Anonym: A Tool for Anonymization of the Internet Traffic

Tanjila Farah and Ljiljana Trajković
Simon Fraser University
Vancouver, British Columbia, Canada
{tfarah, ljilja}@sfu.ca

***Abstract*—Collecting network traffic traces from deployed networks is one of the basic steps in understanding communication networks. Traffic traces are used for network management, traffic engineering, packet classification, and analyzing user behavior to ensure adequate quality of service. Monitored traffic traces should be anonymized for privacy and security reasons. The goal of anonymization is to preserve trace properties while enforcing privacy policies. Various tools and techniques have been implemented for trace anonymization. In this paper, we propose and implement an anonymization tool that executes multi-level anonymization and displays analysis results. We describe architecture and features of the tool and discuss analysis of un-anonymized and anonymized datasets.**

***Index Terms*—Network traffic, anonymization, traffic analysis.**

I. Introduction

Measurement, characterization, and classification of the Internet traffic help ensure security of the Internet users and provide information to the network administrators to better manage the network. Traffic analysis relies on collection of trace logs from network service providers. Network traffic logs include data packet headers, which contain the source and destination Internet Protocol (IP) addresses, port numbers, type of packets and protocols, and information related to the packet source and destination. Sharing these network traces may reveal the network architecture, user identity, and user information. This makes the network vulnerable to attacks. One solution is to employ anonymization of collected traffic. Anonymization is a compromise between preserving research value of the trace and protecting the user privacy.

Réseaux IP Européens (RIPE) [1], Route Views [2], and Corporative Association for Internet Data Analysis (CAIDA) [3] provide anonymized collected traffic traces to the research community. A variety of fields that should be anonymized, anonymization techniques, and anonymization tools have been considered [4]. The prefix preserving anonymization is the most frequently used algorithm for anonymization of IP addresses. This technique was implemented in the Crypto-PAn [5] tool. The Anontool allows per-field anonymization and provides various anonymization algorithms [6]. The FLAIM anonymization tool has an application program interface [4]. There are trade-offs between privacy, security, and efficiency when sanitizing collected data [7]. Finding a balance between security requirements of the organization that shares the trace logs and their research usefulness has been a topic of research interests [8]. Anonymization approaches have various limitations [9]. Trace anonymization should preserve the research value of the trace and, hence, analysis of anonymized and un-anonymized datasets such as statistical distributions, traffic volume, presence of anomalous traffic, and results of network-wide traffic analysis should lead to similar results [10].

We propose a tool named *Anonym* to anonymize stored traffic traces. It supports pcap and mrt text format trace files and provides options for traffic analysis. In this paper, we provide overview of the tool and statistical analysis of anonymized and un-anonumized data.

This paper is organized as follows. Motivation for collecting network traffic traces is given in Section II. Anonymization algorithms and anonymized fields are described in Section III. In Section IV, we describe the *Anonym* tool, its functionality, and discuss analysis of anonymized and un-anonumized datasets. We conclude with Section V.

II. Collection of Network Traffic

Network operators continuously collect and monitor network traffic. Collecting and exploring network traffic help understand its characteristics. They assist in the development of methodologies and techniques to facilitate reliable network operations, discover network traffic patterns, and optimize network resources. Network traces are used in traffic engineering, discovery of Internet topologies, and network security analysis.

1) Traffic Engineering: Measurement, characterization, modeling, and control of the Internet traffic requires understanding network traffic patterns [10]. The role of traffic engineering encompasses network troubleshooting, protocol debugging, workload characterization, performance evaluation, and capacity planning [11].

2) Discovering Internet Topologies: Mapping the Internet network [12], [13] is important for the design of new protocols, applications, and routing policies. The Internet service providers (ISPs) rely on Internet topologies for network planning and network management.

3) Network Security Analysis: Analyzing the network-wide traffic behavior helps detect abnormal events such as malicious attacks and Internet viruses and determine the spread of network anomalies and their classification and diagnosis [10].

III. Anonymization Algorithms

Anonymization is the modification of network traffic data in order to protect the identity of the end users. The goal of anonymization is to remove the ability to identify the connection between two end-points while preserving the usefulness of the data. The network traffic data, also known as a network flow, consists of a sequence of packets from source to destination end-points. It is defined by five fields (5-tuple): source address, destination address, source port number, destination port number, and protocol type. A flow record includes additional fields such as packet length, Media Access Control (MAC) address, flow number, Autonomous System (AS) number, window size, maximum segment size, and payload length. Some of these fields uniquely identify end-points while others identify the pattern and behavior of network users. Anonymization processes consider all these fields in hope to increase the anonymity of the data [14].

