Traffic Engineering and Characterization of High-Rate Large-Sized Flows

A Thesis

Presented to

the faculty of the School of Engineering and Applied Science

University of Virginia

in partial fulfillment

of the requirements for the degree

Master of Science

by

Tian Jin

December

2013

APPROVAL SHEET

The thesis

is submitted in partial fulfillment of the requirements

for the degree of

Master of Science

AUTHOR

The thesis has been read and approved by the examining committee:

Malathi Veeraraghavan

Advisor

Kevin Sullivan

Jack Davidson

Alfred C. Weaver

Accepted for the School of Engineering and Applied Science:

Dean, School of Engineering and Applied Science

December

2013

Abstract

High-rate large-sized (α) flows have adverse effects on delay-sensitive flows. Research-and-education network providers are interested in identifying such flows within their networks, and directing these flows to virtual circuits. To achieve this goal, a design was proposed for a hybrid network traffic engineering system (HNTES) that would run on an external server, gather NetFlow records from routers, analyze these records to identify α-flow source/destination address prefixes, configure firewall filter rules at ingress routers to extract future α flows and redirect them to provisioned virtual circuits. This thesis presents an evaluation of this HNTES design using NetFlow records collected over a 7-month period from four ESnet routers. The results show that the HNTES effectiveness was above 90% for NetFlow records collected at edge routers, which corresponded to file downloads from Department of Energy (DOE) laboratories, while the effectiveness was lower for peering routers whose NetFlow records corresponded to file uploads. With further investigation, we found that uploads were less frequent and involved fewer source/destination pairs than downloads.

The thesis also describes an algorithm for characterizing the size, duration, average rate, and frequency of α flows, from NetFlow records. The algorithm was validated using independently collected usage logs from application servers. This algorithm can be used in a network management system for providers interested in these types of flows, such as research-and-education network providers whose customers move large scientific datasets. We executed the algorithm on the same NetFlow records used in the HNTES evaluation. Flows moving datasets as large as 811 GB and at rates as high as 5.7 Gbps were observed. Some source-destination pairs were found to repeatedly create α flows. An analysis of the rates of the 1596 repeated α flows created by one pair

showed considerable variance, with minimum rate of 100 Mbps, maximum rate of 536 Mbps, and a coefficient of variation of 30%.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Research-and-eduation (REN) network providers have observed that high-rate large-sized flows (henceforth referred to as α *flows* [25]) have adverse effects on delay-sensitive flows. Therefore, there is an interest in identifying these α flows, directing them to separate virtual queues from general-purpose flows, and forwarding them onto virtual circuits.

As IP routers do not offer built-in capabilities to identify α flows, Z. Yan et al. proposed a network management software system called hybrid network traffic-engineering system (HNTES) to be run on an external server [30, 31]. HNTES conducts a posteriori analysis of NetFlow [16] records, which are exported by routers on a periodic basis. NetFlow is a technology that is built into IP routers to sample packets (e.g., 1-in-1000) and store packet-header fields such as source and destination IP addresses, port numbers, and protocol type, along with packet-arrival timestamps. Routers create NetFlow records by aggregating information about multiple sampled packets of the same flow that arrived within a preconfigured duration. HNTES extracts the source and destination addresses/prefixes of α flows, and uses these in a request to a virtual-circuit management system to enable isolation of future α flows from general-purpose flows. The virtual-circuit management system has the authority to configure virtual circuits, configure packet schedulers to support multiple virtual queues in router buffers, and to set firewall filter rules at ingress routers using the source/destination addresses/prefixes provided by HNTES. Future α flows whose addresses/prefixes match those of the firewall filter rules will be automatically directed to a separate virtual queue from general-purpose flows, and will be forwarded on to the established virtual circuits.

The first part of this thesis describes a detailed evaluation of HNTES using NetFlow records from four ESnet [4] routers.

In the second part of this thesis, we describe an algorithm for combining information from multiple NetFlow records to determine the size, duration, and average rate of α flows. The algorithm can be used in a network mangement system that helps network providers to characterize α flows, pinpoint routing misconfigurations, and assist their customers by improving performance.

Given the low NetFlow packet sampling rates used in ESnet [4] (specifically, 1-in-1000)[1], our algorithm needed to be validated. We conducted a validation exercise by procuring GridFTP usage logs [6] from a supercomputing center that is directly connected to ESnet, and NetFlow records from the corresponding ESnet router. The GridFTP usage logs provide file transfer sizes/durations. These values were matched with the flow characteristics determined by executing the algorithm on the ESnet NetFlow records. The algorithm was then applied to characterize α flows observed at four ESnet routers.

The following sections provide background information, describe the problem statement and motivation for the work, state our hypotheses, and list key contributions of this work.

1.1 Background

1.1.1 NetFlow

NetFlow [16] is a feature that enables IP routers to collect packet samples, and save information on a per-flow basis. The defining attributes of a flow can be configured, e.g., the five tuples {source IP address, destination IP address, source port number, destination port number, protocol type}. For each newly observed flow F, NetFlow opens a flow record and stores the arrival time instant of the first observed packet. For every new packet corresponding to flow F that is captured by the sampling process, NetFlow adds one to the flow-record packet count and increases the total size (bytes) by the packet-payload size. It also updates the last-packet timestamp field. At the end of the *active timeout interval* (time since first-packet arrival), *inactive timeout interval* (time since last-

[1] On high-speed core-network links, higher sampling rates are impractical.

packet arrival), or upon observing a TCP FIN or RST segment for flow *F*, the corresponding open NetFlow record is closed. The two timeout intervals are configurable. The closed NetFlow records are sent by the IP router's `NetFlow exporter` to a `NetFlow collector` (a process running on an external host). In ESnet, the packet sampling rate is 1-in-1000, the active and inactive timeout intervals are 60 sec each, and NetFlow records are exported every 5 mins.

1.1.2 ESnet

ESnet is a US-wide core (backbone) high-speed REN that offers IP-routed and dynamic virtual circuit services to DOE national laboratories, such as Argonne National Laboratory, Brookhaven National Laboratory, and several others [4]. As the DOE national laboratories conduct scientific research in many disciplines such as high-energy physics, α flows created by the movement of scientific datasets are observed on ESnet router interfaces.

In 2011, when the NetFlow records used in this thesis were collected, there were 75 routers in total, with 42 routers located in customer premises as provider edge (PE) routers, and the remaining routers were used in the core backbone (routers are located in cities such as Houston, Atlanta, etc.) and in three metro-area rings in Chicago, Northern California, and New York. All backbone links, and links from major PE routers to core routers were 10 Gbps Ethernet. ESnet peers with other US backbone RENs such as Internet2, and international RENs such as GEANT2, and with commercial peers and provider networks.

1.2 Problem Statement

The objectives of this work are to evaluate HNTES and to characterize α flows.

