Research on Errors of Utilized Bandwidth Measured by NetFlow

Haiting Zhu[1], Xiaoguo Zhang[1,2], Wei Ding[1]
[1]School of Computer Science and Engineering, Southeast University, Nanjing 211189, China
[2]Electronic Information Engineering School, Henan University of Science and Technology, Luoyang 471003, China
E-mail: {htzhu, xgzhang, wding}@njnet.edu.cn

***Abstract*—NetFlow is a popular technology in network management nowadays. It supports administrators to get various performance metrics. But metric values estimated from NetFlow records may inaccurate in some circumstance. This paper aims at studying the errors of utilized bandwidth measured by NetFlow. At first, packet streams with a specific format were collected from the backbone by passive measurement and these packet streams are called IP Traces that are used as our experimental data. Then, by using IP Traces, NetFlow records were produced according to NetFlow principles using different sampling ratios and the utilized bandwidths are calculated by NetFlow records at different time granularities. These utilized bandwidths called measured values will be compared to the true values that are directly calculated using IP Traces. At last, the relative errors of utilized bandwidth are presented based on measured values and true values. The experimental results show that bigger time granularity results in smaller error of utilized bandwidth measured by NetFlow, and smaller sampling ratio produces smaller errors. However, if the two factors are considered separately, the error is still large. Only when time granularity is big enough and the sampling ratio is small enough the error could be acceptable. If we want to acquire the satisfying precision, time granularity and sampling ratio need to be set properly. The study results of this paper can provide some guidance and reference for estimating utilized bandwidth with NetFlow.**

Keywords-measurement error; utilized bandwidth; NetFlow; IP Trace

I. INTRODUCTION

With the development of the network techniques, the network scale becomes enormous and the structure is more complex. In order to ensure that network runs well, it is necessary to manage network efficiently. The network utilized bandwidth reflects the running status of network; it is the key data to judge whether network was running normally or not. It also can be used to monitor the network traffic and to analyze network usage and performance by network administrator, so as to find the network bottleneck as early as possible, then to adjust network routing and to achieve the ultimate purpose that network runs efficiently, steadily and reliably[1].

At present, we usually use three methods to analyze the network traffic:

(1) Making use of the generic network monitoring software for link utilization rate, such as Multi Router Traffic Grapher (MRTG). Simple Network Management Protocol (SNMP) can be used to monitor traffic on routers and usually make statistic analysis in the key links and interconnected points. However, SNMP can only collect the packets and bytes that pass in and out the network ports and cannot provide fine-grained traffic statistic analysis such as traffic volumes of different application types.

(2) Deploying probes on the important network nodes, monitoring the ports, collecting the network traffic and upper layer traffic. This method works at packet level and can provide high precision traffic volumes of network application types. However, this method collects extremely large data volume and relies on high-performance hardware, so it hardly uses in the network backbone.

(3) Using NetFlow technology to measure and analyze the detailed behavior pattern of IP network traffic and to provide detailed statistics data [2]. This method works at flow level and can provide high precision traffic volumes of network application types. Moreover, this method just need collect less data volume than deploying the probes.

Thus it can be seen, measuring and analyzing network traffic using NetFlow technology is a more eclectic and effective method. At present, FlowScan is used widely at home and abroad, it is a tool to measure information of flows based on NetFlow. It increments traffic counters at the time when the flows are exported. More precisely, it records the values of these counters with the time-stamp corresponding to the five-minute interval in which cflowd wrote the flow to the raw flow file [3]. This method is efficient and cost-effective. When a flow reaches to the collector, what needs to be done is only to update the traffic value of one time granularity.

However, this traffic measurement method has an important weakness [4]. The statistic data by NetFlow, such as traffic bytes and packets, is very rough relative to SNMP. Because there are two situations that may cause errors while using NetFlow records:

(1) Flow duration time might be longer than measurement time granularity. For example, there are long flows in flow records.

