Understanding the Network Traffic Constraints for Deep Packet Inspection by Passive Measurement

Jun Liu, Chao Zheng, Li Guo, Xueli Liu, Qiuwen Lu
Institute of Information Engineering, Chinese Academy of Sciences
School of Cyber Security, University of Chinese Academy of Sciences
Beijing, China
{liujun9054, zhengchao, guoli, liuxueli, luqiuwen}@iie.ac.cn

Abstract—**The Deep Packet Inspection (DPI) system examines each captured network packet to find malicious events. The quality of network traffic set an upper limit of DPI system's functionality, such as traffic integrity and asymmetric routing. To better understand these constraints on a DPI system, we setup a series of indicators to quantify these factors, these indicators can be classified as basic information (e.g. average packet sizes), link stability (e.g. out of order packets number), connection integrity (e.g. missing packets number) and asymmetric routing (e.g. one-way flow). There are two challenges in measuring these indicators on real network traffic. The first is how to measure these indicators on high-speed traffic with limited resources. The second is how to track TCP flows across multiple inspection points. We tackle this problem with a scalable passive measurement system, which adopts fast packet I/O technique to capture network traffic, and Spark to process the collected data. To prove its practicability, we deploy the system in a carrier grade network that has six data centers. We have found that 1) over 90% of TCP SYN packets have no subsequent data packet, 2) over 90% of TCP flows are asymmetric, unordered or retransmitted, and 3) over 80% of TCP flow's round trip time are less than 400ms.**

Keywords-DPI; Asymmetric traffic; Passive Measurements; Spark; DPDK

I. Introduction

Recently, a trend towards cyber-attacks that actually try to sabotage the cyberspace have arisen. The Deep Packet Inspection (DPI) systems have widely deployed for defending these attacks, including Network Intrusion Detection Systems (NIDS), Data Leakage Prevention Systems (DLP), etc. These systems examine each captured packet for searching malicious activities, such as virus, intrusions and non-compliance. Most classical DPI systems as Snort [1], BRO [2] and Suricata [3] have the ability to reassemble multiple packets to a data stream (i.e. TCP flows), which requires them to do stateful packet processing. Thus, the quality of network traffic is important to these systems to make a correct decision.

In this paper, we investigated the impact of network traffic quality on the performance of the DPI systems, as well as the measurement of traffic quality. For example, as per-connection data inspection needs to buffer unordered packets, their number has a dramatic impact on memory consumption. In general, the quality comes with three reasons:

- The implementation of server and client side network stacks, e.g. the initial TCP window size.
- The network topology of inspection point, e.g. asymmetric routing.
- Incomplete capture, e.g. misconfiguration of port mirroring on the switch, insufficient performance of packet I/O.

In past decades, researchers have proposed various traffic measurement approaches to determine and evaluate their impact. For example, some researchers measure the high-speed network traffic with dedicated hardware [4], [5], and the others propose sampling algorithm to reduce data volume [6]. Besides, there are many traffic tools built to measure the network traffic such as Tcptrace [7], Tstat [8] and Hadoop-based traffic measurement system [9]. However, the evaluation of network traffic quality across multiple inspection points is rarely concerned, and lack of effective approaches to analyze high volume metric data. In summary, following problems have not been thoroughly concerned in previous researches of passive traffic measurement:

1) How to measure the quality of large volume network traffic quality with limited resources?
2) How to measure the network traffic across multiple inspection points? Large enterprises usually have multiple geographically distributed data centers, aggregating these measurement results enables administrators to understand their traffic better.

To tackle above problems, we build a scalable passive measurement system, which combines a DPDK-based fast packet I/O technology [10], and adopts Apache Spark to process the collected data [11]. Our system could measure the fluctuation of traffic quality, which usually suggests topology changes, misconfiguration and attacks. In the course of developing a traffic quality measurement system for multiple data center, we make two research contributions:

1) We propose an accurate approach to track TCP connections across multiple inspection points.
2) We setup a series of indicators to quantify the detectability of network traffic with limited resource.

We evaluated the system by deploying it on 39 servers of an ISP network, which has 6 data centers. The system sampled 400,000 TCP connection's metadata per second.

The rest of this paper is organized as follows: Section 2 briefly introduces background and related work about the topic of traffic measurement and relevant traffic analysis tools. And we introduce the significance of TCP streams attributes and corresponding extraction methods in section 3. Then, we present a software framework for online or offline monitoring and evaluating network traffic in section 4. In section 5, we deploy the above-mentioned system in ISPs network to monitor and analyze network traffic. We conclude our work and discuss future research directions in section 6.