1.2.1 Evaluation of HNTES

The primary goal is to compare HNTES performance when using NetFlow records from different types of ESnet routers. Two of the selected routers whose NetFlow records were analyzed were edge routers, one was a core router with REN-peering, and one was a commercial-peering router.

Two performance metrics were used: (i) effectiveness, and (ii) afflicted-flow packets percentage (AFPP). Effectiveness is the percentage of bytes from α flows that would have been isolated from other flows had HNTES been deployed. The AFPP metric characterizes the percentage of packets from non-α delay-sensitive flows (afflicted-flows) that share address prefixes with α flows. The afflicted flows could suffer potential packet delays because HNTES operation requires IP routers to be configured to direct packets of flows with α prefix IDs to separate virtual queues.

1.2.2 α flow characterization

The goal of this work is to develop methods for characterizing α flows (on their size and duration dimensions) from NetFlow records, and to use these methods to characterize α flows observed at the four ESnet routers.

1.3 Motivation

1.3.1 Evaluation of HNTES

The prior work [30,31] was supported by a University of Virginia (UVA) US Department of Energy (DOE) grant. As a follow-on to this UVA grant, the US DOE funded a second HNTES project in which ESnet is a collaborator. ESnet is interested in enhancing HNTES capabilities and performance for eventual deployment. Further, from a research perspective, it was important to test whether the conclusions about HNTES performance based on the analysis of NetFlow records from a single router [30] are valid when NetFlow records collected from other routers are analyzed.

1.3.2 α flow characterization

Network operators are also interested in characterizing α flows traversing their network for various applications. Two examples are as follows. *First*, while REN peerings are usually the preferred routes for inter-domain traffic within the scientific community, sometimes these α flows moving large scientific datasets appear on the commercial peering links. Such events occur due to Border Gateway Protocol (BGP) misconfigurations. Characterizations of these α flows can assist providers in finding such misconfigurations. A *second* provider application of a system that characterizes α flows is to to assist customers in determining causes of poor performance. For example, if a user experiences high throughput variance determined by the α flow characterization system, PerfSONAR [22] can be used to help pinpoint the source of the problem.

1.4 Hypotheses

1.4.1 Evaluation of HNTES

Our hypothesis was "the effectiveness and AFPP metric values of HNTES computed using NetFlow records collected from different routers may not be the same."

1.4.2 α flow characterization

Our hypothesis was "larger α flows are likely to be observed at edge routers than at core/REN-peering and commercial-peering routers because downloads from national laboratories were observed at the two edge routers while the observation points at the peering routers captured uploads." DOE national laboratories support high-performance computing systems used by the scientific community. Large datasets are created on these systems through the execution of complex models, such as climate simulations, which are then downloaded by scientists to their own storage clusters.

1.5 Key contributions

1.5.1 Evaluation of HNTES

The main contributions of the HNTES evaluation work are: (i) definitions of HNTES performance metrics and relevant traffic measures, (ii) cross-sectional and longitudinal data analysis methods for quantifying these metrics, and (iii) interpreting the values obtained for these metrics toward explaining HNTES behavior. If HNTES is deployed, our software can be used for continuous monitoring of HNTES performance to make improvements if necessary.

These contributions matter because traffic spikes caused by large scientific dataset movement have been observed on research-and-education networks (RENs). Since users at the DOE laboratories use ESnet for both scientific data transfers and general-purpose applications, the ability to identify α flows and isolate them from general-purpose flows will improve user-perceived performance.

Key findings: (i) We found that HNTES effectiveness was higher than 90% if the NetFlow records used were from the edge routers. The samples were collected from the incoming side of externally facing interfaces. Each edge router was connected to only a single customer router, which means that observed α flows were mostly downloads from high-performance data transfer nodes (DTNs) located in the customer networks (ESnet's customers are mostly DOE national laboratories).

(ii) The HNTES metrics depend on two parameters: *aging parameter* and *address prefix length*. The aging parameter is used to age out address prefix entries from the firewall filter to limit its size. The larger the aging parameter, the longer the lifetime of firewall-filter rules, which implies a higher probability of matching newly arriving α flows' source and destination addresses with entries in the firewall filter. This will result in higher effectiveness. The shorter the address prefix length, the greater the number of distinct α flow identifiers that will match each source-destination address prefix in the firewall filter, leading to a larger number of afflicted-flow packets being directed to the same virtual queue as the α-flow packets. On the other hand, if the address prefix length is short, a larger number of newly arriving α flows' source and destination addresses will match prefixes in the firewall filter, which will result in higher effectiveness. Data transfer nodes (DTNs), deployed in server clusters, are typically assigned addresses from the same IP subnet; if an α flow is observed from one DTN and its /24 subnet ID is used in the firewall filter, then a subsequent α flow created from another DTN will have addresses that match the previously created /24 prefix based firewall filter rule, and the flow will hence be isolated. Prior work [31] already observed the tradeoff between effectiveness and AFPP, but in this work, the differences in the extent of this tradeoff at the additional three routers were compared.

For the edge routers, for the particular data sets analyzed, the best combination of high effectiveness and low AFPP was observed to be an aging parameter of 30 days and an address prefix length of /24. In general, an operational HNTES can be configured to continuously monitor its performance, and adjust parameter values to improve performance as network traffic patterns change.

(iii) For the core/REN-peering router and commercial-peering router, the HNTES effectiveness metric was lower than for the edge routers. The obtained NetFlow records were also from the incoming side of externally facing interfaces, which means that the flows corresponded to file uploads to DOE national laboratory data transfer nodes. Through further analysis of other variables, such as the number of α NetFlow records, we concluded that uploads were fewer than downloads, which is consistent with our understanding of how the scientific community uses the high-performance computing systems housed in the DOE national laboratories.

Our findings have shown that our hypothesis is valid.

1.5.2 α flow characterization

While size/rate characterization for all flow types is challenging because of the low packet sampling rates offered by built-in router features such as NetFlow, our work offers a solution for characterizing size and average rate for α flows. Our validation approach of using operational data from two disparate sources (GridFTP usage logs from file-transfer application servers, and NetFlow records from ESnet routers) was challenging to execute because of privacy considerations, but it demonstrates the feasibility of validating proposed solutions in an operational context rather than on an experimental testbed.

Key findings: (i) In spite of low packet sampling rates, the size, duration, and rate of α flows can be accurately estimated from NetFlow records.

(ii) By executing the size-duration computation procedure on NetFlow records gathered from four ESnet routers over a 7-month period, we found flow sizes as large as 811 GB and average rates as high as 5.7 Gbps (backbone link rate in ESnet4 was 10 Gbps).

(iii) A comparison of flow characteristics at different types of routers showed that there were more α flows in the download direction from DOE labs than in the upload direction to DOE labs.