Network data may be protected by controlling the access to data or by applying mechanism such as anonymization of the dataset. A raw packet trace contains various data fields. They may contain sensitive data and, hence, it is important to carefully select fields to be anonymized. Not all fields require anonymization.

Anonymization algorithms or anonymization primitives provide various levels of anonymization of the collected datasets [15]. The precision and efficiency of an anonymization algorithm are important to maintain the research value of the collected data [15]. Listed are some known anonymization algorithms:

Black marker algorithm is the most extreme method of anonymization. It deletes or replaces with a fixed value all information in a field. It may be applied to all fields of a network flow. Although it entirely protects the data, it also reduces the usefulness of the anonymized dataset.

Enumeration algorithm works on a well-ordered set of data. After sorting the data, it selects the first value from the record. For each successive record, it selects a larger number. This algorithm may not be applied to all fields. The enumerated data is useful for the analysis requiring strict sequencing. However, the anonymization destroys the time-stamp information.

Hash algorithm replaces the data with a fixed size bit string. Any change in the data alters the hash value. The results of a hash function are sometimes shorter than the field value. Hence, the hash algorithm is easy to break. One form of hash algorithm is the Hash Message Authentication Code (HMAC).

Partitioning algorithm partitions a set of possible values into subsets by an equivalence relation. A canonical example for each subset is chosen. The anonymization function then replaces each data with the canonical value [4].

Precision degradation algorithm removes the most precise content of a time-stamp field. The anonymized data may not be useful for applications that require strict sequencing of flows.

Permutation algorithm is mostly used for anonymization of the IP and MAC addresses. It applies a random permutation to map un-anonymized addresses to a set of possible addresses. It uses one hash table for mapping from un-anonymized to anonymized IP addresses and another to store all anonymized addresses. There are many variations of permutation functions, each having trade-offs in terms of performance. Permutation functions are random and prefix-preserving.

Prefix-preserving pseudonymization algorithm is similar to the *Permutation* algorithm. A function F is prefix-preserving if two addresses a and b share the first n bits. This is one-to-one mapping on a set of values generated from a block cipher. It is used to anonymize the IP addresses. This algorithm preserves the structure by preserving the prefixes values. Cryptographic keys are used to keep the mapping consistent [16], [17].

Random time shift algorithm adds a random offset to every value in a field.

Truncation algorithm is used to anonymize the IP and the MAC addresses. It deletes a portion of the data while keeping the remaining unchanged. Truncation removes n least significant bits from a field value by replacing them with zeros. This technique is effective to make an end-point non-identifiable.

Reverse truncation algorithm is used to anonymize the IP and MAC addresses. It removes n of the most significant bits. This technique is effective to make a network address or an organization non-identifiable.

Time unit annihilation is a partitioning algorithm used to anonymize time-stamps. It annihilates a portion of the time-stamp by replacing it with zeros.

The usually anonymized fields are: IP address, MAC address, port numbers, length of packets, time-stamps, and counters. Multiple anonymization algorithms are available for fields that commonly appear in the traffic data, as shown in Table I.

TABLE I
Fields and the Related Anonymization Process.

Field	Anonymization algorithm
IP address	Truncation, Reverse truncation, Permutation, Prefix-preserving pseudonymization, Black marker
MAC address	Truncation, Reverse truncation, Permutation, Structured pseudonymization, Black marker
Time-stamps	Precision degradation, Enumeration, Random-time shift, Black marker
Counter	Precision degradation, Binning, Random noise-addition, Black marker
Port number	Binning, Permutation, Black marker

IV. The Anonym Tool

The developed Anonym tool anonymizes time-stamps, IP addresses (IPv4 and IPv6), MAC addresses, port numbers, and packet length fields. The Anonym is a MATLAB-based tool that supports pcap and mrt format input file and provides options to convert there text file to pre-anonymized dataset file. This pre-anonymized file is the parsed input file that contains data columns corresponding to fields to be anonymized. Anonym supports multiple anonymization algorithms and data analysis options. The anonymization results and the graphs of data analysis are displayed in the output and figure screens. The results of anonymization are saved in the same format