II. RELATED WORK

The topic of traffic measurement and analysis has played an important role on network management and has drawn more attention in recent years. There are many popular projects and traffic tools built to measure and analyze the network traffic.

Network traffic measurement approaches can be classified as active measurement and passive measurement. Paris Traceroute [12] is a powerful network diagnosis tool based on active probing. However, active measurement approach is expensive for using extra bandwidth, and perhaps further change the status of observed network. Tcptrace [7] is a passive measurement tool to analyze packet dump and able to provide information of specific TCP connection. Tstat [8] is a passive analysis tool of TCP/IP traffic able to provide many features statistics such as TCP performance, application classification and VoIP characteristics but cannot be used to analyze packet drops.

Sampling algorithm, dedicated hardware and flow export technology is widely used in traffic measurement. Raspall [6] proposed an efficient byte sampling algorithm to avoid the adverse effect of packet size. However, packet sampling algorithm not applies to assess the quality of network traffic. Sangchoolie, B [5] developed Flowstat to detect all kinds of incomplete captures with DAG card. NetFlow [4] is a Cisco hardware-based flow measurement technology based on IP Flow Information Export (IPFIX) standard and efficiently provides a key set of service for IP applications. Flow traffic export technology [13] effectively reduces the need for storage resources and maintains accuracy at a little sacrifice of some packet information. But the disadvantage of dedicated hardware for handling packets is the increased cost of network equipment.

In general, the majority network traffic measurement and analysis tools run on single server [8] and these tools are not capable of measuring a large amount of traffic captured at high-speed links in a scalable manner. To keep traces of network packets, Lee, Y [9] designed a Hadoop-based packet trace processing tool, which needs more storage resources for storing packets in HDFS.

III. TCP TRAFFIC METRICS AND METHODS

A. *The definition of a TCP Flow*

According to RFC 5470 [13], the definition of IP flow was described by the IP Flow Information Export (IPFIX):" a flow is defined as a set of IP packets passing an observation point in the network during a certain time interval. All packets belonging to a particular flow have a set of common properties". In general, flow defined by the 5-tuple of source and destination IP address and port number and transport layer protocol number [13]. Due to the protocol of NAT and Cloud technology is widely used in network, there is a phenomenon that port reuses and tuple4 reuses in real network. For TCP stream, 5-tuples {src_ip, src_port, dst_ip, dst_port, proto_id} is not the unique properties for defining a flow. So, we redefine a TCP flow by 6-tuple {init_seq_c2s, src_ip, src_port, dst_ip, dst_port, protocol_id}.

B. *Main Attributes and Methods*

TCP traffic is a larger composition of Internet traffic. Measurement tools should drown more attention to analyze TCP traffic and detect TCP behaviors. For the purpose of quantifying the quality of TCP traffic, we extract attributes of TCP stream, as illustrated in table 1. In order to better express it, we define a TCP flow with complete three-way handshake packets as handshake completeness flow. The attribute of FlowType is used for distinguishing the type of TCP flow such as two-way flow (csc), client-to-server flow (c2s), server-to-client flow (s2c). And if the upstream of a flow and the downstream of a flow are correlated, the type of flow is c2s&s2c, and the attribute of observer_point_num is equal to two.

1) Round Trip Time Using Two-way Traffic Flow

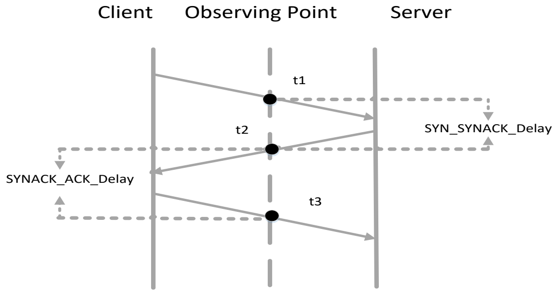

Fig.1 Overview of TCP connection step

Customers' demand for better quality of service (QoS) and quality of experience (QoE) is increasing [9]. Network latency is a very important factor for user behaviors in a network. Measuring and monitoring the Round Trip Time (RTT) can provide an indicator of network performance for network management. The TCP setup is a composite of a three-way handshake with SYN, SYN+ACK and ACK packets [14], as illustrated in Fig.1. Since the measurement point is in the middle of the client and server, we will use the follow formula:

$$RTT=SYN\_SYNACK\_Delay+SYNACK\_ACK\_Delay \qquad (1)$$

for estimating the RTT.