(iv) To study persistency, we determined the number of flows created by each source-destination IP address pair. The maximum number of flows that exceeded 5 GB in size and 100 Mbps in rate, for a single source-destination pair was 1596, of which 75% experienced less than 167 Mbps while the highest rate was 536 Mbps. Such information is useful for initiating diagnostics to improve performance.

Our findings have shown that our hypothesis is valid.

1.6 Thesis Organization

The rest of the thesis is organized into four chapters.

Chapter 2 describes related work, which is addressed in two parts. First, an overview of HNTES is provided along with terminology that is reused in our evaluation of HNTES. Next, publications by other researchers related to our work are reviewed.

Chapter 3 presents our detailed evaluation of HNTES based on NetFlow records collected from four ESnet routers. Explanations are provided for the observed differences in the effectiveness and AFPP metrics corresponding to the four routers.

Chapter 4 presents our algorithm for flow reconstruction from NetFlow records. The algorithm was validated using a set of GridFTP logs collected from operational data transfer nodes. The results of applying the algorithm to the 7-month NetFlow records collected from four ESnet routers are presented, and causes for the observed differences are discussed.

Chapter 5 concludes the thesis and identifies future work items.

Chapter 2

Related Work

In Section 2.1, we provide an overview of the HNTES architecture after defining the terminology. In Section 2.2, related work by other researchers is reviewed.

2.1 HNTES Overview

2.1.1 Terminology

A NetFlow record r is represented as

$$\{\omega_r, f_r, l_r, v_r, o_r\} \tag{2.1}$$

where ω_r is the (5-tuple) flow identifier, f_r is the Coordinated Universal Time (UTC) timestamp of the first packet in the record, l_r is the UTC timestamp of the last packet in the record, v_r is the number of packets in the record, and o_r is the cumulative number of octets (bytes) in the record.

The flow identifier ω_r is defined as

$$\omega_r \triangleq \{s_r, d_r, p_r, q_r, y_r\} \tag{2.2}$$

where s_r: source IP address, d_r: destination IP address, p_r: source port number, q_r: destination port number, y_r: protocol type.

If the *active timeout interval* is configured to be τ_{max}, for all NetFlow records

$$0 \leq l_r - f_r \leq \tau_{max} \tag{2.3}$$

α NetFlow record: A NetFlow record r is said to be an α *NetFlow record* if:

$$o_r \geq H \tag{2.4}$$

where H is a size threshold.

α flow: Any flow that has at least one α NetFlow record is classified as an α *flow*.

2.1.2 HNTES Overview

In prior work [30], Z. Yan et al. proposed a hybrid network traffic-engineering system (*HNTES*) for α-flow identification and isolation of future α flows from general-purpose flows. Since the setup phase in virtual-circuit (VC) networking allows for path selection, REN providers, such as ESnet, Internet2, JGN-X, GEANT2, and others, have deployed a dynamic VC service to complement their basic IP-routed service [4]. As α flows require high rates, the use of VCs would allow the circuit scheduler (called an Inter-Domain Controller (IDC) [12]) to choose a less-utilized path. The term "hybrid network" in the name HNTES thus denotes a network with both virtual-circuit and IP-routed services.

HNTES is a network management software system that is deployed on an *external* server. It communicates with the routers, IDC, and NetFlow collector within its own network (as illustrated in Fig. 2.1 [29, 31]). Its functions are described below.

α-flow address prefix identification: Periodically HNTES obtains NetFlow records from the NetFlow collector, and analyzes these records to identify the source and destination IP address prefixes of α flows.

For each α NetFlow record r, the tuple consisting of source and destination IP address prefixes $\{s'_r, d'_r\}$ corresponding to $\{s_r, d_r\}$ (see definition (2.2)) is referred to as the flow's α *prefix ID*. As-

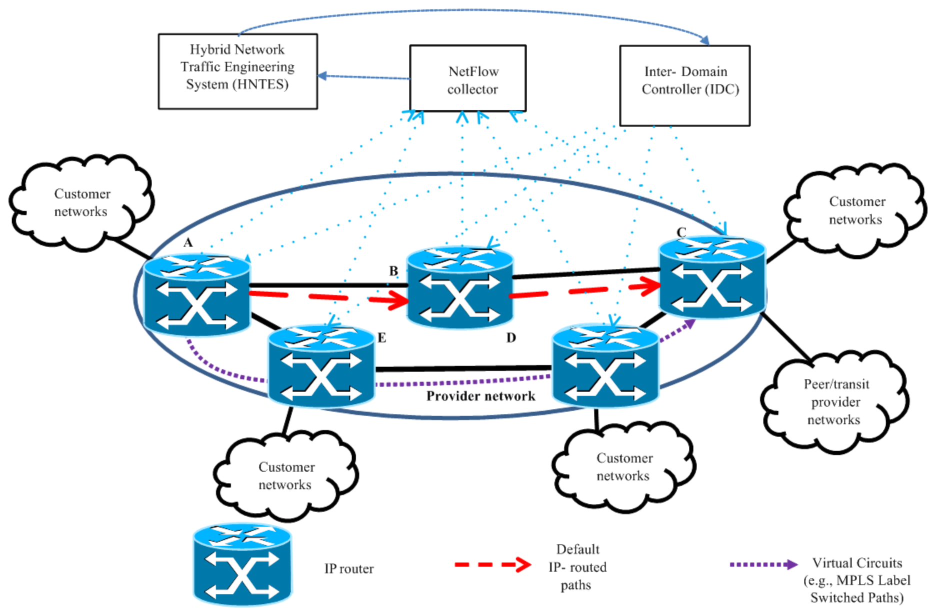

Figure 2.1: Illustration of the role of Hybrid Network Traffic Engineering System (HNTES) [29,31]

suming that HNTES runs on a nightly basis, it creates a list of α *prefix IDs* to store in a set $\mathbf{F}_i$, where i is a per-day index.

Configuring routers for future α-flow redirection: The source-destination IP address prefix pairs $\{s', d'\}$ in $\mathbf{F}_i$ are used to set firewall filter rules at each ingress router to separate out packets from future α flows and redirect them to traffic-engineered, QoS (Quality of Service)-controlled virtual circuits. While the REN virtual-circuit services are being developed for inter-domain usage, adoption by providers is proceeding slowly. Therefore, HNTES is currently designed to use only intra-domain circuits. The technological solution of carrying IP packets over Multiprotocol Label Switching (MPLS) label switched paths (LSPs) for segments of an end-to-end path is leveraged by HNTES. On each day i, HNTES determines the egress router E corresponding to each new destination d' in $\mathbf{F}_i$, and sends requests to the IDC for an LSP, if one does not already exist. The IDC executes three steps: (i) sets up the LSP between ingress router I and egress router E, (ii) configures QoS mechanisms such as weighted fair queuing (WFQ) scheduling [32], and (iii) configures a rule in the firewall filter at router I to identify packets corresponding to $\{s', d'\}$ and direct them to the virtual queue served by the MPLS LSP. If an LSP already exists between I and E corresponding to a new $\{s', d'\}$ entry in $\mathbf{F}_i$, HNTES communicates directly with the routers to accomplish the actions of steps (ii) and (iii).

