphs we can observe that the sum of the flow sizes initiated in each second is closely related to the sum of flow sizes that are terminated in each second. These plots are in agreement with the new flow and terminated flow count plots that are described above.

In the Figures that represents sum of flow sizes during loss period, more points are observed that are above 1Gbits line than the Figures of no loss time means that the sum of flow sizes is larger during loss time than no loss time.



(a) Heavy Losses **(a) Rare Losses**

Figure 34. TCP New Flow Size Distribution



(a) Heavy Losses **(a) Rare Losses**

Figure 35. TCP Terminated Flow Size Distribution

5.5.5. Distinct Sources

Each dot in the Figure 36 and 37 represents the number of different destination addresses to which each distinct source go, during heavy loss and rare loss time period respectively. Next, the distinct source addresses are ranked in decreasing order with rank 1 being the source address that connects to highest number of distinct destinations and so on. We can observe a linear pattern from the log-log plots shown below.

From the plots below it is evident that the maximum number of distinct destinations (492) to which distinct source addresses connect during loss time is little higher than (460) the time when loss is not detected. This implies that source addresses are connecting to little more destinations during loss period. This result concurs with our previous finding that more number of flows are obtained during loss period because more pairs of distinct source and destination addresses are obtained.

However, numbers of distinct source addresses during both times are very close to each other like 63 for no loss time and 66 during loss time. This data is collected from the Nakwon link which is connected to 6 buildings each with 10 apartments. Each apartment is assigned a unique IP address which is a source address here. The expected number of unique source addresses is 60. However; they turn out to be little more than that because some apartments have hubs installed in them. Hence, more than one person can use the internet.

This analysis confirms the fact that IP spoofing is not present in our traffic. Further, the number of distinct source addresses observed in the data is very close to expected value. Hence, the data is behaving logically.

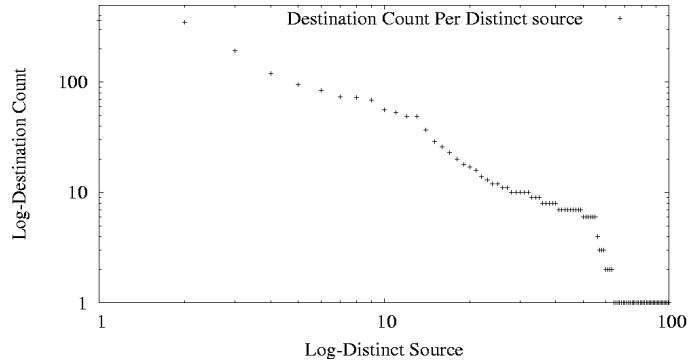


Figure 36. Distinct Destination with Heavy Loss

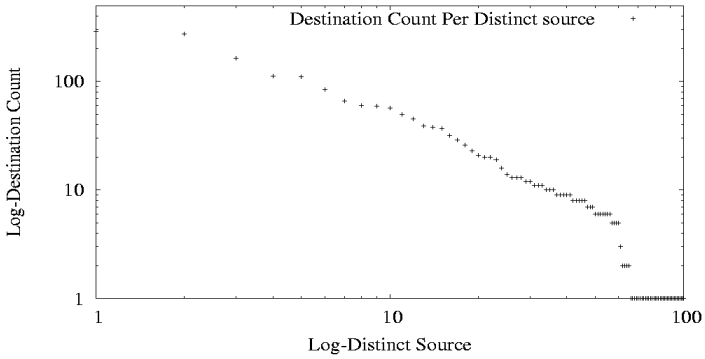


Figure 37. Distinct Destination with Rare Loss

5.5.6. TCP Large Flow Analysis

We are especially interested in analyzing large TCP flows. Because TCP flows are most frequently found in our data and they are large. When the flow is large it means it has more experience. Hence, if we analyze such a large flow, we can see what is happening in the link from the TCP flow experience.

Figure 38 illustrates the cumulative distribution of total traffic. x axis shows the flows that are ranked with flow 1 being the largest flow and so on and y axis depicts the cumulative distribution of total traffic in terms of percentage. From the

plot we can observe that the 10 longest flows contribute to 50% of the traffic and 50 longest flows contribute to 80% of total traffic. This proves that few large flows contribute to huge portion of the total traffic.

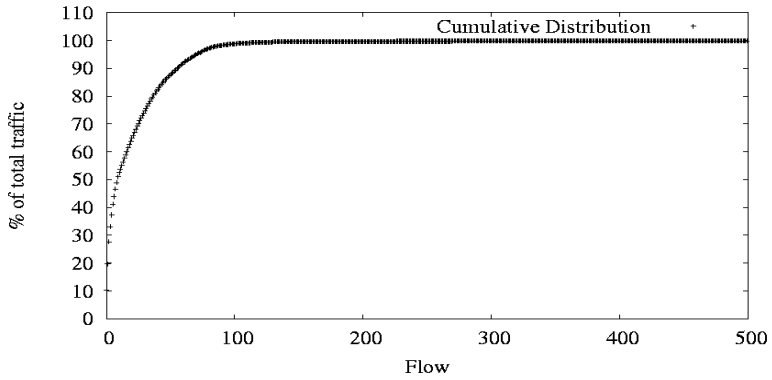


Figure 38. Traffic Cumulative Distribution

Selecting large flows for study is an important step and needs to be done efficiently. Some flows might be long however they may send very few data bytes. For example, http 1.1 makes a persistent connection. Consequently, if some person is surfing a site then flow generated from this connection could be very long but actual data transfer during this time will be less. Such a flow can tell little about the actually link or switch condition as it has little experience. The flows that are transferring data continuously over long time interval will have the real experience, which will reflect the link condition with high reliability.

Hence, we selected a reasonably large time interval of 15mins from the heavy loss and rare loss time periods. Next, we generated and selected the 10 longest flows traveling through these time intervals. We plotted sequence number distribution and throughput in terms of bytes per second for these flows over entire 15 min interval to study their overall behavior; whether they are sending multiple files or single file, if they are any ideal periods.