2) Time To Live

Time-to-live (TTL) is a mechanism for limiting the lifetime of each packet in network. The TTL value of each packet in one flow may be different because of routing police, asymmetric routing or other reasons. However, there is a new way of censorship circumvention to avoid inspecting packet. The TTL in the insertion packet, is manipulated to prevent the acceptance of the first injected SYN by the server a packet with a lower TTL value would never reach the intended server [15]. Although an attacker can forge ant field in the IP header, he cannot falsify the number of hops an IP packet takes to reach its destination, Jin, C [16] builds an accurate IP to hop-count mapping table to detect and discard spoofed IP packets. So it is also essential for monitoring and analyzing the situation of TTL value. For the processing performance of network traffic, we only count the initial TTL value, the maximum and minimum TTL value. And if a TCP flow whose existing TTL value is different among packets, we define the flow as inconsistent TTL flow.

TABLE.1 TCP STREAM INFORMATION

Series	Metrics	Total # of metrics
Basic information	ISP, Area addr, Data Center, inspection point, Init_sequence_C2S, Client IP, Client port, Server IP, Server port, Create time, Total number of packets of C2S and S2C, Total bytes of C2S and S2C, Number of packet with payload of C2S and S2C, Flow Duration, Flow Close Reason	18
Link Stability	Delay of SYN_SYNACK and SYNACK_ACK, MIN, MAX, AVG, Standard Deviation of RTT, Number of duplicate packets of C2S and S2C, Flow ratio of duplicate packets and packet ratio of duplicate packets, Number of out-of-order packets of C2S and S2C, Flow ratio of out-of-order packets and packet ratio of out-of-order packets	14
Connection Integrity	Three-way handshake completeness, Number of missing packets, Bytes of missing packets, Flow ratio of missing packet	5
Asymmetric routing	Flow Type Inconsistent TTL Num of inspection point, data center, area addr and ISP when the asymmetric traffic correlated	6

3) *Missing packet, Out-of-order Packet and Duplicate packet*

The quality of captured traffic plays an important role in decision made by systems like intrusion detection/prevention systems and firewalls [5]. TCP is a connection-oriented, reliable transmission protocol. We measure the condition of missing packets, out-of-order packets and duplicate packets to represent the characteristics of TCP connection based on DPDK and adequate buffers.

a) Duplicate packet: if the sequence number of the received packet has already been observed before [17];

b) Out-of-order packet: if the sequence number of the received packet is not the expected [17];

c) Missing packet: the packet is not effectively captured due to insufficient performance of packet I/O or asymmetric routing.

For the processing performance of network traffic, we compare the current sequence number and previous sequence number of a packet for computing the number of duplicate packets and out-of-order packets. And we set adequate buffers to cache current several network packets for computing missing packets.

4) *One-way Traffic and Asymmetric Traffic*

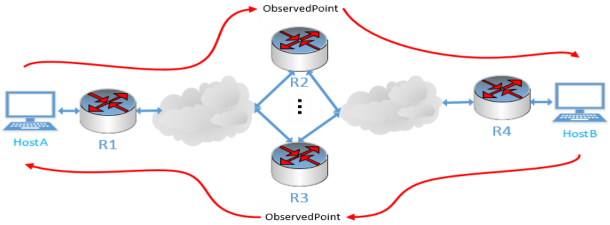

Fig.2 Overview of asymmetric traffic

We define a flow that only observed packets of forward or reverse direction on the physical links as a one-way flow. One-way flow is asymmetric traffic. In terms of experiments, we find that the one-way traffic is a large component of Internet traffic. Glatz, E [18] classifies the one-way traffic into seven types, including Malicious Scanning, Benign P2P, Service Unreachable, Suspect Benign, Bogon, Backscatter and other. From the perspective of DPI, the asymmetric traffic poses security audit challenge such as the upstream and the downstream of a flow passes from different links can result in the vulnerability of security audit due to the lack of interaction of flow state information kept in firewall, as illustrated in Fig.2. Furthermore, asymmetric traffic poses challenges for flow identification, malicious/anomaly detection, etc.