Incoming flows on day i whose source and destination addresses match one of the α prefix IDs in the firewall filter $\mathbf{F}_i$ will be automatically classified as α flows by the router and directed to the virtual queue for the corresponding MPLS LSP. Thus if α flows are repeatedly created between the same source-destination hosts/subnets, then the HNTES solution will be highly effective in isolating α flows from other flows. To prevent the firewall filter from growing too large, an aging parameter A (e.g., 30 days) is used to delete rules corresponding to which no flows have been observed. Thus, HNTES changes the set $\mathbf{F}_i$ on a daily basis.

In *summary*, the HNTES design uses an *offline* approach, in which α prefix IDs are determined through a posteriori analysis, in contrast to an *online* approach in which α flows would be identified from a live analysis of ongoing traffic.

2.2 Related Work

Terms such as “elephant” flows have been used to characterize large-sized flows by other researchers [1, 11, 19, 28], while the term “α flows” was introduced by Sarvotham et al. [25]. Definitions of elephant or α flows differ in these papers based on their objectives. Papagiannaki et al. [19] discussed the potential use of their techniques for identifying elephant flows in traffic engineering applications.

General methods for traffic classification include port and payload based techniques, both of which have limitations (port numbers are ephemeral and payload based techniques are hindered by encryption) [17]. General machine learning techniques for traffic classification are of interest in the research community [20, 23, 24]. These techniques are more complex but have broad applicability.

In contrast, our proposed technique for HNTES works for large scientific data transfers as the servers/clusters used for such transfers have static public IP addresses.

There are several papers proposing methods for identifying large flows or high-rate flows with new router hardware. These include ElephantTrap [14], RATE [10], CATE [7], an FPGA-based cache solution [33], and a Grid flow real-time detector for 1 Gbps links [18]. Also Hohn and

Veitch [8] proposed a scheme for finding the spectral density, distribution of the number of packets per flow, and showed why alternate sampling techniques were need to obtain this second-order statistic about flows. Given our focus on designing network management systems and not new router hardware, our scheme relies on the built-in NetFlow system supported in most deployed provider routers.

Kamiyama and Mori propose a short-timeout method to identify high-rate flows [9] and elephant (large) flows [15] with low false-positive and false-negative rates, but not to determine the flow rates or sizes. Zhang, Fang and Zhang [34] proposed a Bayesian single sampling method to identify high-rate flows, but again not to characterize their sizes/rates.

Duffield, Lund and Thorup [3] had a goal of finding information about flows in unsampled packets using information in sampled packets.

In contrast, our goal is more specific to characterizing α flows. Given the higher rate of sampling of these flows, our method of characterizing α flows will result in higher accuracy but is not as general in its scope [3].

The impact of packet sampling on traffic classification and characterization was studied in [2, 26].

Chapter 3

Evaluation of HNTES

3.1 Introduction

This chapter extends the prior [30, 31] evaluation of HNTES in the following ways:

1. The prior work [30,31] evaluated HNTES performance using NetFlow records collected from only one router, which was an edge router, while our work evaluated HNTES performance using NetFlow records collected from three other routers. We did not modify the method developed in prior work [30,31] for determining the set of source-destination address prefixes to include in the router firewall filters.

2. The prior work [30, 31] defined effectiveness as a per-month metric, which we replaced with two new metrics: (i) daily effectiveness (we expect HNTES to be configured to execute its analysis programs once per day), and (ii) cumulative effectiveness. For afflicted-flows, in addition to AFPP, we computed a second metric, daily total number of afflicted-flow packets. Further, we characterized several traffic-related variables such as daily number of α NetFlow records, total number of α prefix IDs, and total number of days when no α flow appeared. These characterizations were used to explain the differences in the effectiveness and AFPP metrics corresponding to the four routers.

3. To compute the new metrics, we implemented new analysis programs in Java (prior work was coded in R), and parallelized the programs to run them on UVA's research computing cluster

(`fir` [5]) (while the R program took 3 days to analyze one month's data our Java program took just a few hours).

4. We provided explanations for the results obtained from the analysis. First, we recognized that the NetFlow records collected at the core/REN-peering and commercial-peering routers were for file uploads to DOE laboratories, while the NetFlow records collected at the two edge routers were for file downloads from DOE laboratories. Second, the daily number of α NetFlow records showed that there were fewer uploads than downloads. Also, the number of source/destination pairs that engaged in high-rate large-sized uploads to DOE laboratories were fewer than the number engaged in downloads. These findings offered an explanation for why the history-based HNTES approach was less successful (the effectiveness metric was lower) for routers at which uploads were observed than for routers at which downloads were observed.

The following sections provide a description of the routers from which NetFlow records were collected, define HNTES performance metrics, and present results.

3.2 Obtaining NetFlow records for evaluation

To evaluate HNTES, we obtained NetFlow records from four ESnet routers for a 7-month period (May-Nov. 2011, a period of 214 days), and analyzed these records. The four routers were carefully selected to represent different roles as shown in Fig. 3.1. `Router-1` and `router-2` are provider-edge (PE) routers located in ESnet customers' sites, and hence connected to a single customer (DOE national laboratory) network each. `Router-3` is a core router connected to multiple ESnet PE routers, and multiple national and international REN peers, such as Internet2 and AARnet. While the REN peers connect to ESnet at some of its other core routers, the ESnet PE routers connected to `router-3` are not connected to any other ESnet routers. Thus, all packets from/to the set of customer networks connected to `router-3` that are not destined to/sourced from networks within that set pass through `router-3`. `Router-4` is one of several ESnet routers used for commercial peering.