The Figures 39 to 42 illustrate sequence number and throughput distribution for the two flows out of selected 10 largest flows during heavy and rare loss period. We observed that flows are larger in size during loss period and all of them are sending data continuously over entire 15min without having any sleeping; ideal period. Some of the flows during rare loss period are large in size however; they are shorter (in time) than loss period flows.

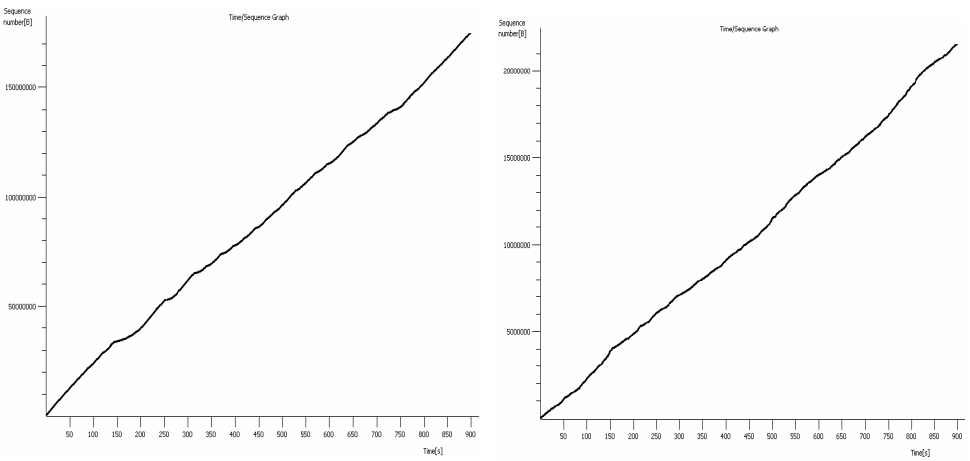
From the sequence number distribution (Figure 39 and 41) we can coarsely observe that the data delivery during rare loss period looks smoother and the flows during loss period are showing more variations in the slop. Height of sequence number plots during both heavy and rare loss period is high. Because these flows are traveling through 1Gbps underutilized link and probably they have only one bottleneck at dorm backbone link that slows them down by small amount so they can transfer data at high rate.

Flow 1 during heavy loss period and rare loss period respectively have size of 179MB and 140 MB during these 15 min interval. From Figure 40(a) and 42(a) we can see that the flow 1 during loss period is facing too many variations in the throughput and take longer time to deliver data while flow 1 during rare loss period is experiencing few variations in the throughput and hence comparatively takes lesser time to transfer data.

We can observe that the throughput of the Large TCP flow is lower during heavy loss period than the rare loss period, which is consistent with what should happen. These flows are generated from the traffic that is collected from Nakwon link. After these flows enter the switch through Nakwon link, they travel to interface 2 (backbone link) and from there they are routed outside the POSTECH campus. When packets are lost on interface 2, no ACK packet is received back. Hence, the

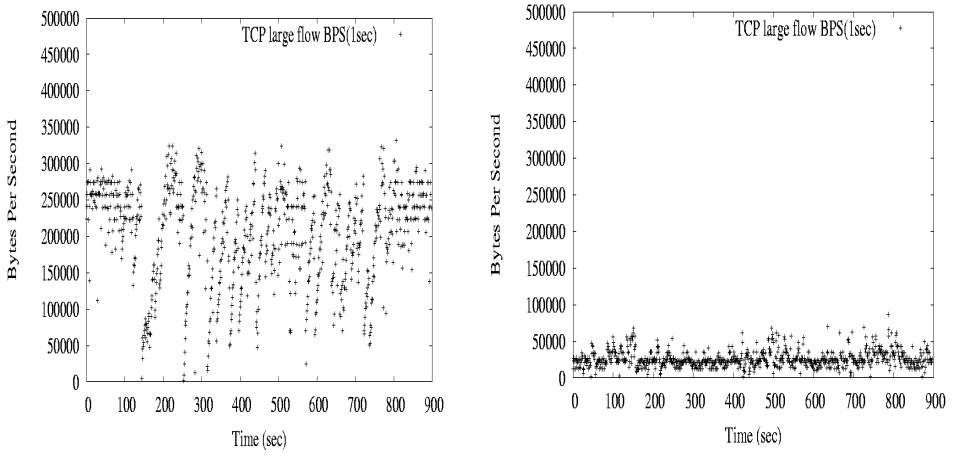
TCP flow throttle down and reduces its throughput. The lower throughput of the flows gives a hint that probably the flows are experiencing packet losses.

Next, we focus on a small time interval like 50 second to study large TCP flows so that we can distinguish their detail characteristics like transmission timeout events, retransmission of same sequence number packets, and etc.