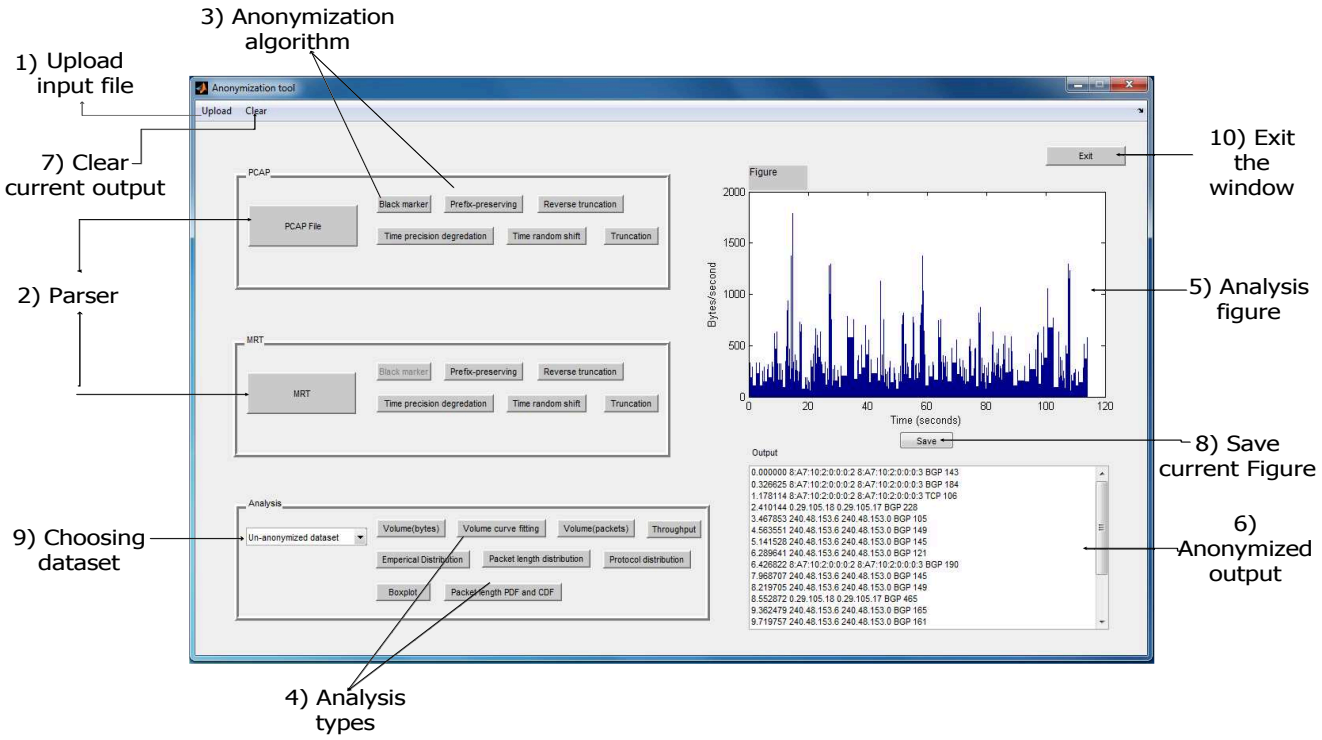

Fig. 1. Interface of the *Anonym* tool.

as the input file (pcap or mrt). Anonym supports Linux and Windows operating systems.

Anonym is developed with a graphical user interface (GUI). The functionality of the *Anonym* GUI is shown in Figure 1, where:

1) *Upload* option is used for uploading pcap and mrt input files.
2) *PCAP and MRT* options are used for parsing the input files for anonymization and analysis.
3) *Anonymization algorithms* currently supported are: Black marker, Prefix-preserving, Reverse truncation, Time precision degradation, Time random shift, and Truncation.
4) *Analysis types* currently supported are: Volume (bytes), Volume curve fitting, Volume (packets), Empirical distribution, Packet length distribution, Throughput, Protocol distribution, Boxplot, and Packet length PDF and CDF.
5) *Analysis figure* appears in the figure window.
6) *Anonymized output* appears in the output window.
7) *Clear* option removes results from the figure and output windows.
8) *Save* option saves the figure that has been displayed in the figure window.
9) *Choosing dataset* option selects the dataset for analysis.
10) *Exit* option closes the tool.