5) *Origin-Destination Flow Measurement*

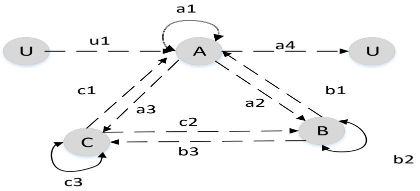

Fig.3 : bidirectional arrow represents two-way flow; dottedarrow represents one-way flow; the weight represents the degree of relationship between nodes

Origin-destination (OD) flow measurement can research the relationship between two network elements, which is important to network management task such as capacity planning, traffic engineering, anomaly detection, and network reliability analysis. We define a flow whose upstream passes R2 and downstream passes R3 as an origin- destination (OD) flow of the two network elements of R2 and R3, as shown in Fig.3. Meanwhile, we define a matrix to represent the degree of the relationship between network elements, as illustrated in formula (2). Our goal is to measure the relationship between nodes in terms of ratio of flow. The weight in the figure represents the traffic relationship between nodes. As illustrated in Fig.3, we define a matrix to represent the relationship between A, B, C and U (unknown network element is not our measurement point):

$$RelationshipSet = \begin{bmatrix} a1 & a2 & a3 & a4 \\ b1 & b2 & b3 & b4 \\ c1 & c2 & c3 & c4 \\ u1 & u2 & u3 & 0 \end{bmatrix} \qquad (2)$$

$$\text{Rij} = \frac{|<Ni,Nj>|}{|Ni|} \qquad (3)$$

Where, $|<Ni,Nj>|$ represent the number of flow that upstream of a flow passes the network element Ni and downstream of a flow passes the network element Nj. And |Ni| indicates the number of all flow that passes network element Ni. The formula (3) represents the relationship between Ni and Nj. For example, a1 represents the ratio of two-way flow, and a2 represents the ratio of one-way flow whose upstream passes the network element A and downstream passes the network element B, as illustrated in formula (2) and Fig.3. There is a key difficulty of measuring the relationship of interconnected among network elements is accurate asymmetric traffic correlation. Network traffic dynamic transfer in space and time, it is very hard to associate upstream and downstream. For conveniently matching C2S directional flow and S2C directional flow, we treat 6-tuple {init_seq_c2s, src_ip, src_port, dst_ip, dst_port, protocol_id} as new 6-tuple {init_seq_c2s, client_ip, client_port, server_ip, server_port, protocol_id} with the port number comparison.

IV. TRAFFIC MEASUREMENT AND ANALYSIS SYSTEM WITH SPARK

A. *Measurement scenario*

As shown in Fig.4, there are 39 measurement nodes of 6 data centers and more than 3Gbps of bandwidth per node. Many passive measurement nodes generate large data volumes. One key difficulty is a demand for efficient technology for processing large-scale dynamic data stream in real time. DPDK is a high performance packet I/O framework. Apache Spark is designed for real time analysis of high velocity live data streams [11]. In order to meet the high bandwidth and multiple inspection points measurement requirements, we designed and implemented the scalable passive network traffic measurement and analysis system, as shown in Fig.5.

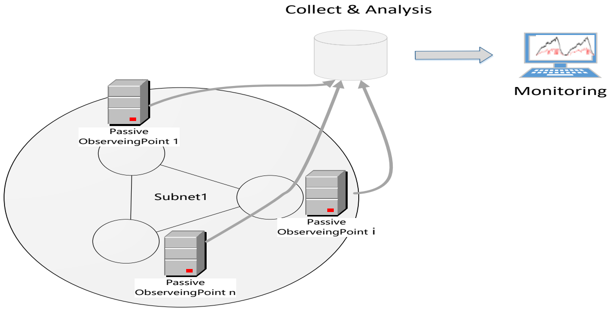
Collect & Analysis
Monitoring
Passive ObserveingPoint 1
Subnet1
Passive ObserveingPoint i
Passive ObserveingPoint n

Fig.4 Overview of Traffic Measurement System deployment in ISPs network

B. *Traffic Measurement and Analysis Framework*

As shown in Fig.5, our system consists of four modules. Packet Processing Module is a DPDK-based high-performance packet processing system, which has four functions that are packet capture, flow sampling, flow information extract and flow information in JSON format sent to Kafka. Kafka is a scale-out and high throughput distributed streaming platform [19]. Big- Data Analysis Module is a Spark-based memory computing system, which analyzes network traffic in real time and stores the result in database or HDFS. And Data-Visualization Module is used for visualizing the analyzed results into a chart in real time.

1)TCP Traffic Sampling

Flows provide an aggregated view of network traffic by grouping streams of packets [20]. Multiple passive observed points can generate a large amount of traffic data. To effectively reduce the network traffic data and keep the characteristics of the network traffic data information and maintain data consistency on multiple nodes, we propose a proper flow sampling algorithm that sample client IP or server IP of new 6-tuple {init_seq_c2s, client_ip, client_port, server_ip, server_port, protocol_id}. For example, as shown in Fig.2, if the upstream of a flow that passes the router R2 has been sampled, the downstream must be sampled at router R4 for keeping data consistency.