(2) Even if flow duration time is shorter than measurement time granularity, such a situation still exists—some flow records begin in one time granularity and end in another time granularity.

It is obvious that this is an eclectic between precision and simplicity. In some cases, long flow will cause the traffic to

increase sharply in some time granularity and the precision is affected enormously.

NetFlow might consume a large quantity of resource of equipment especially at backbone routers. It supports exporting NetFlow record under certain sampling ratio. However, sampling ratio is worth studying. If sampling ratio is big, the pressure of the equipment is big and might affect the functions of the equipment. Little sample ratio could ease the pressure but the credibility of traffic analysis upon these records might be low [5] [6].

This paper researches on measurement errors of utilized bandwidth at different time granularities and sampling ratios by a large number of experiments. The introduction points out influencing factors of utilized bandwidth measurement using NetFlow mechanism. Section 2 analyzes the working principle of NetFlow. Section 3 introduces the source and structure of our experimental data. Section 4 shows the errors of utilized bandwidth measured using NetFlow based on these data at different time granularities and sampling ratios.

II. NETFLOW

NetFlow is an important network protocol developed by Cisco Systems for collecting IP traffic information. NetFlow has become an industry standard for traffic monitoring. At present, there are two important versions 5 and 9. Although version 9 was accepted by IETF in 2004 and became the IP Flow Information Export standard in RFC3917, version 5 is also applied popularly now.

In NetFlow mechanism, a flow is identified as a unidirectional stream of packets between a given source and destination that are both defined by a network layer IP address and transport layer source and destination port numbers. A network flow has been defined in many ways. The traditional Cisco definition is to use a 7-tuple key, where a flow is defined as a unidirectional sequence of packets all sharing all of the following 7 values [2]:

(1) Source IP address
(2) Destination IP address
(3) Source port number
(4) Destination port number
(5) Layer 3 protocol type
(6) ToS byte
(7) Input logical interface (ifIndex)

These seven key fields define a unique flow. If a flow has one different field than another flow, then it is considered a new flow. A flow contains other accounting fields (such as the AS number in the NetFlow export Version 5 flow format) that depend on the version record format that you configure for export. Flows are processed in a NetFlow cache. The NetFlow cache is built by processing the first packet of a flow through the standard switching path. A Flow record is maintained within the NetFlow cache for all active flows. The routing device checks the NetFlow cache once per second and expires the flow in the following instances:

(1) The flow cache has become full.
(2) Transport is completed (TCP FIN or RST).
(3) The inactive timer has expired after 15 seconds of traffic inactivity.
(4) The active timer has expired after 30 minutes of traffic activity.

As the cache becomes full a number of heuristics are applied to aggressively age groups of flows simultaneously. TCP connections which have reached the end of byte stream (FIN) or which have been reset (RST) will be expired. Expired flows are grouped together into NetFlow export datagrams for export from the NetFlow-enabled device. On a Cisco routing device, the inactive timer exports a packet with a default setting of 15 seconds of traffic inactivity. You can configure your own time interval for the inactive timer between 10 and 600 seconds. The active timer exports a packet after a default setting of 30 minutes of traffic activity. You can configure your own time interval for the active timer between 1 and 60 minutes [2].

Figure 1 illustrates how flow AT1 expires because the active timer for the flow exceeds the default value of 30 minutes. AT2 is the second flow which expires because the inactive timer exceeds the default value of 15 seconds.

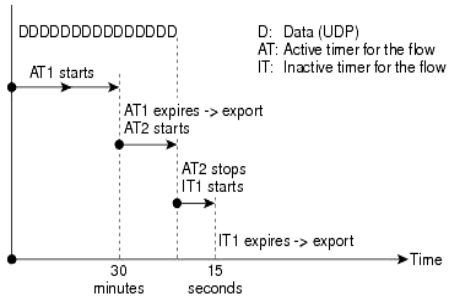

Figure 1. Active and inactive timers

NetFlow export datagrams may consist of up to 30 flow records for version 5 or 9 flow export. NetFlow functionality is configured on a per-interface basis. To configure NetFlow export capabilities, the user simply needs to specify the IP address and application port number of the Cisco NetFlow or third-party Flow-Collector.