The *Anonym* tool includes thirty options. The operational diagram of the *prefix preserving* option is shown in Figure 2. The user first uploads an input file and chooses an appropriate parsing function. After the parsing, a function is called to separate IPv4 and IPv6 flow records. Two functions for anonymizing IPv4 and IPv6 flow records perform the *prefix preserving* anonymization on IP address fields. The *Time precision degradation* function anonymizes time-stamp field while the *Enumeration* function anonymizes packet length field. The anonymized IPv4 and IPv6 flow records are saved in two separate files. A function assembles these flow records in the order as they appeared in the input file. The output function then displays the anonymized results in the output window. Another function rewrites the input file with anonymized fields.

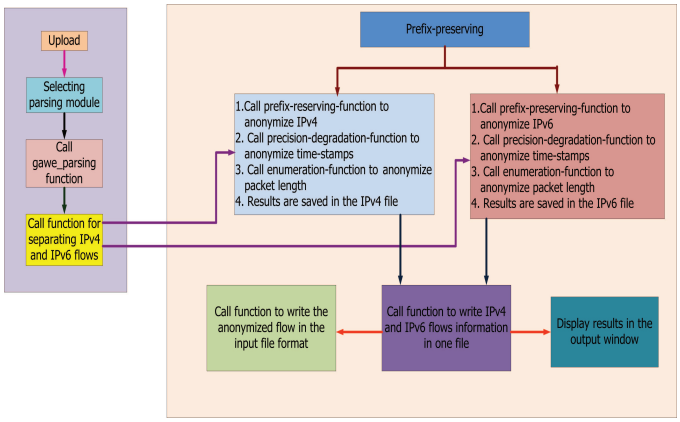

Fig. 2. Operational diagram of the prefix-preserving option.

Anonym supports options that are not available in the existing anonymization tools. One such option is the anonimyzation of IPv6 addresses. The IPv6 addresses were introduced to the Internet in June 2012 and, hence, the existing tools do not perform IPv6 anonimyzation. IPv6 uses 128 bit addresses and its address structure is shown in Figure 3.

Un-anonimyzed dataset:
0.000000 2001:4958:10:2::2 2001:4958:10:2::3 BGP 143
4.563551 206.47.102.206 206.47.102.201 BGP 149
Anonimyzed dataset:
0.000000 8:A7:10:2:0:0:0:2 8:A7:10:2:0:0:0:3 BGP 243
4.560000 240.48.153.6 240.48.153.0 BGP 249

Fig. 3. Anonymization process of IP addresses.

A MAC address uniquely identify a device in a network interface. Anonym tool provides *Truncation* and *Reverse truncation* algorithms to anonimyze MAC addresses. The *Truncation* anonymization process is shown in Figure 4.

Un-anonimyzed dataset:
1.178114 Cisco_ e7:a1:c0 (00:1b:0d:e7:a1:c0)
JuniperN_ 3e:ba:bd (78:19:f7:3e:ba:bd) TCP 106
Anonimyzed dataset:
1.178114 Cisco_ e7:a1:c0 (00:1b:0d:e7:0:0)
JuniperN_ 3e:ba:bd (78:19:f7:3e:0:0) TCP 106

Fig. 4. Anonymization process of MAC addresses.

Anonym provides option to apply the Kolmogorov-Smirnov (K-S) test to compare datasets with various reference probability distributions and to infer the underlying structure of the network traffic. The test helps extract variables, detect outliers and anomalies, test underlying assumptions, and develop a theory-based models of the traffic trace.

We compared the Anonym tool with two existing tools: Anontool and FLAIM. The comparison is shown in Table II.

TABLE II
COMPARISON OF ANONYMIZATION TOOLS.