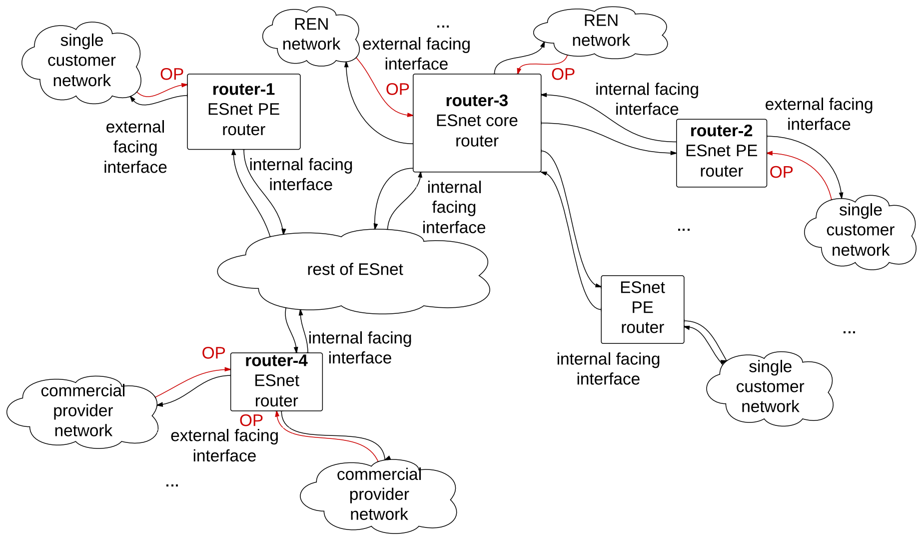

Figure 3.1: NetFlow records were obtained from Observation Points (OP) for four ESnet routers, **`router-1`**, **`router-2`**, **`router-3`**, **`router-4`**

Our NetFlow observation points (OP), as shown in Fig. 3.1, include only the input side of external-facing (inter-domain) interfaces to avoid double counting flows. For example, α flows in which files are being transferred *from* the customer network connected to `router-1` will be identified from NetFlow records collected at `router-1`, while α flows in which files are being transferred *to* the customer network connected to `router-1` will be identified from NetFlow records collected on the input-side of inter-domain links at other routers (e.g., OPs at the other three routers in Fig. 3.1).

The values of τ_{max} (see definition (2.3)) and H (see definition (2.4)) from this collected NetFlow records set are 60 sec and 1 GB, respectively. NetFlow is configured for 1-in-1000 packet sampling in ESnet routers.

3.3 Effectiveness Analysis

3.3.1 Methodology

Let $\mathbf{A}_i$ be the set of α NetFlow records on day i ($1 \leq i \leq 214$), and O_i be the total number of bytes reported in α NetFlow records (α *bytes*) on day i:

$$O_i = \sum_{\forall r \in \mathbf{A}_i} o_r. \tag{3.1}$$

Flows whose source and destination addresses have corresponding entries in the firewall filter $\mathbf{F}_i$ on day i will be automatically isolated by the routers as described in Section 2.1. Therefore, the total number of bytes that would have been redirected on day i, denoted by $\tilde{O}_i$, is given by:

$$\tilde{O}_i = \sum_{\forall r \in \mathbf{A}_i \wedge \{s'_r, d'_r\} \in \mathbf{F}_i} o_r. \tag{3.2}$$

Two types of *effectiveness* are evaluated:

$$\textit{Cumulative effectiveness } C_i = \frac{\sum_{k=1}^{i} \tilde{O}_k}{\sum_{k=1}^{i} O_k}, \tag{3.3}$$

$$\textit{Daily effectiveness } E_i = \frac{\tilde{O}_i}{O_i}, \tag{3.4}$$

when $O_i \neq 0$; if $O_i = 0$, E_i is said to be "not applicable." The goal of HNTES is to achieve high effectiveness so that few, if any, α flows will get routed to the same virtual queue as general-purpose flows.

3.3.2 Results – Impact of aging parameter

Both effectiveness and the size of the firewall filter are dependent on the value of aging parameter A. Therefore, we first characterize the effect of different values of A on these measures. Fig. 3.2 shows the growth in the size of the firewall filter at `router-1` for four values of the aging parameter (in the ∞ setting, firewall filter rules would not be aged out). Firewall filters should be kept small for operational reasons, and also because some routers have small size limits for such filters. Graphs for the other 3 routers are similar in that past day 100, the size of the firewall filter is almost stable

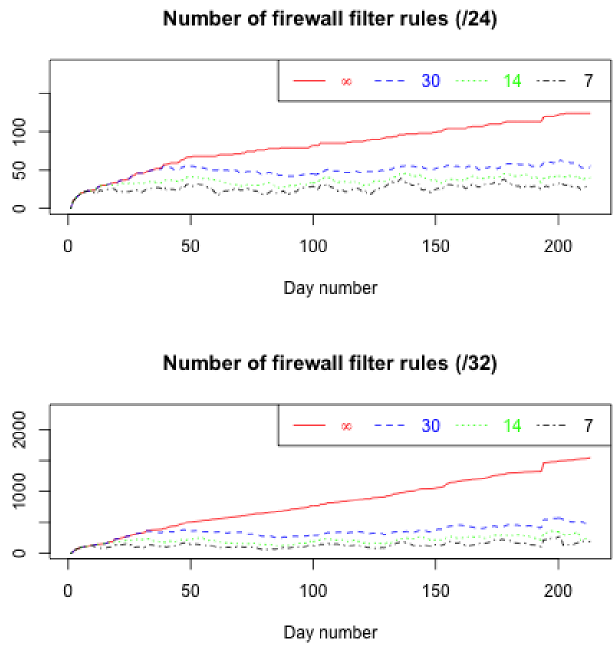

Figure 3.2: Growth of firewall filter in `router-1` for four values of the aging parameter in days

when the aging parameter is 30 or smaller (see the low coefficient-of-variation (cv) values in the first three rows of Table 3.1).

Fig. 3.3 compares the cumulative effectiveness for `router-1` under the same four aging parameter values for the /24 address prefix case as in Fig. 3.2. With an aging parameter of 30 days, cumulative effectiveness values are close to the best-case values when rules are never aged out. Similar results are observed for the other 3 routers. As a value of $A = 30$ days offers a good tradeoff between high effectiveness and firewall filter size, this value is assumed in the analysis presented in the following sections.

3.3.3 Results – Effectiveness comparison

Row 4 of Table 3.1 shows the cumulative effectiveness for each router for /24 and /32 address prefixes. For all routers, *this measure is higher for /24 address prefixes*. This is because clusters in the same /24 subnet are often used for data transfers, which means that an α flow from a new host (i.e., one from which there were no previously observed α flows) will be redirected with /24 prefix

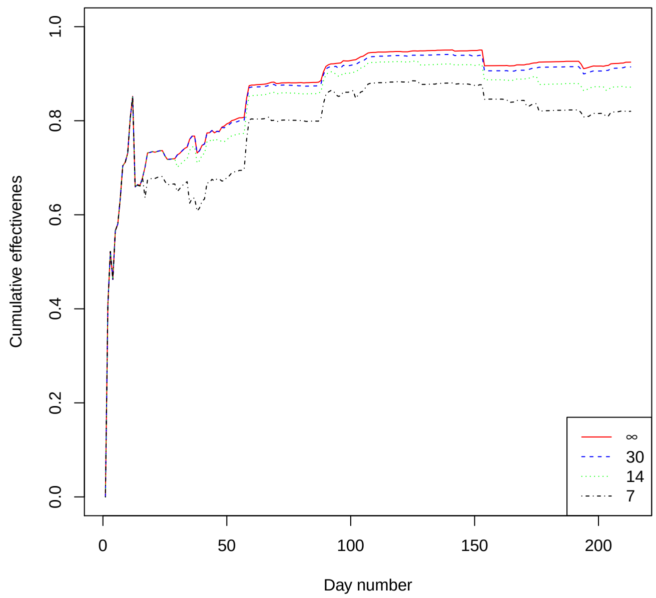

Figure 3.3: Cumulative effectiveness for the /24 prefix case at `router-1` for four values of the aging parameter in days

based firewall filter rules, but not with /32 based rules.