III. EXPERIMENTAL DATA

IP packet is the basic unit of data transmission for IP layer. It is also the basic unit of passive measurement. So, data collected by passive measurement is packet streams and these packet streams are usually called IP Trace.

Based on IP Trace from China Education and Research Network (CERNET) backbone, this paper aimed at finding out the potential errors between estimation values of utilized bandwidth from NetFlow records and real network utilized bandwidth values from IP Trace.

IP Trace was passively collected at the border router of the CERNET Jiangsu Province backbone, in which more than 100 colleges and universities are included. The link channel used to be OC48, and it updated to OC192 in Jan. 2006. Each IP Trace is consisted of many Trace files. Trace files are published after anonymization, and about 140 Mb each. Each trace file is consisted of 3084047 IP packet records. The structure of each IP packet record shows in figure 2, and these records are stored in time order.

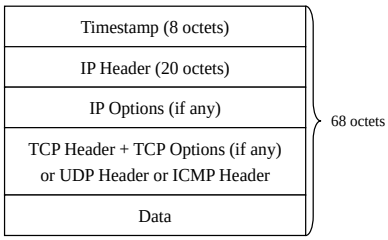

Figure 2. IP trace packet structure

One IP record unit contains 68 bytes with timestamp using 8 bytes and IP headers of each packet using the left 60 bytes. Timestamps use 'struct timeval' in C language to present. If the packet is less than 60 bytes, the left place will be filled with random values.

IP Trace used in this thesis is from 14:00 to 16:00 on Nov.19, 2009, and the size is about 129 GB. This trace is used to calculate the true network utilized bandwidths first. In order to get NetFlow records of this trace, a program was written according to Cisco's NetFlow rules above then NetFlow records of this IP Trace could be produced by the program. Sampling ratios would be configured in this program to generate NetFlow records at different sampling ratios. These are the preparation steps of our study.

IV. Experimental Methods And Results

A. Experimental Methods

There are mainly two factors which could cause the estimation of utilized bandwidths by NetFlow records inaccurate. In this paper, we use flow's end-time to decide which time granularity it belongs to. If one flow's end-time belongs to a time granularity, the traffic of flow would be added into traffic of this time granularity.

1) Different time granularities used to calculate the network utilized bandwidths

Analyses are done with the 12 different time granularities values from 1 second to 2048 seconds. The average relative errors and max relative errors are calculated separately.

2) Different sampling ratio used in NetFlow record generation

Sampling ratio is a configurable parameter in NetFlow support equipments, so different sampling ratios are also used in our analyses. Sampling ratio ranges from 1 to 1/262144; each is half of the former ratio. There are a total of 18 different sampling ratio levels.

All the calculations are all based on the same IP Trace records, the details are:

First, generating NetFlow records from the original IP Traces at different sampling ratios (no sampling means sampling ratio is 1). Then estimating the utilized bandwidths based on each time granularity. At last, compare the estimation values to the true values of utilized bandwidths calculated from IP Traces, and give the maximum relative errors and average relative errors to the real utilized bandwidth values. If the sampling ratio is '1/r', then the estimation values should multiple 'r' first, and then do the comparisons.