Tool	Input	Fields
Anontool	pcap, netflow (v5 and v9)	IPv4 address, MAC address, port numbers, length of packets, time-stamps, and counters
FLAIM	pcap, nfdump	IPv4 address, MAC address, port numbers, length of packets, time-stamps, and counters
Anonym	pcap, mrt	IP address (v4 and v6), MAC address, port numbers, length of packets, time-stamps, and counters

V. ANALYSIS OF DATASETS

Analysis of the network traffic [18] is an essential element of understanding network requirements and capabilities. It addresses average load and the bandwidth requirements for various applications. Statistical analysis of network traffic may also reveal the user behavior pattern. Anonymization may cause loss of information in collected datasets. We employ statistical modeling and analysis of both anonymized and un-anonymized datasets and show that anonymization does not significantly change their statistics. The options include the distribution of various protocols, analysis of traffic volume as function of time, and statistical distributions of packet length. The traffic traces are processed in a human readable form. The statistical analysis is performed using the Anonym tool.

A. Traffic Volume

Analysis of traffic volume is used for classification of network traffic, for traffic engineering, measuring network usage, and anomaly detection. Traffic volume is measured as the number of bytes or packets sent and received. Traffic volumes of the Border Gateway Protocol (BGP), Transmission Control Protocol (TCP), and User Datagram Protocol (UDP) are shown in Figure 5. The data was collected from the BCNET between May and September 2012 [19], [20]. The same traffic patterns appear for anonymized and un-anonymized datasets.

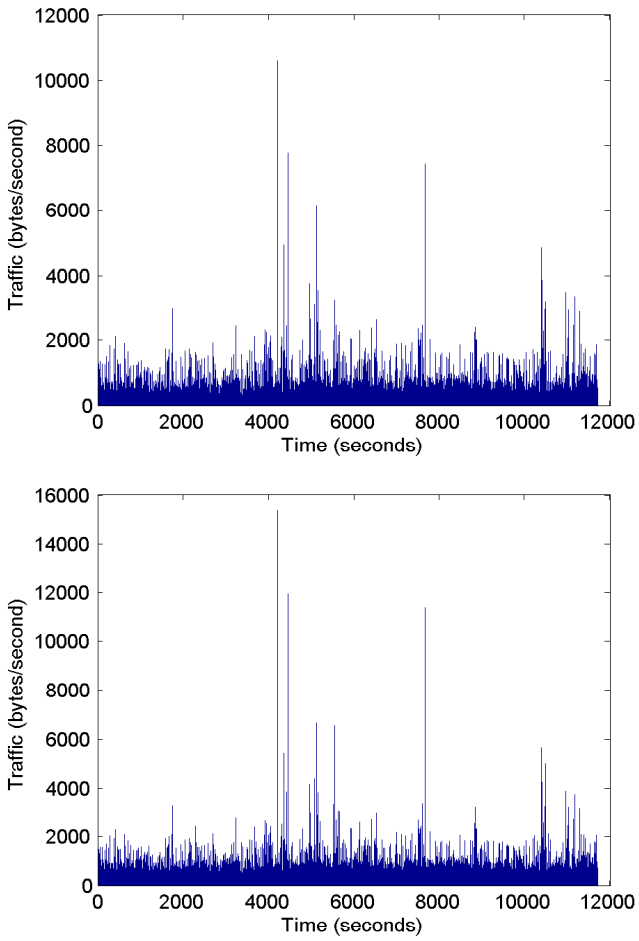

Fig. 5. Traffic volume in bytes: un-anonymized (top) and anonymized (bottom) datasets.

The visual inspection of datasets shows no significant difference and, hence, we employed statistical tests to analyze their distribution patterns. We considered exponential, polynomial, and Fourier distributions of the un-anonymized and anonymized datasets, as shown in Figure 6.

The minimum, maximum, mean, median, and standard deviation for un-anonymized and anonymized datasets are shown in Table III. The minimum, maximum, mean, and median values

