Automated Measurement of High Volume Traffic Clusters

Cristian Estan, Stefan Savage, George Varghese
{cestan,savage,varghese}@cs.ucsd.edu

Abstract— **Traffic measurement often focuses on measuring traffic at various granularities. Our paper considers an approach that generalizes previous solutions: we define a *traffic cluster* to consist of all traffic that matches a specified set of values for certain header fields. While existing technology (e.g., Cisco ACLs) allows managers to measure specific traffic clusters, this requires a priori knowledge of what traffic clusters are worth watching. The main contribution of this paper is to suggest that automatically identifying and measuring high volume traffic clusters provides useful traffic reports to network managers *without a priori knowledge*.**

I. Introduction

Operating a large IP network involves many activities that rely on measuring the traffic composition: computing the "traffic matrix" for traffic engineering, measuring the contribution of various applications, detecting denial of service attacks, etc. All these activities aim to measure large aggregates of traffic, but they have *different ways of defining the aggregates*. In this paper we propose a new way of building traffic reports that relies on automatically identifying high volume traffic clusters that generalize the other aggregates. Thus a single concise report is used by all. While we do have promising preliminary results we consider the design of efficient algorithms for producing reports based on traffic clusters an open question.

II. Problem definition and related work

We focus on traffic measurement systems that measure traffic composition at various points in an IP network. The network operator would like to have accurate information that gives her flexibility in the analyses she performs. Further, the data should ideally be delivered in a timely manner that allows prompt reactions to events in the network. The most important limiting factor that thwarts the simultaneous realization of these two goals is the bandwidth required to deliver the data to the Network Operations Center (NOC). With all traffic measurement solutions one can increase the accuracy by increasing the size of traffic reports. Another common limiting factor is the amount of memory and the processing power available at the traffic measurement module. Existing solutions include:

NetFlow: In the widely deployed NetFlow [4], the traffic measurement module sees the headers of all (or sampled) packets and groups packets into fine-grained flows (source and destination IP and port, protocol, TOS byte). The traffic report consists of the flow identifiers, the packet and byte counts for each flow, and the time when the first and last packet of the flow was received. This type of fine-grained information allows very *flexible* queries such as counting all traffic to a subnet. The main problem is that the size of the report can be so large that it overwhelms the collection server or the network.

Aggregated NetFlow: Cisco's aggregated NetFlow solves the problem of large reports by combining fine-grained flow information into coarser aggregates (e.g., based on source prefix or port numbers) before sending to the NOC. The main problem is that since the data is aggregated at the router, the NOC loses the flexibility of running arbitrary queries because information is lost during aggregation. For example, if the data was aggregated by source and destination prefixes, port information was lost.

Large Flow Identification: Estan and Varghese [2] propose dynamically identifying and reporting only the large "flows" at a router, since usually applications are interested mostly in these. A nice property of this approach is that it bounds the size of the report: if large flows are defined as those sending more than 1% of the traffic, the report will contain at most 100 records. Unfortunately, the algorithms require all flows to be defined by the same fields. This limits the queries the NOC can perform.

Sampled Charging: Duffield et al [1] propose a technique that reduces the size of NetFlow traffic reports while still allowing the NOC to *flexibly* compute per customer aggregates of traffic for billing purposes. Sampled charging starts with the same fine-grained flow records as NetFlow but includes into the traffic report only flows that are

System	Flex.	Aut.	Acc.	Rep.	Mem.
NetFlow	+	+	+	-	-
Aggr. NetFlow	-	+	+	+	-
Large Fl. Ident.	-	+	+	+	+
Sampled Chrg.	+	+	-	+	-
sFlow	+	+	-	-	+
ACL	+	-	+	+	+
Traffic cluster	+	+	+	+	+

TABLE I

COMPARISON OF TRAFFIC MEASUREMENT SYSTEMS. THE COLUMNS HAVE THE FOLLOWING MEANINGS: 1) DOES IT ALLOW FLEXIBLE QUERIES? 2) IS IT AUTOMATIC? 3) ARE REPORTS ACCURATE? 4) ARE REPORTS CONCISE? 5) IS IT FRUGAL IN MEMORY USAGE AT THE ROUTER?

above a sampling threshold; flows below the threshold are included with probability proportional to their size. At the NOC, when the fine-grained flow records are aggregated into per customer totals (for arbitrary definitions of customer), one compensates for the dropped records. This results in an unbiased estimator for the traffic of the high-usage customers, but one with a large variance when applied over short timescales [1].