TABLE I. Errors On Bytes At Different Time Granularities

Time Granularity (Seconds)	Out		In	
	Maximum Error	Average Error	Maximum Error	Average Error
1	265.9318	0.6260	191.6857	0.5327
2	149.8727	0.5423	102.4550	0.4513
4	79.8854	0.4772	54.6736	0.3874
8	40.4449	0.4281	27.5677	0.3283
16	19.7264	0.3901	13.8433	0.2890
32	9.6066	0.3622	6.9035	0.2597
64	4.5590	0.3356	3.1858	0.2378
128	2.1161	0.3174	1.4541	0.2094
256	0.8750	0.2959	0.6795	0.1922
512	0.5812	0.2405	0.4757	0.1644
1024	0.4924	0.1752	0.3858	0.1356
2048	0.2525	0.1722	0.1998	0.1366

B. Experimental Results

1) Estimation errors at different time granularities without sampling

We get the estimation errors (both maximum estimation errors and average estimation errors) at different time granularities in Table I. The first column is the time granularities in second. The second and third columns are the relative errors in link direction 'out'. The fourth and fifth columns are the relative error in link direction 'in'.

TABLE II. Maximum Errors At Different Sampling Ratios

Time Granularity (Seconds)	Maximum Error on Bytes										
	Sampling Ratio 2	Sampling Ratio 4	Sampling Ratio 8	Sampling Ratio 16	Sampling Ratio 32	Sampling Ratio 64	Sampling Ratio 128	Sampling Ratio 256	Sampling Ratio 512	Sampling Ratio 1024	Sampling Ratio 2048
1	242.2404	212.7110	162.6798	114.7660	70.8305	43.4747	35.1887	30.8267	29.0305	18.8354	16.5427
2	138.6581	124.5788	102.2397	85.4389	57.5531	25.9525	18.6515	16.1527	16.4440	11.8930	8.1552
4	75.0081	68.5157	59.2523	50.9060	37.2917	18.9699	10.1120	8.0755	8.2098	5.8053	4.1680
8	38.2883	35.4682	31.1617	27.1275	20.2290	9.6828	5.5093	4.0439	4.0697	3.0790	2.7430
16	18.6976	17.3089	15.1980	13.1904	9.8597	4.9195	2.7271	1.9925	1.9742	1.5072	1.3893
32	9.0945	8.5035	7.4591	6.4618	4.9098	2.4828	1.3770	0.9187	0.9439	0.6989	0.7007
64	4.3821	4.0462	3.5564	3.0984	2.3216	1.1127	0.5901	0.4655	0.4772	0.3972	0.3959
128	2.0626	1.8997	1.6770	1.4909	1.0929	0.5604	0.3114	0.2049	0.2577	0.1881	0.2010
256	0.9045	0.8420	0.7686	0.6778	0.4952	0.3497	0.2504	0.1565	0.1163	0.0901	0.0738
512	0.5299	0.4928	0.4571	0.4081	0.3405	0.2615	0.1753	0.1124	0.0666	0.0413	0.0421
1024	0.4458	0.4028	0.3652	0.3202	0.2671	0.1908	0.1289	0.0861	0.0555	0.0402	0.0310
2048	0.2031	0.1721	0.1522	0.1253	0.1074	0.0877	0.0539	0.0298	0.0180	0.0111	0.0133