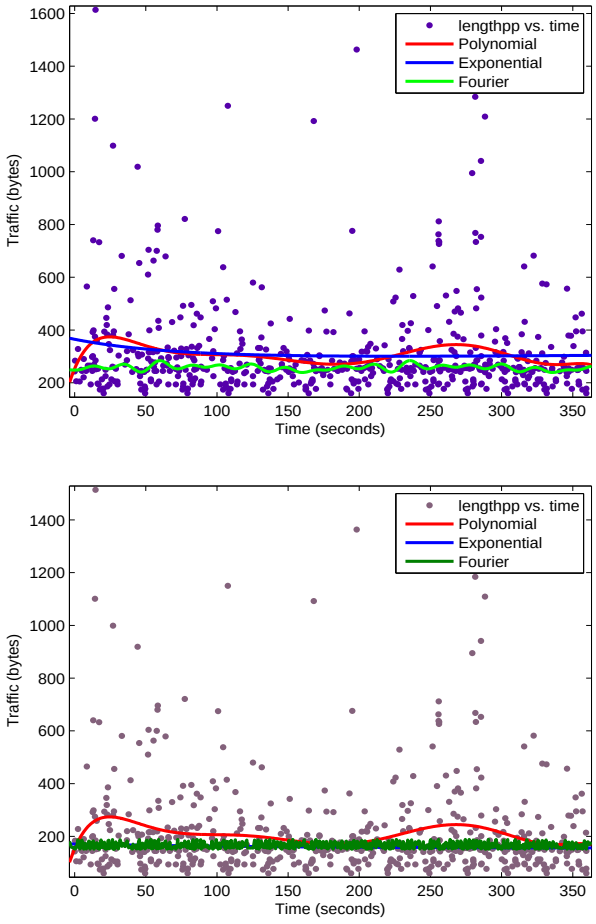

Fig. 6. Fitting curves to traffic volume graph: un-anonymized (top) and anonymized (bottom) datasets.

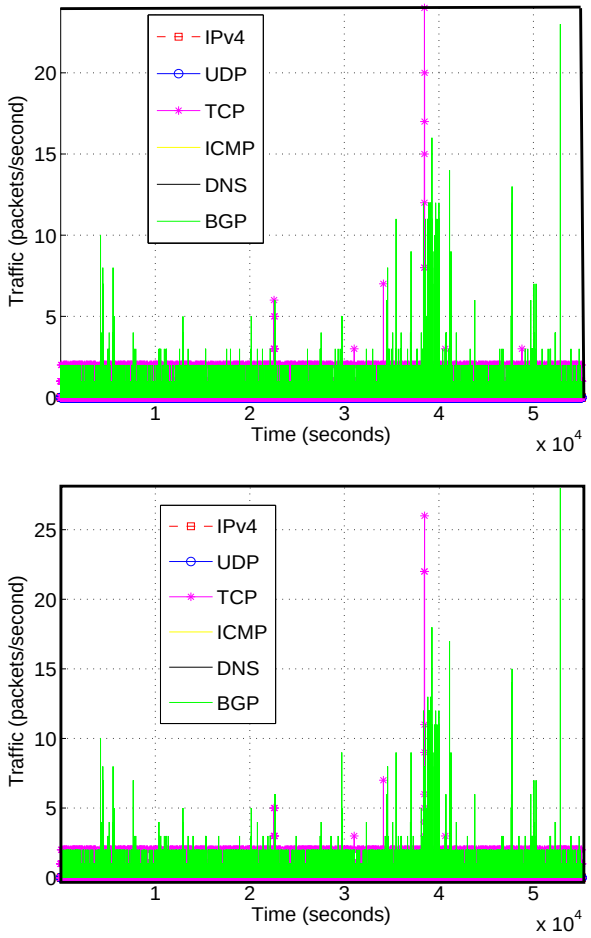

Fig. 7. Packet distribution by protocols: un-anonymized (top) and anonymized (bottom) datasets.

have been increased by 100 bytes due to the *Enumeration* algorithms used for anonymizing the dataset. However, the standard deviation values indicate that the structure of the datasets remains unchanged.

TABLE III
STATISTICS OF DATASETS.

Statistics	Un-anonymized dataset	Anonymized dataset
Minimum	60	160
Maximum	1514	1614
Mean	246.2475	346.2475
Median	157	257
Standard deviation	259.4509	259.4509

B. Protocol Distribution

Packets were classified according to protocols, as shown in Figure 7. The BGP packets are most common. In the anonymized dataset, time field is anonymized using the *Precision degradation* algorithm. Traffic throughput is shown in Figure 8. Anonymization has not change the throughput statistics of the dataset.

C. Packet Length Distribution

Histogram of packet length distribution is shown in Figure 9. The patterns of un-anonymized and anonymized datasets show insignificant variations. The only visible difference is the change of the packet lengths due to the *Enumeration* algorithm.