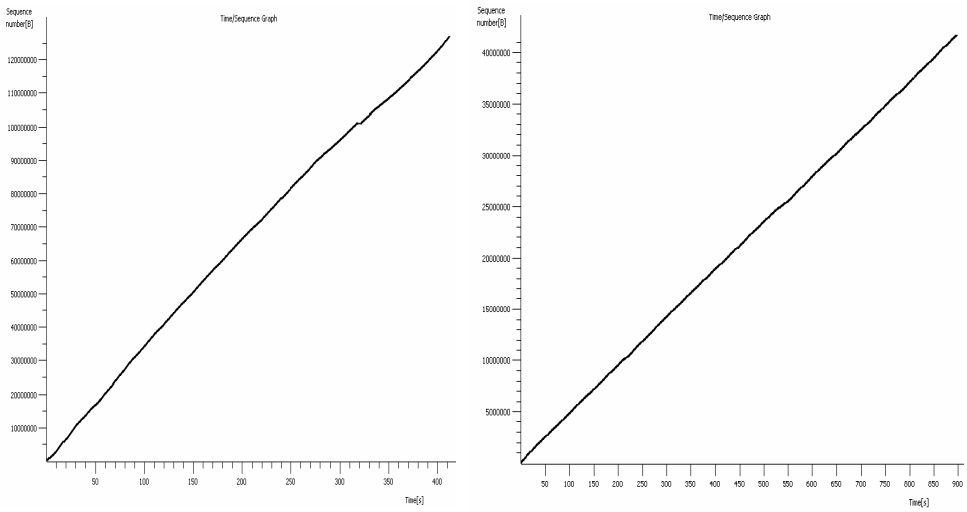
(a) Flow 1 **(b) Flow 2**

Figure 39. TCP Large Flow Sequence Number distribution with Heavy Loss



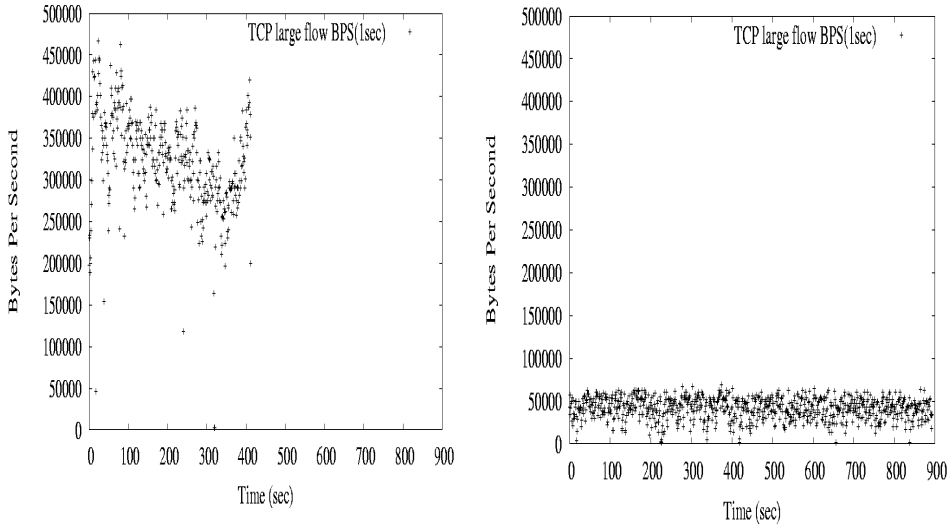
(a) Flow 1 **(b) Flow 2**

Figure 40. TCP Large Flow Throughput Distribution with Heavy Loss



(a) Flow 1 **(b) Flow 2**

Figure 41. TCP Large Flow Sequence Number distribution with Rare Loss



(a) Flow 1 **(b) Flow 2**

Figure 42. TCP Large Flow Throughput Distribution with Rare Loss

5.5.6.1. Throughput Analysis

Figures 43 (a) and (b) respectively illustrate the throughput of large TCP flow in terms of packets per millisecond when heavy losses are detected and when rare losses are detected for a small zoomed in time interval. The characteristic sawtooth pattern was not clearly visible from the one second granularity plots. Hence, the throughput is plotted at 1 millisecond time granularity.

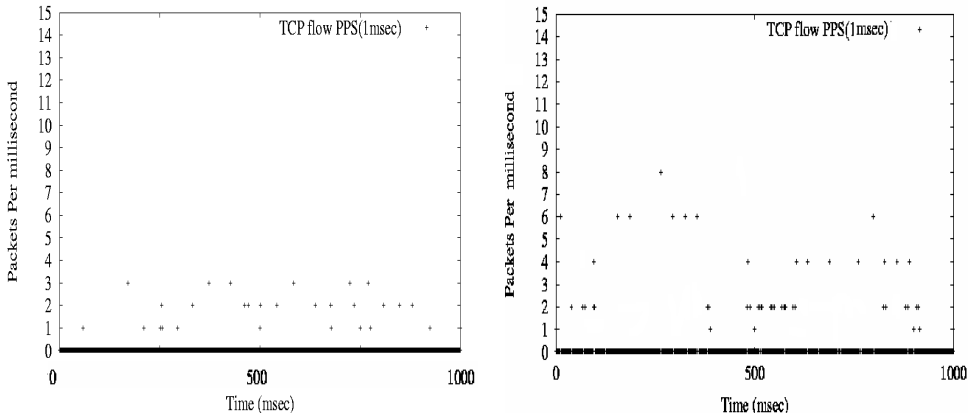
From both heavy loss and rare loss plots the sawtooth pattern is clearly visible, which is obtained due to linear increase/exponential decrease TCP congestion control mechanism. Further, height of sawtooth is lesser for the flows those experience frequent packet losses than the flows those encounter few losses. While delivering packets TCP starts with transmission window of size one and increases it at linear rate. However, if for certain transmitted packet, ACK packet is not received within the timeout period, congestion is detected and the TCP flow backs off exponentially by reducing the transmission window size, then again starts increasing the window and so on. Hence, when losses are occurring frequently TCP can not increase the window size to very high and it has to back off frequently.

Sometimes ACK packet can not be received within timeout interval due to packet delay, which also indicates existence of congestion but not the packet loss. Thus, all TCP backoff events do not necessarily imply occurrence of packet loss still roughly from overall TCP behavior losses are indicated here.

Figures 44 (a) and (b) respectively illustrate the distribution of throughput in term of bytes per seconds. The Figures concur on the conclusions that are drawn from the Figure 43.