Header Export: The sFlow protocol [5] removes the traffic measurement module from the router: the router exports the full headers of sampled packets to the collection station. While this works well for small networks, for large networks the amount of data generated would likely overwhelm the NOC. One can reduce traffic by sampling less often, but this reduces accuracy.

ACLs: Access Control Lists are a feature of routers that allows the network operators to express complex classification rules that defines traffic clusters. While the original purpose of ACLs was to implement fine-grained access control policies, a corresponding counter can also be incremented for any packet that matches an ACL rule.

Table I compares existing traffic measurement systems with the strategy proposed in this paper. Note that all previous schemes are either inflexible, require manual configuration, generate large reports, or are inaccurate. This motivates us to search for a new solution that is flexible, automatic, accurate, and generates concise reports.

III. TRAFFIC REPORTS BASED ON TRAFFIC CLUSTERS

Each cluster has a definition which contains a pattern for each field. A packet belongs to the cluster if all its fields match the respective field patterns. For simple fields such as the protocol number, there can be only two types of patterns: exact match or a wild card (abbreviated to "*") that will be matched by all values. For IP addresses we also allow the patterns to be defined as prefixes of any length between 0 (which is equivalent to *) and 32 (equivalent to exact match). For ports we also allow range matches (e.g., source port between 8000 and 9000).

We could restrict prefixes to those present in a routing table but [3] argues that this reduces robustness. Thus we prefer to detect the relevant prefixes based only on current traffic. Many applications (e.g. NetMeeting) use port ranges instead of fixed ports. We would also like the algorithm to detect such significant port ranges based on actual traffic. Of course, this should not preclude the network operator from manually specifying prefixes and port ranges.

Our ultimate goal is to build efficient algorithms that produce a traffic report from which one can accurately estimate the size of all clusters whose traffic is above a certain threshold. Note that each packet belongs to many overlapping clusters, thus there can be much more than 100 clusters above 1% of the link traffic. We have preliminary solutions based on explicitly identifying the high volume traffic clusters and including them in the traffic report.

IV. CONCLUSIONS AND FUTURE WORK

In this paper we propose a new strategy for building concise traffic reports that allow flexible and accurate queries at the NOC. Our algorithms explicitly identify and measure high volume clusters within the traffic. Preliminary analyses and simplified experiments show promise.

Further work needs to address these problems:

- Algorithms that can efficiently (in both time and space) generate reports based on traffic clusters and if possible further increase the accuracy of these reports;
- Studying how these algorithms can support at the same time both clusters automatically identified and clusters manually configured by network operators.

This work was made possible by a grant from NIST for the Sensilla Project.

REFERENCES

[1] Nick Duffield, Carsten Lund, and Mikkel Thorup. Charging from sampled network usage. In *SIGCOMM Internet Measurement Workshop*, November 2001.

[2] Cristian Estan and George Varghese. New directions in traffic measurement and accounting. In *Proceedings of the ACM SIGCOMM*, August 2002.

[3] Anja Feldmann et al. Deriving traffic demands for operational IP networks: Methodology and experience. In *Proceedings of the ACM SIGCOMM*, pages 257–270, August 2000.

[4] Cisco NetFlow. `http://www.cisco.com /warp /public /732 /Tech /netflow`.

[5] Peter Phaal, Sonia Panchen, and Neil McKee. RFC 3176: sFlow, September 2001.