Table 3.1: Rows 1 − 3: across values from day 100 to day 214; Rows 4 − 8: across the whole 214-day period; The aging parameter A value is assumed to be 30 days (rows 7 and 8 are unaffected by the aging parameter)

Row	Statistics		router-1		router-2		router-3		router-4	
			/24	/32	/24	/32	/24	/32	/24	/32
1	Size of firewall filter	max	63	572	120	969	34	63	41	74
2		mean	53.41	406.77	91.63	384.32	24.63	48.82	29.36	8.4
3		cv	0.08	0.18	0.18	0.77	0.18	0.18	0.18	1.29
4	Cumulative effectiveness, C_{214}		91%	82%	92%	83%	83%	76%	67%	50%
5	# of days when $E_i = 1$		90	3	49	21	104	72	86	60
6	# of days when $E_i = 0$		1	5	2	4	12	23	25	51
7	# of days when no α flow appeared		1	1	0	0	21	21	35	35
8	total # of α prefix IDs		125	1548	281	1639	104	228	117	239

Row 4 of Table 3.1 also shows that the *effectiveness values are lower for* `router-3` and `router-4` when compared to the PE routers, `router-1` and `router-2`. For an explanation, consider the following observations made from the results in Rows 4-8 of Table 3.1, Table 3.2, Table 3.3,

Fig. 3.4, and Fig. 3.5:

Table 3.2: Results when firewall filter entries are not aged out

	`router-3`		`router-4`	
	/24	/32	/24	/32
Cumulative Effectiveness, C_{214}	87%	80%	72%	53%
# of days when $E_i = 1$	117	80	99	64
# of days when $E_i = 0$	8	15	22	42

Table 3.3: Number of per-day α NetFlow records

	`router`			
	`1`	`2`	`3`	`4`
Min	0	2	0	0
1st Qu.	27	140.2	8	1
Median	68.5	371.5	23.5	3
Mean	188.2	619.7	97.7	4.5
3rd Qu.	195	823.8	106	5.75
Max	2337	7345	1411	62

1. The high cumulative effectiveness for the PE routers, `router-1` and `router-2`, for the /24 prefix, shown in Row 4 of Table 3.1 is supported by Fig. 3.4, Fig. 3.5, and Row 5 of Table 3.1. Fig. 3.4 shows that the `router-1` daily effectiveness value is 1 on many days (quantified as 90 days in Row 5), which means that a significant fraction of α flows would have been identified and directed to the appropriate virtual circuits because of firewall filter entries. This is consistent with Fig. 3.5, which shows that daily effectiveness, $E_i > 90\%$ for more than 150 days for `router-1` and more than 130 days even for `router-3`.

2. The lower cumulativeness effectiveness for `router-3` and `router-4` in Row 4 of Table 3.1 is supported by the higher number of days when $E_i = 0$ for these routers as seen in Row 6 of Table 3.1, and the larger (0,0.1) bar for router-3 in Fig. 3.5. The numbers presented in Table 3.2 suggest that a larger aging parameter at `router-3` and `router-4` can be used to improve effectiveness. Given the fairly small firewall-filter sizes for these routers seen in

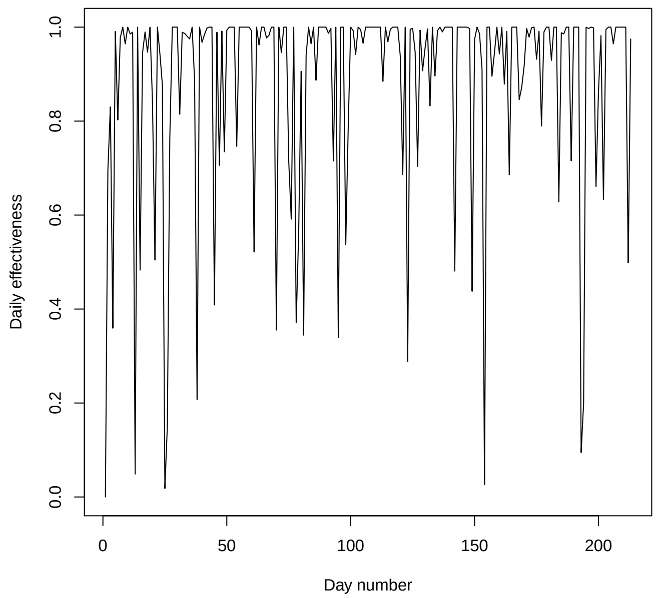

Figure 3.4: Daily effectiveness for `router-1` with /24 prefixes and $A = 30$

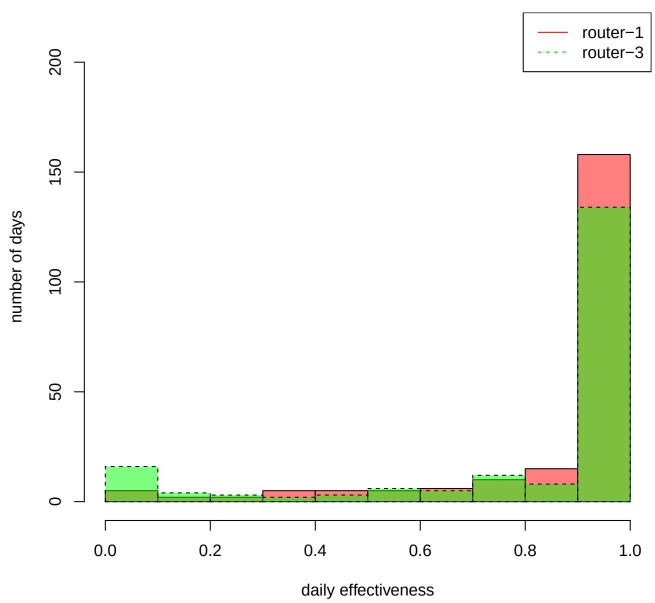

Figure 3.5: Histogram of E_i across the 214-day period when A is 30 (/24); view electronically for colors

Row 1 of Table 3.1, higher number of days in which α flows were not observed at `router-3` and `router-4` (see Row 7 of Table 3.1), and the lower number of α NetFlow records as seen in Table 3.3 (a maximum value of only 62 at `router-4`), the firewall filter size should be acceptable even at higher values of the aging parameter.