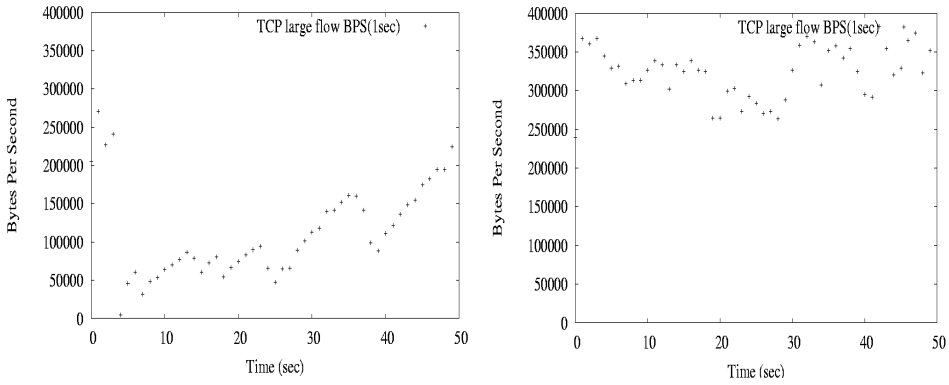
Next, we studied the destination addresses of these flows. They testify that the flows from both heavy and rare loss time periods, do not share the resources

outside the POSTECH campus as they do not go to the same destination. This reveals that probably, these flows have only dorm backbone switch as their dominant shared bottleneck.



(a) Heavy Losses **(a) Rare Losses**

Figure 43. Large Flow Throughput in Packets per Millisecond



(a) Heavy Losses **(a) Rare Losses**

Figure 44. Large Flow Throughput in Bytes per Second

5.5.6.2. Sequence Number Analysis

Figure 46 illustrates the sequence number distribution for a TCP stream during loss period. These Figures demonstrate only a small zoomed in portion (50 seconds interval) of a large TCP stream. Figures 46 and 47 demonstrate many variations in the slop, which indicate changes in the data transmission rate of TCP flow. We can observe regions in Figure 46 that indicate a small ideal period and then retransmission of same sequence number packet which, implies that the particular packet was timed out; no ACK packet was received within the timeout period. Timeout of a packet leads to congestion prediction in TCP, which in turn causes TCP stream to backoff by reducing the transmission rate (decrease in slop) as can be seen from the Figure. We can observe many retransmission event occurrences and variations in the transmission rates from the Figure coarsely indicating that this stream was facing congestion situation and in turn the packet loss events.

Using our analysis tools we can calculate the loss rate seen by the TCP flow. This flow is experiencing the loss rate of 37 packets per second. Average loss rate obtained by SNMP measurements during this time interval is 40 packets per second which are comparable. This implies that the TCP flows are experiencing similar losses that are seen at backbone link.

Figure 45 illustrates packet loss distribution at 1 second granularity. Even though the TCP flows are experiencing similar number of losses as are seen at backbone link, we can not match exact time of packet loss seen by TCP flow with time when the SNMP measurements are showing losses. Reason is that, at the granularity of one second SNMP measurements are not accurate as discussed in section 3.1.1. Because of the low response rate of SNMP there are many blank spots in the data.

It may be the case that when these large flows try to send data at high rate, they reach a saturation phase and create microcongestion in the switch which leads to losses. Means these large flows might be responsible for the losses on account of their linear increase/exponential decrease congestion control mechanism. However, the evidences that we have are not enough to draw any flawless scientific conclusion.

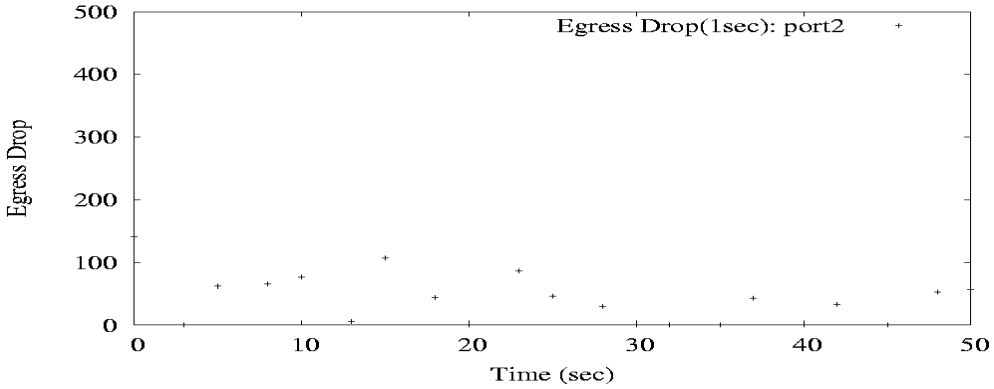


Figure 45. Egress Packet Drop Distribution at Backbone link

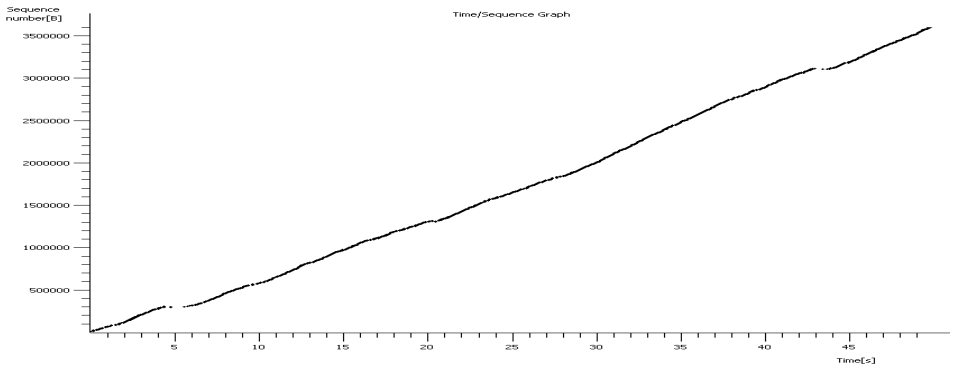


Figure 46. Sequence Number Distribution for Flow1 with Heavy Loss

Figures 48 and 49 demonstrate the sequence number distribution for two separate flows when packet rare losses are indicated by SNMP measurements. The

plot indicates smooth data delivery as not many losses are experienced by these flows. However, there are some regions where we can observe little decrement in slop means apparently the timeout or packet loss event. However, the SNMP measurements at the backbone link show that no losses exist during this time either because no SNMP response was received back from the switch or there are really no losses during that time. In later case, we can conclude that losses that are seen by this flow are not visible at the dorm switch backbone link may be because these flows have some other private bottleneck and at that bottleneck these losses will be visible.