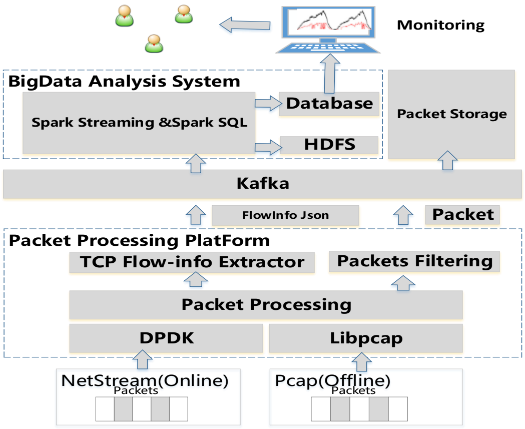

Fig.5 Compositions of the traffic measurement and analysis architecture

2)Flow Information Extractor

Before performing traffic analysis in spark, we need to extract TCP stream information such as table 1. To gather all network packets of a flow with a set of common properties by using traffic processing technology, we defined a large flow table and used multi-thread technology to process traffic. If the captured packet is SYN or SYN+ACK packet and the packet is sampled successfully, we will build a record in the flow table and assign a thread to process subsequent network packets belong to the flow. When captured a RESET or FIN packet or timeout signal of the flow record in the flow table, we transfer the flow information in JSON format to Kafka and remove the corresponding record in the flow table. The process is presented in Fig.6.

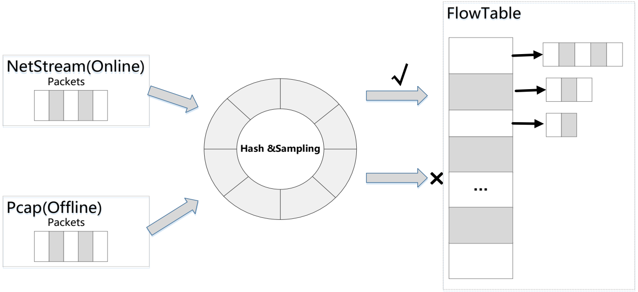

Fig.6 Overview of Flow Information Extractor

3)Real-Time Big-data Analysis System

We use Spark Streaming technology, Spark SQL technology and other complex analysis algorithm to analyze the data that ingested from Kafka. In this module, we exploit windowed computations mechanism for analyzing the data. And the analyzed result can be saved at HDFS or database. In the end, Data-Visualization Module is used for visualizing the analyzed results into a chart in real time.

V. REAL DATA STUDY

We deploy our Packet Processing Platform in 6 data centers of ISPs network for capturing sampled TCP stream in real time and will generate 400,000 TCP connection's metadata per second. And Spark analysis procedure runs on a spark cluster of 23 servers and occupies 32 cores in total and 2GB memory per node.

In this paper, we measure the characteristics of TCP streams such as RTT, out-of-order packets, TTL, etc. Then we analyze the condition of asymmetry traffic and the relationship between 6 data centers. And we promise not to abuse the privacy of network traffic.

A. *Impact of the Interval size*

Since bandwidth is relatively stable, we can use D(t) to represent the data volume and P(t) represent the performance of analysis procedure when processing the data volume of D(t). In experiment, we find that the performance of processing data is proportional to the amount of data. When the spark cluster resources limited to 32 cores in total and 2GB memory per node, we observe that 90 seconds is expected interval size, as is present in Fig.7. If the data is consumed slowly, the data will be blocked in Kafka and dropped. And the more resources are needed if the window length is large.

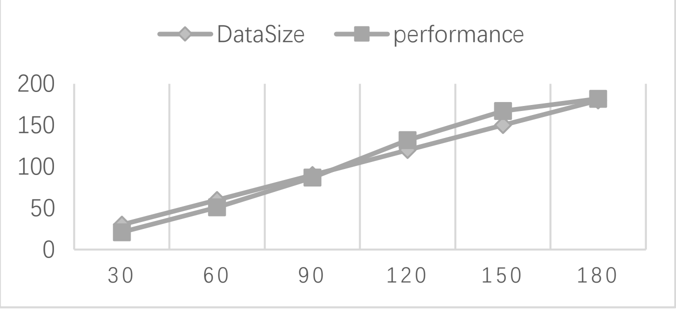

Fig.7 Data Volume and Performance

B. *Anomalies/Noise data*

In experiment, we find that there is a large amount of TCP traffic without data and account for more than 70% of client-to-server (C2S) traffic, as illustrated in Fig.8. Furthermore, utilizing Spark FPGrowth algorithm [11] to count frequent item, we find that there are a few IP address and network port that appear higher frequency, as is show in Table.2. It seems like a DDOS attack or malicious scanning have occurred in the network. Here we will temporarily treat one-way traffic without data as noise data. Subsequently, we measure and analyze network traffic after excluding the noise data.