VI. CONCLUSION

Sharing network trace data helps improve quality of service and network security. In this paper, we introduced *Anonym*, a network trace anonymization tool. The tool is used to anonymize time-stamps, the IPv4 and IPv6 addresses, MAC addresses, port numbers, and packet length fields. We compared un-anonymized and anonymized datasets. Traffic data collected from BCNET were used to analyze temporal patterns and for fitting traffic volume to known distributions. The traces are anonymized and analyzed using the developed tool. Investigation of packet distributions shows no significant variations between the un-anonymized and anonymized traces.

REFERENCES

[1] (1989-2012) RIPE (Réseaux IP Européens). [Online]. Available: http://www.ripe.net.

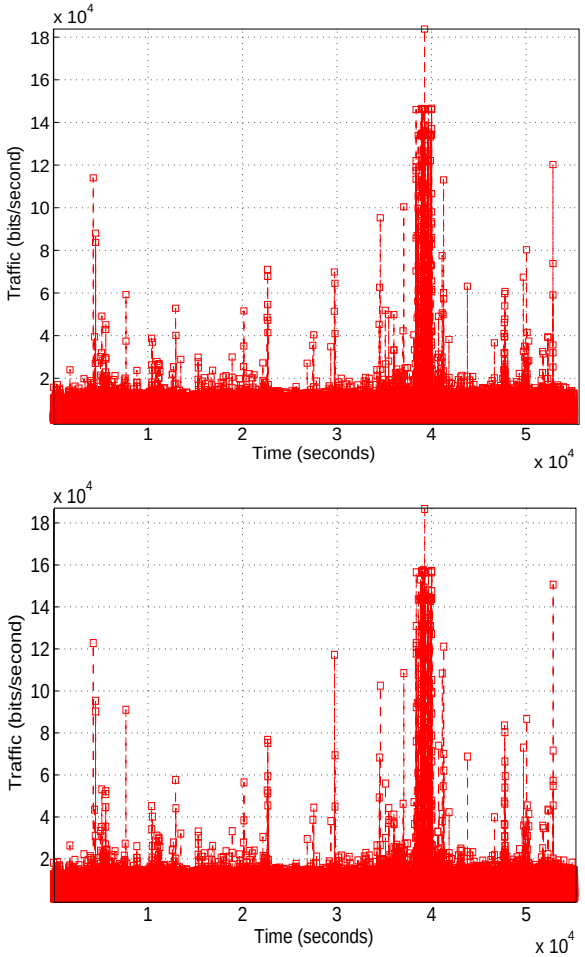

Fig. 8. Throughput: un-anonymized (top) and anonymized (bottom) datasets.

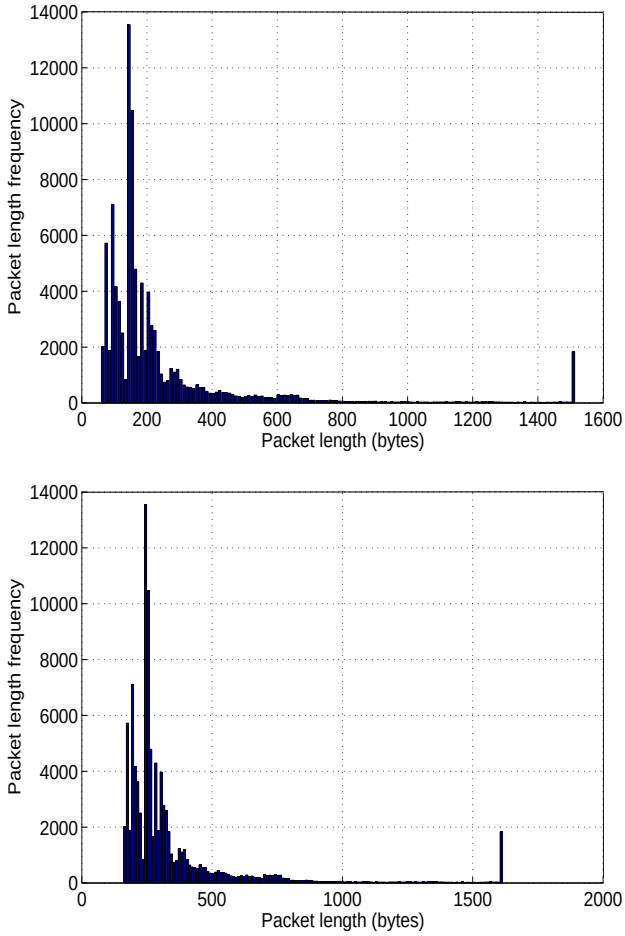

Fig. 9. Distribution of packet length: un-anonymized (top) and anonymized (bottom) datasets.