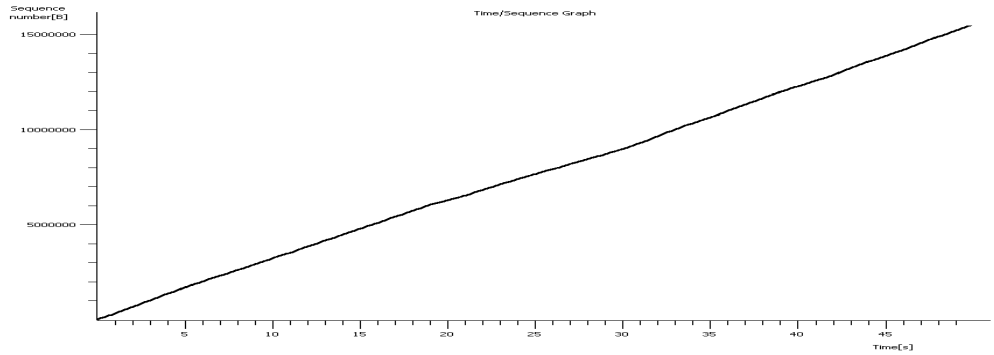


Figure 47. Sequence Number Distribution for Flow1 with Rare Loss

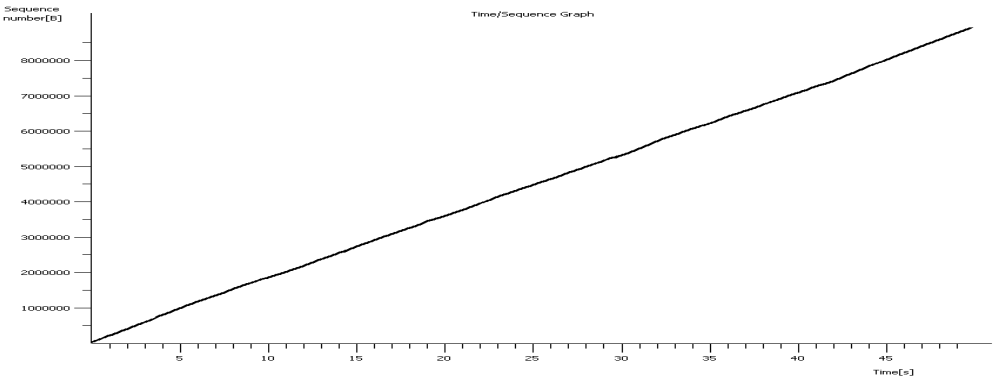


Figure 48. Sequence Number Distribution for Flow2 with Rare Loss

6 Concluding Remarks & Future Work

In this work, we studied traffic and packet loss characteristics to determine the root cause of packet loss in the underutilized enterprise network links. We monitored the selected link and switch using our traffic monitoring modules; SNMP polling and packet capture that are implemented in our Linux monitoring system. We defined clear and complete framework to approach the problem in logical and systematic way. We developed efficient tools to execute the framework and analyze packet loss with various types of traffic metric at various time granularities.

Our analysis confirms that the switch we selected for study is highly underutilized along with all the links connected to it. TCP traffic is occupying the highest percentage (96%) of the total traffic. The analysis divulges that non-IP traffic is negligible and malicious traffic is not found in our link. Reasonable amount of broadcast packets exist in monitored traffic however they hardly seem to interfere with the stable switch operation to cause losses.

The packet loss analysis established that the losses are highly correlated; generally losses are detected on multiple links at the same time. We discovered that number of TCP flows is always higher during the time when losses are detected. Especially, large (in size and duration) TCP flows are present in our link. Few large flows occupy 80~90% of the total traffic and they are trying to deliver data at high rate. Our analysis provide evidences that the large TCP flows that are mainly cause due to large file transfer might be related to losses on account of their linear increase/exponentially decrease congestion control mechanism. However, we can not come to legitimate conclusion due to the SNMP inaccuracies that causes problems in verifying the exact correlation of TCP flow behavior with the packet loss events.

Our analysis methodology and the collection of tools that we have developed are efficient, independent and flawless. They provide impeccable logical, structured and systematic way to approach the complex problem that we are dealing with in this project. This methodology and tools provide the right direction and strong base to this work, which will be of great use for future research. They are the valuable **contribution** from this work and will be used in future while continuing work on this topic.

For future work, we will do more sophisticated analysis by avoiding SNMP inaccuracies to determine the relationship between large TCP flow and packet loss. We will collect traffic in both ingress and egress direction of the same link and from the backbone link which will help to study TCP characterization in further detail.

We have studied traffic characteristics at the edge switch. It will be interesting to see if the traffic across core switches also has the similar characteristics. Further, large TCP flows are present in our traffic and hence, can be suspected as one of the loss reasons. In future, we can check if losses exist in the links that do not convey large TCP flows. We will try to collect data sets from different networks and link to verify our analysis and results. One of the data sets could be obtained from IPv6 networks. It will be interesting to study the traffic and loss characteristics on the IPv6 network as in future all networks are expected to turn into IPv6. Finally, if we can confirm that large TCP flows are the actual reason of loss then we should determine a way to avoid these losses. Probably, by designing a switch that can handle the microcongestion created by large TCP flows.

References

[1] Konstantina Papagiannaki, Darryl Veitch and Nicolas Hohn, “Origins of Microcongestion in an Access Router,” Passive & Active Measurement Workshop, Antibes, France, April, 2004.

[2] Konstantina Papagiannaki, Rene Cruz and Christophe Diot, “Network Performance Monitoring at Small Time Scales,” Internet Measurement Conference, Miami, Florida USA, October 2003.

[3] N. Hohn, D. Veitch, K. Papagiannaki and C. Diot, “Bridging Router Performance and Queuing Theory,” ACM SIGMETRICS, New York, NY, 2004.