TABLE III. AVERAGE ERRORS AT DIFFERENT SAMPLING RATIOS

Time Granularity (Seconds)	Average Error on Bytes										
	Sampling Ratio 2	Sampling Ratio 4	Sampling Ratio 8	Sampling Ratio 16	Sampling Ratio 32	Sampling Ratio 64	Sampling Ratio 128	Sampling Ratio 256	Sampling Ratio 512	Sampling Ratio 1024	Sampling Ratio 2048
1	0.5995	0.5795	0.5542	0.5282	0.4883	0.4318	0.3711	0.3126	0.2683	0.2456	0.2404
2	0.5117	0.4877	0.4656	0.4389	0.4038	0.3536	0.3011	0.2554	0.2156	0.1921	0.1832
4	0.4436	0.4152	0.3892	0.3651	0.3335	0.2851	0.2423	0.2051	0.1748	0.1531	0.1396
8	0.3916	0.3612	0.3353	0.3075	0.2777	0.2295	0.1927	0.1666	0.1387	0.1194	0.1078
16	0.3475	0.3169	0.2921	0.2599	0.2310	0.1882	0.1534	0.1302	0.1110	0.0960	0.0844
32	0.3158	0.2863	0.2530	0.2233	0.1922	0.1488	0.1228	0.1015	0.0869	0.0765	0.0646
64	0.2833	0.2553	0.2225	0.1924	0.1638	0.1203	0.0989	0.0817	0.0687	0.0604	0.0486
128	0.2624	0.2299	0.1987	0.1719	0.1430	0.1005	0.0731	0.0585	0.0527	0.0428	0.0352
256	0.2497	0.2107	0.1804	0.1522	0.1231	0.0803	0.0559	0.0447	0.0334	0.0265	0.0214
512	0.2075	0.1765	0.1536	0.1299	0.1019	0.0590	0.0385	0.0285	0.0201	0.0168	0.0138
1024	0.1364	0.1127	0.1000	0.0889	0.0652	0.0377	0.0276	0.0255	0.0159	0.0138	0.0101
2048	0.1215	0.0888	0.0692	0.0555	0.0446	0.0329	0.0213	0.0106	0.0079	0.0066	0.0067

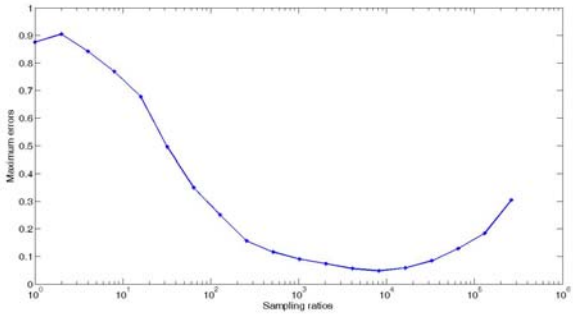

Figure 3. Trend of maximum errors and sampling ratios

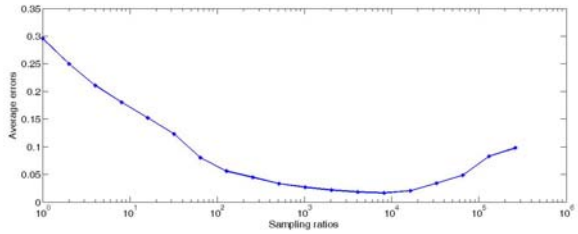

Figure 4. Trend of the average errors and sampling ratios

TABLE IV. SAMPLING RATIOS OF MINIMAL ERRORS

Time Granularity (Seconds)	Maximum Error (Minimum, Sampling Ratio)	Average Error (Minimum, Sampling Ratio)
1	(5.1400, 1/32768)	(0.2404, 1/2048)
2	(16.1527, 1/256)	(0.1832, 1/2048)
4	(8.0755, 1/256)	(0.1396, 1/2048)
8	(4.0439, 1/256)	(0.1035, 1/4096)
16	(0.4007, 1/8192)	(0.0742, 1/8192)
32	(0.9439, 1/512)	(0.0501, 1/4096)
64	(0.4655, 1/256)	(0.0355, 1/8192)
128	(0.2049, 1/256)	(0.0241, 1/4096)
256	(0.0478, 1/8192)	(0.0164, 1/8192)
512	(0.0421, 1/2048)	(0.0088, 1/4096)
1024	(0.0339, 1/16384)	(0.0056, 1/4096)
2048	(0.0111, 1/1024)	(0.0066, 1/1024)

From Table I, it is obvious that the bigger the time granularity to estimate utilized bandwidth becomes, the less the errors becomes. Even though the errors become smaller as the time granularities increase, the average error at max time granularity 2048s (more than half an hour) is also more than 10%. So, using NetFlow records to estimate utilized bandwidth at time granularities less than half an hour could cause large errors according to our experiment results.