[2] University of Oregon Route Views Project. [Online]. Available: http://www.routeviews.org.

[3] The Corporative Association for Internet Data Analysis. [Online]. Available: http://www.caida.org/data.

[4] A. Slagell, K. Lakkaraju, and K. Luo, "FLAIM: a multi-level anonymization framework for computer and network logs," in *Proc. 20th Conference on Large Installation System Administration*, Washington, DC, USA, July 2006, pp. 101–115.

[5] J. Xu, J. Fan, M. Ammar, and S. B. Moon, "On the design and performance of prefix-preserving IP traffic trace anonymization," in *Proc. 1st ACM SIGCOMM Workshop on Internet Measurement*, San Francisco, CA, USA, Nov. 2001, pp. 263–266.

[6] M. Foukarakis, D. Antoniades, S. Antonatos, and E. P. Markatos, "Flexible and high-performance anonymization of netflow records using anontool," in *Proc. Third International Workshop on the Value of Security Through Collaboration*, Nice, France, Sept. 2007, pp. 33–38.

[7] M. Bishop, B. Bhumiratana, R. Crawford, and K. Levitt, "How to sanitize data," in *Proc. 13th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises*, Washington, DC, USA, June 2004, pp. 217–222.

[8] R. Pang, M. Allman, V. Paxson, and J. Lee, "The devil and packet trace anonymization," *ACM SIGCOMM*, Aug. 2006, vol. 36, pp. 29–38.

[9] R. Crawford, M. Bishop, B. Bhumiratana, L. Clark, and K. Levitt, "Sanitization models and their limitations," in *Proc. 2006 Workshop on New Security Paradigms*, Schloss Dagstuhl, Germany, Mar. 2006, pp. 41–56.

[10] A. Lakhina, M. Crovella, and C. Diot, "Mining anomalies using traffic feature distributions," in *Proc. 2005 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, New York, NY, USA, Aug. 2005, vol. 35, pp. 217–228.

[11] D. Awduche, A. Chiu, A. Elwalid, I. Widjaja, and X. Xiao, "Overview and principles of Internet traffic engineering," RFC 3272, *IETF*, May 2002.

[12] G. Siganos, M. Faloutsos, P. Faloutsos, and C. Faloutsos, "Power-laws and the AS-level Internet topology," *IEEE/ACM Trans. Networking*, vol. 11, no. 4, pp. 514–524, Aug. 2003.

[13] Lj. Trajković, "Analysis of Internet topologies," *IEEE Circuits and Systems Magazine*, Sept. 2010, vol. 10, no. 3, pp. 48–54.

[14] E. Bosch and B. Trammell, "IP flow anonymization support," RFC 6235, *IETF*, May 2011.

[15] M. Foukarakis, D. Antoniades, S. Antonatos, and E. P. Markatos, "On the anonymization and deanonymization of netflow traffic," in *Proc. Conference for Network Flow Analysis*, Savannah, GA, USA, Jan. 2008, pp. 272–275.

[16] J. Fan, J. Xu, M. H. Ammar, and S. B. Moon, "Prefix-preserving IP address anonymization: measurement-based security evaluation and a new cryptography-based scheme," *Computer Networks*, vol. 46, pp. 253–272, Oct. 2004.

[17] M. Peuhkuri, "A method to compress and anonymize packet traces," in *Proc. 1st ACM SIGCOMM Workshop on Internet Measurement*, San Francisco, CA, USA, Nov. 2001, pp. 257–261.

[18] W. E. Leland, W. Willinger, D. V. Wilson, and M. S. Taqqu, "On the self-similar nature of Ethernet traffic," *IEEE/ACM Trans. Networking*, vol. 2, no. 1, pp. 1–15, Feb. 1994.

[19] BCNET. [Online]. Available: https://wiki.bc.net.

[20] T. Farah, S. Lally, R. Gill, N. Al-Rousan, R. Paul, D. Xu, and Lj. Trajković, "Collection of BCNET BGP traffic," in *Proc. 23rd International Teletraffic Congress*, San Francisco, CA, USA, Sept. 2011, pp. 322–323.