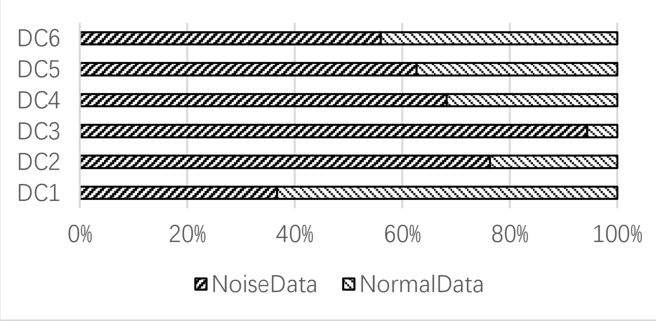

Fig.8 Noise Data Ratio

TABLE.2 IP ADDRESS AND NETWORK PORT FREQUENT STATISTICS

Src_ip	Src_port	Dst_ip	Dst_port	Datasize	Frequency
*	49400	Server1	22	0	607575
*	*	Server2	80	0	530035
Server3	9500	*	*	0	860817
Server4	7511	*	*	0	506809

C. *Missing packet, Out-of-order Packet and Duplicate packet*

For characterizing the TCP connection, we measure the condition of missing packet, out-of-order packet and duplicate packet. These features of TCP connection have an impact on the decision of the IDS. As shown in Fig.9, on average, less than 7% of TCP flow has loss packet in six data centers. As is show in Fig.10, more than 90% of TCP flow out of sequence. Moreover, the average ratio of out-of-order packet is less than 20%. It suggests that the phenomenon of out-of-order is pervasive in ISPs network but the condition of out-of-order packet is not very serious. And as shown in Fig.11, we find that the average flow ratio of c2s directional retransmission is higher than of s2c directional retransmission.

$$\text{DifferentTypeFlowRatio} = \frac{Num\ of\ Different\ Type\ Flow}{Num\ of\ All\ Flow} \quad (4)$$

$$\text{OutOfOrderPacketRatio} = \frac{Num\ of\ Out\ Of\ Order\ Packet\ in\ a\ flow}{Num\ of\ All\ Packet\ in\ a\ flow} \quad (5)$$

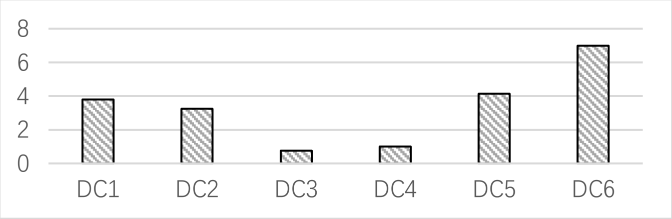

Fig.9 Missing packet Flow Ratio

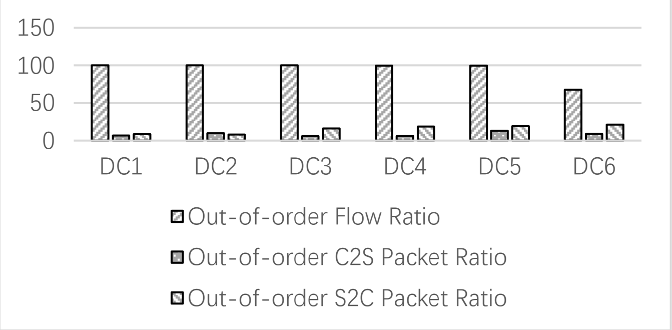

Fig.10 Out of order packet Flow Ratio and Packet Ratio

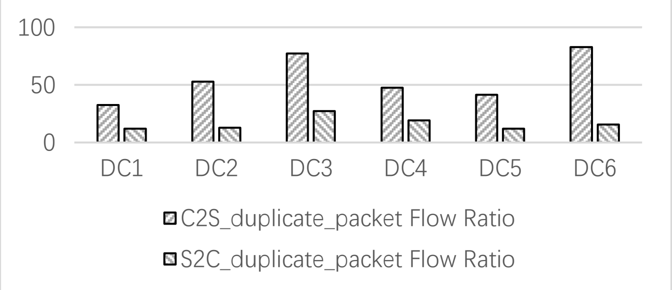

Fig.11 Duplicate Packet Flow Ratio

D. *Round-Trip Time(RTT)*

The value of RTT provides an indicator of network performance. We analyze the value of RTT based on different data centers, as shown in Fig.12. We found more than 80% of latency are 400ms and there are no dramatically different among 6 data centers.