3. There are fewer α prefix IDs (Row 8 of Table 3.1) but larger number of days when $E_i = 1$ at `router-3` and `router-4` than at the PE routers for /32 addresses.

4. For /24 addresses, the number of α prefix IDs is lower for `router-3` than for `router-2` (see Row 8 of Table 3.1), even though the latter is one of the PE routers connected to the former.

3.3.4 Explanations for observations

Observation 1: The PE routers are connected to ESnet customer sites that house supercomputing facilities on which scientists run their applications and generate datasets. As scientists repeatedly use these facilities, α flows occur between the same source-destination pairs. A firewall filter rule created with an address prefix pair observed on one day is repeatedly able to redirect packets from future α flows.

Observation 2: The lower number of α NetFlow records at `router-3` and `router-4` are because there are fewer uploads of large datasets than downloads from ESnet customer sites. Since these sites are DOE national laboratories with the supercomputing centers, more α flows are likely to be downloads from ESnet customer site servers rather than uploads. Recall the observation points shown in Fig. 3.1 from which the NetFlow records are collected for each router. As NetFlow records are collected for the input-side of the interface connecting each PE router to its customer network, α flows generated by downloads from ESnet customer sites will be identified in the `router-1` and `router-2` records. In contrast, since the observation points for `router-3` and `router-4` are on the input-side of interfaces from RENs and commercial peers, only uploads made to ESnet customer sites will appear as α flows in these NetFlow records, and as there are likely to be fewer of these uploads, we see fewer α NetFlow records at `router-3` and `router-4`.

Given the lower frequency of uploads, HNTES effectiveness is lower since repeated α flows are not observed between the same source-destination pairs. Table 3.2 shows that there were 22 days when α flows appeared between two *new* /24 subnets at `router-4`. There is one other possible explanation for the lower effectiveness at `router-3` and `router-4`. As these routers have higher loads than the PE routers, 1-in-1000 NetFlow packet sampling rate may have led to missed α-flow packets.

Observation 3: It appears that fewer *servers* are used in uploads to DOE laboratories than in downloads, which explains the higher number of days when $E_i = 1$ for /32 addresses at `router-3` and `router-4` than at the PE routers.

Observation 4: Given the connectivity of `router-2` to `router-3`, as shown in Fig. 3.1, we expected a larger number of α prefix IDs at `router-3` than at `router-2`. However, the numbers are reversed, with 281 α prefix IDs observed at `router-2` for the /24 case, which is more than double the number (104) observed at `router-3`. Our explanation for observation 2 that the number of downloads are greater than the number of uploads is likely the reason for this observation too.

A *conclusion* from this analysis is that given the higher effectiveness rates of HNTES for NetFlow records collected at PE routers, NetFlow records could be obtained for both directions of external-facing interfaces at PE routers. Since ESnet does not offer transit service, all α flows are sourced from or destined to ESnet customer sites, and therefore locating observation points at just these routers is sufficient for complete coverage. Given the lower traffic loads at the PE routers when compared to core routers, it is more likely that packets from a majority of α flows will be sampled at these routers than at core routers through which α flows from/to multiple sites traverse.

3.4 Afflicted-flow Characterization

Section 3.3 illustrated that the effectiveness metric is higher with /24 address prefixes. However, the negative aspect of this choice is that β (non-α) flows whose source and destination addresses are within the address ranges of the prefixes stored in the firewall filter $\mathbf{F}_i$ will be directed to the α-flow virtual queues. Packets from these β flows could then be subject to increased delays and jitter. Since flows from interactive applications are sensitive to delay/jitter, the subset of β flows generated by non-file-transfer applications are referred to as "afflicted flows." The /24 and /32 choices are compared on measures related to afflicted-flow packets.

In this section, we determine the percentage of afflicted-flow packets in samples of β-flow packets.

3.4.1 Methodology

On any given day i, set $\mathbf{A}_i$ represents the set of α NetFlow records as defined in Section 3.3. A set $\mathbf{P}_i$ of α prefix IDs for day i is defined to include address prefixes of all α flows observed in set $\mathbf{A}_i$. Then a set $\mathbf{B}_i$ of *non-*α *NetFlow records* (denoted by all NetFlow records that do not cross the H-bytes threshold in (2.4)) is extracted for day i such that $\forall r \in \mathbf{B}_i, o_r < H, \{s'_r, d'_r\} \in \mathbf{P}_i$. Packets from flows represented by NetFlow records in set $\mathbf{B}_i$ form a sample of packets that would be directed to the α-flow virtual queue because they unfortunately share α prefix IDs. As assumption is made here that all prefix IDs in set $\mathbf{P}_i$ are in the firewall filter (a fair assumption for most days as seen in Fig. 3.4).

Towards identifying the percentage of non-file-transfer (non-FT) flow packets within set $\mathbf{B}_i$, we apply three steps in sequence. First, we extract out NetFlow records corresponding to α flows identified by set $\mathbf{A}_i$. Next, we find the set of NetFlow records from file transfers using a heuristic. Third, we separate out NetFlow records from connections with well-known port numbers. These steps are applied in sequence to distinguish flows from `scp`, a file transfer application that uses the ssh well-known port number (some of these flows could fall in the first α-flow category or second non-α flow file transfer category) from interactive ssh flows, such as those from a remote terminal application such as SecureCRT (third category). Flows from the third category and the leftover NetFlow records are the ones considered to be "afflicted."

NetFlow records in sets $\mathbf{B}_i$, $1 \leq i \leq 214$, are classified into four groups:

- $\mathbf{C}_i$, set of records from α flows: $r \in \mathbf{C}_i$ iff there is a record $r' \in \mathbf{A}_i$ such that $s_r = s_{r'}$, $d_r = d_{r'}$, $p_r = p_{r'}$, $q_r = q_{r'}$, and $y_r = y_{r'}$ (see Section 2.1 for notation).

- $\mathbf{D}_i$, set of records from other file transfers: $r \in \mathbf{D}_i$ iff $r \in \mathbf{B}_i - \mathbf{C}_i$, $o_r/v_r > 1000$ bytes, $o_r > G$ where $G < H$, and there exists another record $r' \in \mathbf{B}_i - \mathbf{C}_i$ such that $s_r = s_{r'}$, $d_r = d_{r'}$, $p_r = p_{r'}$, $q_r = q_{r'}$, $y_r = y_{r'}$, $o_{r'}/v_{r'} > 1000$ and $o_{r'} > G$. Observations have shown that flow records that meet these criteria are typically from file-transfer applications.