[4] Klaus Mochalski, Jorg Micheel and Stephen Donnelly, “Packet Delay and Loss at the Auckland Internet Access Path,” Passive and Active Measurement Workshop, Fort Collins, Colorado USA, March 2002.

[5] Seungh-Hwa Chung, Deepali Agrawal, Myung-Sup Kim, James W. Hong, and Kihong Park, “Analysis of Bursty Packet Loss Characteristics on Underutilized Links Using SNMP ”, E2EMON, San Diego, California, USA, October 2004.

[6] Seungh-hwa Chung, “Analysis of Bursty Packet Loss Characteristics on Underutilized Links,” MS Thesis, Dept. of Computer Science and Engineering, POSTECH, Feb. 2005.

[7] Seung-Hwa Chung, Young J. Won, Deepali Agrawal, Seong-Cheol Hong, James W. Hong, Hong-Taek Ju and Kihong Park, “ Detection and Analysis of Packet Loss on Underutilized Enterprise Networks”, E2EMON, Nice, France, May 2005.

[8] J.Case, M. Fedor, M. Schoffstall and J. Davin, “A Simple Network Management Protocol,” RFC 1157, May 1990.

[9] Endace, DAG 4.3GE card, User Manual, EDM01.02-01

[10] Endace, DAG Programming Guide, API v1.1

[11] Cisco, “MIB Compilers and Loading MIBs,” Cisco Technical Notes, http://www.cisco.com/en/US/tech/tk648/tk362/technologies_tech_note09186a00800b4cee.shtml.

[12] Cisco, “Input Queue Overflow on an Interface,” Cisco Technical Notes, http://www.cisco.com/en/US/products/hw/modules/ps2643/products_tech_note09186a0080094a8c.shtml.

[13] Tom Chirstiansen and Nathan Torkington, “Perl Cookbook”, First Edition, August 1998.

[14] Thomas Williams and Colin Kelley, “An Interactive Plotting Program”, a brief Manual and Tutorial, December 1998.

[15] Leslie Lamport, “LATEX A document Preparation System User’s Guide and Reference Manual”, Second Edition, 1994.

[16] DELL, “How Much Broadcast and Multicast Traffic Should I Allow in My Network”, Power Connect Technical Notes 5, November 2003.

[17] Kihong Park, “Internet as a Complex System”, SFI Workshop, March 2001.

[18] Zhi-Li Zhang, Vinay J. Ribeiro, Sue Moon and Christophe Diot, “Small-Time Scaling Behaviors of Internet Backbone Traffic: An Empirical Study”, INFOCOM, San Francisco, USA, March 2003.

[19] Wenyu Jiang, Henning Schulzrinne, “Modeling of Packet Loss and Delay and Their Effect on Real-Time Multimedia Service Quality”, ACM NOSSDAV, Chapel Hill, North Carolina, June 2000.

[20] Velibor Markovski, Fei Xue, and Ljiljana Trajkovic, “Simulation and Analysis of Packet Loss using User Datagram Protocol”, The Journal of Supercomputing, Kluwer, vol. 20, no. 2, pp. 175-196, Sept. 2001.

[21] F. Xue, V. Markovski, and Lj. Trajkovic, “Packet loss in video transfers over IP networks”, Proc. IEEE Int. Symp. Circuits and Systems, Sydney, Australia, May 2001, vol. II, pp. 345-348.

[22] Paul Barford and Joel Sommers, “Comparing Probe and Router Based Packet-Loss Measurement”, IEEE Internet Computing, Vol. 8, No. 5, September 2004, pp. 50-56.

[23] J Ignacio, Alvarez-Hamelin and Pierre Fraigniaud, “Reducing Packet-Loss by Taking Long Range Dependences into Account”, NETWORKING Proceedings, Athens, Greece, May 2004, pp. 1096 - 1107.

[24] Sharad Jaiswal, Gianluca Iannaccone, Chrostophe Diot, Jim Kurose and Don Towsley, “Inferring TCP Connection Characteristics through Passive Measurements”, IEEE INFOCOM, Hong Kong, March 2004.

[25] Bill Gibson, “Congestion and Delay Hamper the Global Internet”, white paper, Niwot Networks, Inc. April 5, 2000.

[26] W. Richard Steven “TCP/IP Illustrated Volume 1, the Protocols”, October 2002.

[27] H. Sanneck and G. Carle, “A Framework Model for Packet Loss Metrics Based on Loss Runlengths”, SPIE/ACM SIGMM Multimedia Computing and Networking Conference 2000 (MMCN 2000), pages 177-187, San Jose, CA, January 2000.

[28] James Hall, Ian Pratt, Ian Leslie and Andrew Moore, “The Effect of Early Packet Loss on Web Page Download Times,” Passive and Active Measurement Workshop, La Jolla, California USA, April 2003.

APPENDIX A – SNMP Tools

- **NAME**

  Interface Number

  SYNOPSIS

  Perl interface_number.perl [input file name] [number]

  DESCRIPTION

  Interface_number reads the input file. It first calculates the avergae values and then counts the number of interfaces those were experiencing losses simalteniously.

  OPTIONS

  [number] It calculates average values of the input file data using the number specified in the input. If averaging of values is not needed then simply one can be used in input number.

  OUTPUT FORMAT

  Text file with the UNIX time that is obtained from the input SNMP data file and the count of interfcaes experiencing losses simalteniously during that second.

- **NAME**

  Interface Index

  SYNOPSIS

  Perl interface_index.perl [input file name] [number]

  DESCRIPTION

  Interface_index reads the input file. It first calculates the avergae values and then gets the index of interfaces those were experiencing losses simalteniously.

  OPTIONS

[number] It calculates average values of the input file data using the number specified in the input. If averaging of values is not needed then simply one can be used in input number.

OUTPUT FORMAT

Text file with the UNIX time that is obtained from the input SNMP data file and the indexnt of interfcaes experiencing losses simalteniously during that second.