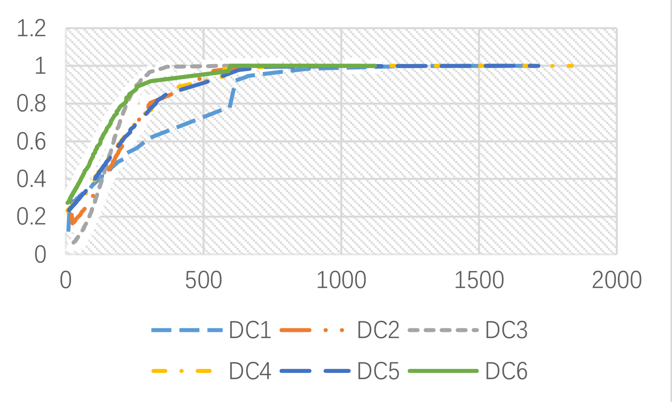

Fig.12Cumulative Probability of RTT

E. *Time-To-Live (TTL)*

As illustrated in Fig.13, we observe that less than 0.14% of inconsistent TTL flow and more than 98% of TCP flow whose TTL values are same. The phenomenon of inconsistent TTL caused by asymmetric routing or malicious activities. But, it is hard to distinguish which reason causes the phenomenon. Moreover, the DPI system rarely concern the phenomenon of inconsistent TTL. So, we monitor the indicator to provide suggests for network security management. For example, we can specifically secure inspects these a small amount of network traffic and these nodes with higher ratio.

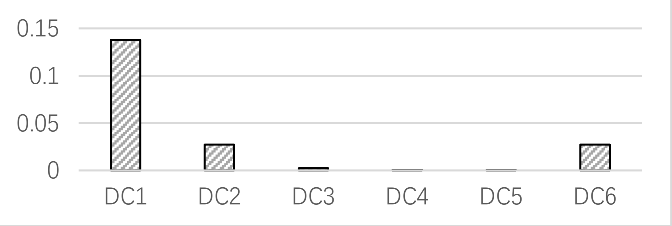

Fig.13 Inconsistent TTL Flow Ratio

F. *One-way Flow and Asymmetric Traffic*

Traffic asymmetry is widespread throughout networks and is an inevitable phenomenon due to Equal Cost Multi-Path routing, load balance, route police, malicious scanning, DDOS and etc. As shown in Fig.14, the one-way traffic accounts for more than 70%, but the data center three (DC3) only accounts for 26.3%. It suggests that asymmetric routing is pervasive phenomenon throughout network. The phenomenon that may be caused by route police or malicious events deserves our further study to explain this reason. It may be very useful for ISPs network management.

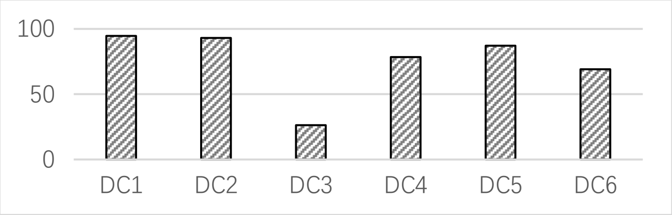

Fig.14 One-way Traffic Flow Ratio

G. *Origin-Destination Flow Measurement*

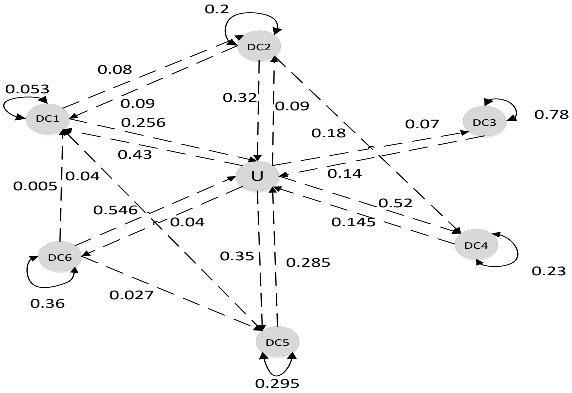

Fig.15 The Relationship Among Data Center

We propose the measurement method that can depict the relationship among ISPs, data centers and inspect nodes. In experiment, we measure the relationship among six data centers, as shown in Fig.15.The weight in the figure represents the relationship between data centers. For example, 8% of upstream passes DC1 and downstream passes DC2, and 9% of upstream passes DC2 and downstream passes DC1. Due to asymmetric traffic correlation is hard and limited the number of inspects nodes, we observe that there is partial traffic be correlated. As long as deploying more measurement nodes, we will obtain accuracy relationship among data centers. This actual relationship is very useful for network management such as capacity planning, anomaly detection, network reliability analysis, etc.