2) Estimation errors at different sampling ratios

To quantify the impact of sampling ratio, we use 18 different sampling ratios to generate the NetFlow records, and then calculate the utilized bandwidth from those records. Only first 11 kinds of sampling ratios' error results are shown in Table 2 and Table 3 for the limitation of the table size. Table 2 and Table 3 show the maximum errors and average errors of each sampling ratio and time granularity separately. The first columns of these tables are all different time granularities (seconds), the next 11 columns show the results of the first 11 kinds of sampling ratios used in our experiments.

It is clear in Table II and Table III that under the same time granularities, both maximum errors and average errors decrease when the sampling ratio decreases and can reach to a minimal value. Throughout this process, the error has trivial fluctuation occasionally, but the trend still exists. Also, under the same sampling ratio, the maximum and average errors decrease when time granularities increases.

In order to see the error trends clearly, we choose a time granularity that is 256 seconds and plot the error trends based on this time granularity in Figure 3 and Figure 4. Figure 3 plots the maximum errors under different sampling ratios; we can see that the minimal value of maximum errors is about 0.0478 where sampling ratio is 1/8192. Figure 4 plots the average errors under different sampling ratios; we can also find that the minimal value of average errors is about 0.0164 where sampling ratio is 1/8192.

Table IV lists minimal value of errors at all sampling ratios under certain time granularity, which means that once the time granularity to calculate the utilized bandwidth is chosen for different applications, the best sampling ratio is recommended here to archive the least estimation errors.

V. CONCLUSIONS

This paper analyzes the two factors that could cause errors in estimating utilized bandwidth measurement from NetFlow records, one is time granularity, and the other is sampling ratio. Our experiments were based on IP Trace collected at CERNET Jiangsu backbone.

The results of our experiments show that if sample ratio is fixed, the error of utilized bandwidth measured by NetFlow decreases gradually as the increase of time granularities. However, even if the time granularity is quite big, the error is still large. If time granularity is fixed, as sampling ratio decrease the error will decrease and has a minimal value. But most of these minimal errors are still large. Only when time granularity is big enough and the sampling ratio is small enough, the error can be acceptable.

Thus it can be seen, large error will occur when utilized bandwidth is measured by NetFlow without sampling, but we can satisfy the required precision if we adjust the time granularities and sampling ratios properly.

Our research has limitations for lack of data of long-term IP Trace from different networks. In the experiments we only use 2 hours' data from CERNET's backbone which is 10 gigabits/second. Links with lower bandwidth might have different errors under the same sampling ratio and time granularity. Larger time granularity could be set if we have sufficient long-term data. More experiments need to be done at data from various networks. These will be researched in our future work.

ACKNOWLEDGMENT

We thank the anonymous reviewers for their valuable comments. This work is supported by the National Basic Research Program of China under Grant No. 2009CB320505 and the National Key Technology R&D Program of China under Grant No.2008BAH37B04.

REFERENCES

[1] Carey Williamson, Internet Traffic Measurement, IEEE Internet Computing, Volume 5 Issue 6, November 2001, pp.70-74.

[2] Cisco Systems Inc. NetFlow Services Solutions Guide. http://www.cisco.com/en/US/docs/ios/solutions_docs/ netflow/nfwhite.html

[3] Dave Plonka. FlowScan:A network traffic flow reporting and visualization Tool. In Proceedings of the Fourteenth Systems Administration Conference(LISA-00), 2000, pp.305-318.

[4] Robin Sommer and Anja feldmann, NetFlow: Information loss or win?, IWW 02, Nov.6-8, 2002, Marseille, France.

[5] Baek-Young Choi, supratik Bhattacharyya. Observations on Cisco Sampled NetFlow. ACM SIGMETRICS Performance Evaluation Review, 2005.

[6] Nick Duffield, Carsten Lund, Mikkel Thorup, Properties and Prediction of Flow Statistics from Sampled Packet Streams, Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurement, 2002.