- **NAME**

  average

  SYNOPSIS

  Perl average.perl [input file name] [number]

  DESCRIPTION

  average script reads the input file. Calculate the average values.

  OPTIONS

  [number] It calculates average values of the input file data using the number specified in the input. If averaging of values is not needed then simply one can be used in input number.

  OUTPUT FORMAT

  Text file with the UNIX time that is obtained from the input SNMP data file and the average values for the specified number.

- **NAME**

  total

  SYNOPSIS

  Perl total.perl [input file name] [number]

DESCRIPTION

total script reads the input file. It first calculates the avergae values and then calculate the total values across all the interfaces of the switch.

OPTIONS

[number] It calculates average values of the input file data using the number specified in the input. If averaging of values is not needed then simply one can be used in input number.

OUTPUT FORMAT

Text file with the UNIX time that is obtained from the input SNMP data file and the total values.

- **NAME**

response_impluse

SYNOPSIS

Perl response_impluse.perl [input file name]

DESCRIPTION

response_impluse script reads the input file. Check all the times when the SNMP response is not received.

OUTPUT FORMAT

Text file with the UNIX time when the response from the switch is not received with value 1 (to plot impulse)

APPENDIX B – DAG Log Tools

- **NAME**

  Utilization measure

  SYNOPSIS

  ./utilization_measure [data_source_dir] [list_file]

  DESCRIPTION

  This program reads each packet trace file in the input list one by one. Check each packets destination IP addresses to determine whether it is intra domain traffic (check against POSTECH prefix range) or inter domain traffic. Sum the respective bits of packets received during one second. There are three different versions of this program to calculate the bit counts at 1 second, 1millisecond and 1 microsecond scales respectively.

  OPTIONS

  [list_file] it is a mandatory field. List of the files provides flexibility of choosing the time window on which the program needs to be executed.

  OUTPUT FORMAT

  Text file with per second inter domain and intra domain bit counts

- **NAME**

  Protocol measure

  SYNOPSIS

  ./protocol_measure [data_source_dir] [list_file]

  DESCRIPTION

  This program reads each packet trace file in the input list one by one. It checks each packets destination IP addresses to determine whether it is intra domain traffic (check against POSTECH prefix range) or inter domain traffic. Next, check is done identify the

protocol of each packet and sum of the bits is calculated that are received during one second according to different protocols. There are three different versions of this program to calculate the bit counts at 1 second, 1millisecond and 1 microsecond scales respectively.

OPTIONS

[list_file] it is a mandatory field. List of the files provides flexibility of choosing the time window on which the program needs to be executed.

OUTPUT FORMAT

File with per second bit counts for

1) Inter domain: TCP, UDP and Other 2) Intra domain: TCP, UDP and Other

- **NAME**

Flow generator

SYNOPSIS

./flow_generator [data_source_dir] [list_file]

DESCRIPTION

This program reads each packet trace file in the input list one by one. It inserts packets with same 5 tuple (src/dst IP, src/dst port and protocol) in the hash table to generate flows. If no packet of the same flow is observed during timeout period (configurable) then the flow is considered as terminated.

OPTIONS

[list_file] it is a mandatory field. List of the files provides flexibility of choosing the time window on which the program needs to be executed.

OUTPUT FORMAT

Binary flow file (binary format flows) and ascii flow file(human readable format flows.)

Each file contains: 5 tuples, flow start time, end time and flow size for each flow.

- **NAME**

  Flow Count

  SYNOPSIS

  ./flow_count

  DESCRIPTION

  This program reads the binary flow file that is present in the same diretory as this script. Then it counts the number of flows per second.

  OUTPUT FORMAT

  Text file with inter/intra domain flow counts per second

- **NAME**

  Flow lifetime and size

  SYNOPSIS

  ./flow_lifetime_size

  DESCRIPTION

  This program reads the binary flow file that is present in the same diretory as this script. Then checks if the flow is inter-domain or intra-domain (using destination IP.) Calculates the flow lifetime using flow start and end time. Further, it gets the flow size that is present in binary flow file.

  OUTPUT FORMAT

  Text file that lists lifetime and size of each inter/intra domain flow.

- **NAME**

  New and terminated Flow

  SYNOPSIS

  ./new_terminated_flow

  DESCRIPTION

  This program reads the binary flow file that is present in the same diretory as this script. It counts all the flows that are initiated in the same second. Further, it counts all the flows that are terminated in the same second. This program also calculates sum of the sizes of new and terminated flows per second.

  OUTPUT FORMAT

  File that list new and terminated flow counts and sum of their sizes per second.

- **NAME**

  New and terminated Flow

  SYNOPSIS

  ./new_terminated_flow

  DESCRIPTION

  This program reads the binary flow file that is present in the same diretory as this script. It counts all the flows that are initiated in the same second. Further, it counts all the flows that are terminated in the same second. This program also calculates sum of the sizes of new and terminated flows per second.

  OUTPUT FORMAT

  File that list new and terminated flow counts and sum of their sizes per second.

- **NAME**

Distinct destination

SYNOPSIS

./distinct_destination

DESCRIPTION

This program reads the binary flow file that is present in the same diretory as this script. It addes each distinct source IP in a hash table. Then for each source IP adds the distinct destinations to which it connects in a linked list. Finally, it count the number of destinations to which each source connect by counting the length of linked list.

OUTPUT FORMAT

File with count of distinct destinations to which each source IP connects.

- **NAME**

  top_n_flow

  SYNOPSIS

  ./ top_n_flow

  DESCRIPTION

  This program reads the binary flow file that is present in the same diretory as this script. It checks sizes of all the flows and select the flows whose size fall in the specified range.

  OUTPUT FORMAT

  Binary flow file (binary format flows) and ascii flow file(human readable format flows.) Each File contains: 5 tuples, flow start/end time and size for each selected flow.