VI. CONCLUSION

In this paper, we propose a scalable passive traffic measurement system combines DPDK-based traffic processing technology and Spark-based streaming computing technology to overcome the difficulty of limited computing and storage resources for network traffic measurement and analysis. The system can process 400,000 TCP connection's metadata per second in real time and visualize the analyzed results into a chart for monitoring the fluctuation of a series of indicators. These indicators can be classified as basic information, link stability, connection integrity and asymmetric routing.

Through experiments conducted in a carrier grade network that has 6 data centers, we have found that 1) over 90% of TCP SYN packets have no subsequent data packet, 2) over 90% of TCP flows are asymmetric, unordered or retransmitted, and 3) over 80% of TCP flow's round trip time are less than 400ms. Moreover, we have measured OD flow to represent the relationship among 6 data centers and asymmetric traffic. We believe that the system can provide useful help for network management and enhance the understanding of network traffic constraints for DPI. Furthermore, we can monitor the tendency and fluctuation of a series of indicators. It can help in monitoring inspects nodes which have an abnormal behavior such as delay, packets drop, malicious scanning, etc. In fact, we indeed found that there is traffic anomaly in data center and suggested ISPs correcting problems in real-time.

In future work, we plan to use machine learning to further analyze network traffic based on this system. And we hope that the Spark-based passive network traffic measurement and analysis system designed and implemented practically by us can be useful for operators and researchers.

REFERENCES

[1] M. Roesch, "Snort, intrusion detection system," Jul. 2008. [Online]. Available: http://www.snort.org

[2] V. Paxson, ″Bro: a system for detecting network intruders in real-time,″ Computer Network,vol. 31, no. 23–24, pp. 2435–2463, 1999

[3] Suricata,https://suricata-ids.org/

[4] NetFlow, http://www.cisco.com/c/en/us/products/ios-nx-os-software/ios-netflow/index.html

[5] Sangchoolie, B., Nasab, M. R., Olovsson, T., & John, W. (2012). Assessing the quality of packet-level traces collected on internet backbone links. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 7617 LNCS, 184–198.

[6] Raspall F. Efficient packet sampling for accurate traffic measurements. Computer Networks, 2012,56(6):1667−1684.

[7] Tcptrace, http://www.tcptrace.org

[8] Tstat ,http://tstat.tlc.polito.it/

[9] Lee, Y., Kang, W., & Lee, Y. (2011). A hadoop-based packet trace processing tool. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 6613 LNCS, 51–63.

[10] DPDK, http://www.dpdk.org/

[11] Spark, http://spark.apache.org

[12] Paris-traceroute, www.paris-traceroute.net

[13] Sadasivan, G., Brownlee, N., Claise, B., Quittek, J.: Architecture for IP Flow In- formation Export. RFC 5470 (Informational) (2009)

[14] Høiland-Jørgensen, T., Ahlgren, B., Hurtig, P., &Brunstrom, A. (2016). Measuring Latency Variation in the Internet. Proceedings of the 12th International on Conference on Emerging Networking EXperiments and Technologies -CoNEXT '16, 473–480.

[15] Your State is Not Mine: A Closer Look at Evading Stateful Internet Censorship

[16] Jin, C., Wang, H., & Shin, K. G. (2003). Hop-Count Filtering: An Effective Defense Against Spoofed DDoS Traffic. Proceedings of the 10th ACM Conference on Computer and Communication Security - CCS '03, 30.

[17] Mellia, M., Meo, M., Muscariello, L., Elettronica, D., & Torino, P. (n.d.). TCP Anomalies: identification and analysis.

[18] Glatz, E., &Dimitropoulos, X. (2012). Classifying internet one-way traffic. ACM SIGMETRICS Performance Evaluation Review, 40(1), 417.

[19] Kafka, http://kafka.apache.org/

[20] Hofstede, R., Drago, I., Sperotto, A., Sadre, R., &Pras, A. (2013). Measurement artifacts in NetFlow data. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 7799 LNCS, 1–10.