- **NAME**

  Flow_throughput

  SYNOPSIS

  ./ flow_throughput [data_source_dir] [list_file] [flow_index]

DESCRIPTION

This program reads the specified flow from the binary flow file that is present in the same diretory as this script. Then go through DAG log files that are present in the data_source_directory. It finds the same flow in the DAG log files and counts the packets per second and bits per second for the flow. There are three versions of this program to get the counts at 1 second, 1millisecond and 1 microsecond scales respectively.

OPTIONS

[list_file] it is a mandatory field. List of the files provides flexibility of choosing the time window on which the program needs to be executed.

[flow_index] flow index is also a mandatory field. It allows us to select any flow and generate data rate for it.

OUTPUT FORMAT

File with pps and bps of the selected flow

- **NAME**

  Run length magnitude and loss rate

  SYNOPSIS

  ./run_length_magnitude [data_source_dir] [list_file] [flow_index]

  DESCRIPTION

  This program reads the specified flow from the binary flow file that is present in the same diretory as this script. Then go through DAG log files that are present in the data_source_directory. It finds the same flow in the DAG log files and checks their sequence number. When a same sequence number packet is retransmitted it marks the beginning and end of the run-length and then calculates magnitude of each run-length. It

assigns magnitude of run-length to start time of run-length. Further, it calculates the interval between the two run-lengths in microseconds to calculate the packet loss rate.

OPTIONS

[list_file] it is a mandatory field. List of the files provides flexibility of choosing the time window on which the program needs to be executed.

[flow_index] flow index is also a mandatory field. It allows us to select any flow and generate data rate for it.

OUTPUT FORMAT

File that list run-length magnitudes and interval between two run-lengths in microseconds.

- **NAME**

broadcast packets

SYNOPSIS

./ broadcast_ packets [data_source_dir] [list_file]

DESCRIPTION

It scans each file specified in the list one by one. Next, it checks the source and destination address of the packets (against 255) to determine if they are broadcast packets. If broadcast packets are found count their number for each second.

OPTIONS

[list_file] it is a mandatory field. List of the files provides flexibility of choosing the time window on which the program needs to be executed.

OUTPUT FORMAT

File that list per second count of IP level broadcast packets.

- **NAME**

  non-IP packets

  SYNOPSIS

  ./ nonip_ packet [data_source_dir] [list_file]

  DESCRIPTION

  It scans each file specified in the list one by one. It Checks if the packet is non-IP packet (Ethernet type != 0x0800). If non-IP packets are found increment their count. Next, it verifies if the non-IP packet is an ARP packet (Ethernet type == 0x0806). If it's an ARP packet then check if it is request packet or reply packet. Then separately counts the request and reply packets received per second.

  OPTIONS

  [list_file] it is a mandatory field. List of the files provides flexibility of choosing the time window on which the program needs to be executed.

  OUTPUT FORMAT

  File that list per second count of non-IP packets, total ARP packets, ARP request and ARP reply packets.

Resume

Name: Deepali Agrawal

Date of Birth: 1979-10-19

Place of Birth: Yeotmal, India

Address: 1-503, OKLIM Apartment, Majeon-Dong, Geoje-si,

Gyeongsangnam-do, KOREA 656-714

Education

- Bachelor of Electronics and Telecommunication Engineering, 1997.5 – 2001.5

Electronics and Telecommunication Department, Pune University, India

- Master of Computer Science, 2004.7 – 2006.2

Division of Electrical and Computer Engineering, POSTECH, South Korea

Job Experience

1. POSTECH Information Research Laboratory (PIRL), KOREA

   Designation: Full time Researcher

   (Distributed Processing & Network Management Lab)

   Duration: Nov 2003 ~ Sep 2004

2. INFOSYS Technologies Ltd., INDIA

   Designation: Software Engineer

   Duration: May 2002 ~ May 2003

3. KOPERKAR Infocraft Ltd**.,** INDIA

   Designation: Software Programmer

   Duration: (2001-5 to 2002-5 full time) & (1997-5 to 2000-12 Part time)

♦ Conference Papers

- Seung-Hwa Chung, Deepali Agrawal, Myung-Sup Kim, James W. Hong, and Kihong Park, "Analysis of Bursty Packet Loss Characteristics on Underutilized Links Using SNMP", 2004 E2EMON, San Diego, California, USA, October 3, 2004.
- Seung-Hwa Chung, Young J. Won, Deepali Agrawal, Seong-Cheol Hong, James W. Hong, Hong-Taek Ju and Kihong Park, " Detection and Analysis of Packet Loss on Underutilized Enterprise Networks", E2EMON, Nice, France, May 2005.

♦ Projects

- Cause Analysis of Packet Loss in Underutilized Enterprise Network Links executed in collaboration with Purdue University, USA.

Acknowledgements

First, I will want to thank God for showering his blessings on me and for always giving me strength to work hard. Then I would like to Prof. Hong for giving me a chance to join DPNM lab first as a researcher and then as master student. He is a brilliant, supportive and extremely understanding and kind professor. He always helped me to solve my all kind of problems and guided me in a right direction. I really thank him for everything. He is one of my ideal professors and will always remain so.

This work was carried out in collaboration with the Purdue University, USA. Prof. Park from Purdue was guiding me in this work. He is a source of knowledge and brilliant ideas. I have learnt so many things from him. He has given really a good direction to this project and I am really glad that I got a chance to work with him. I would also like to thank Prof. Ju who helped me getting new analysis ideas.

I am very thankful to all the professors I took courses with and specially Prof. Young-Joo Suh and Prof. Jong Kim who always solved all my doubts and provided all the help I needed to do to well my course work.

I would like to thank my husband who always stood by me and believed in me even when I could not. He always encouraged me to work harder to achieve my goals. He always cheered me up when I was nervous. Thanks for always being there for me.

I thank to youngjoon who helped in my project setup and he was always there whenever I wanted to have some discussion. I also feel thankful to all lab members who accepted me so well and made my stay in Pohang pleasant.

Last but not the least I will like to thank my parents and family members who always inspired me to do right things and supported me in all my decisions.