- $\mathbf{W}_i$, set of non-FT NetFlow records with well-known port numbers: $r \in \mathbf{B}_i - \mathbf{C}_i - \mathbf{D}_i$, iff p_r or q_r is one of several well-known port numbers (ssh, http, imap, smtp, ssmtp, https, nntp, imaps, imap4ssl, unidata, rtsp, rsync, sftp, bftp, ftps, pop3 and sslpop)

- $\mathbf{L}_i$, set of leftover NetFlow records, which is $\mathbf{B}_i - \mathbf{C}_i - \mathbf{D}_i - \mathbf{W}_i$

Let $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$, $\mathbf{W}$, and $\mathbf{L}$ be the aggregate set of the corresponding per-day sets, e.g., $\mathbf{B} = \bigcup_{1 \le i \le 214} \mathbf{B}_i$. Flows corresponding to the NetFlow records in set $\mathbf{W} + \mathbf{L}$ are considered to be afflicted flows.

The two metrics for afflicted-flow analysis are as follows: the daily number of packets in NetFlow records in set $\mathbf{W} + \mathbf{L}$, and *afflicted-flow packets percentage*, which is given by

$$AFPP_i = \frac{\sum_{k=1}^{i} \sum_{\forall r \in (\mathbf{W}_k \cup \mathbf{L}_k)} v_r}{\sum_{k=1}^{i} \sum_{\forall r \in (\mathbf{D}_k \cup \mathbf{W}_k \cup \mathbf{L}_k)} v_r}, 1 \le i \le 214. \tag{3.5}$$

Unlike in the effectiveness analysis where bytes were used, here packets are used because the estimation of bytes with the multiplier factor of 1000 is less accurate with non-α flows (recall the 1-in-1000 NetFlow packet sampling rate).

3.4.2 Results

Fig. 3.6 shows the daily number of afflicted-flow packets in $\mathbf{W} + \mathbf{L}$ in `router-1`, when G is set to 10MB. Similar graphs are observed for the three other routers. On this measure, /32 address prefixes in firewall filters enjoys an advantage over /24 address prefixes because of the former's higher specificity. This contrasts with the advantage enjoyed by /24 address prefixes over /32 prefixes in the effectiveness measure.

Table 3.4 shows the second metric, afflicted-flow packets percentage, over the 214-day period. These percentages are not significantly high even for /24 address prefixes. Furthermore, considering

Table 3.4: Percentage of afflicted-flow packets, $AFPP_{214}$

	router			
	1	2	3	4
/24	10.39%	23.84%	6.22%	25.37%
/32	11.22%	13.18%	3.43%	25.51%

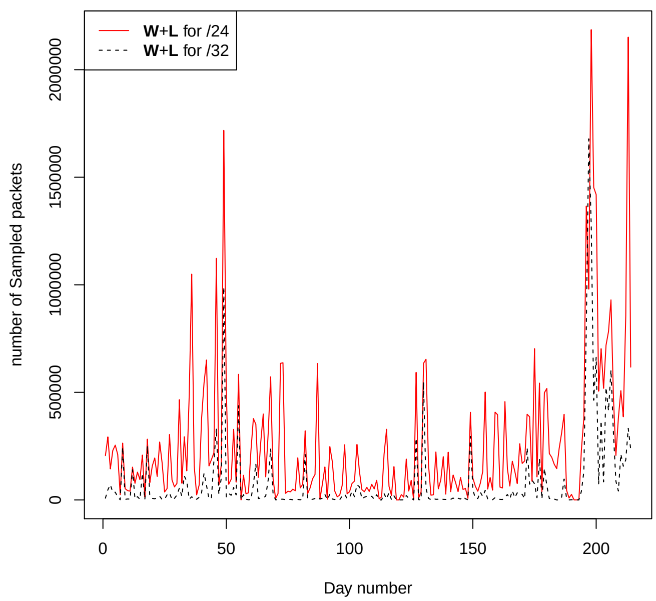

Figure 3.6: Number of packets in $\mathbf{W}+\mathbf{L}$ from `router-1`'s records

that the number of non-α flows that do not share α prefix IDs is much higher than that of α flows, when the afflicted-flow packets are considered as a percentage of the total number of non-α-flow packets, the relative negative effect of using /24 prefixes is even lower.

We *conclude* therefore that the choice of /24 address prefixes for the firewall filter is better than /32. If /32 prefixes are used, then there is a higher probability that an α flow is sent to the virtual queue served by IP-routed service where it can negatively impact the delay/jitter of many more non-α flows. On the other hand, if /24 prefixes are used, then a small percentage of non-α flows are subject to the adverse effects of α flows by being directed to the α-flow virtual queue.

3.5 Conclusions

The key findings of the evaluation of HNTES are: (i) We found that HNTES effectiveness was higher than 90% if the NetFlow records used were from the edge routers. The samples were collected from the incoming side of externally facing interfaces. Each edge router was connected to only a single customer router, which means that observed α flows were mostly downloads from high-performance data transfer nodes (DTNs) located in the customer networks.

(ii) The HNTES metrics depend on two parameters: *aging parameter* and *address prefix length*. For the edge routers, for the particular data sets analyzed, the best combination of high effectiveness and low AFPP was observed to be an aging parameter of 30 days and an address prefix length of /24. In general, an operational HNTES can be configured to continuously monitor its performance, and adjust parameter values to improve performance as network traffic patterns change.

(iii) For the core/REN-peering router and commercial-peering router, the HNTES effectiveness metric was lower than for the edge routers. The obtained NetFlow records were also from the incoming side of externally facing interfaces, which means that the flows corresponded to file uploads to DOE national laboratory data transfer nodes. Through further analysis of other variables, such as the number of α NetFlow records, we concluded that uploads were fewer than downloads, which is consistent with our understanding of how the scientific community uses the high-performance computing systems housed in the DOE national laboratories.

Chapter 4

Characterization of α flows

4.1 Introduction

In this chapter, we describe an algorithm for characterizing the size, duration, average rate, and frequency of α flows from NetFlow records. The algorithm was validated using independently collected usage logs (GridFTP usage logs [6]) from application servers. We executed the algorithm on NetFlow records from 4 ESnet routers collected over a 7-month period. Flows moving datasets as large as 811 GB and at rates as high as 5.7 Gbps were observed. Some source-destination pairs were found to repeatedly create α flows. An analysis of the rates of the 1596 repeated α flows created by one pair showed considerable variance, with minimum rate of 100 Mbps, maximum rate of 536 Mbps, and a coefficient of variation of 30%.

4.2 Terminology

4.2.1 Flow

A *flow* is defined to consist of all packets arriving with the same 5-tuple values (see definition (2.2)) {source IP address, destination IP address, source port number, destination port number, protocol type} with no consecutive inter-packet gaps greater than some fixed time threshold τ. Inter-packet gaps within the period of a NetFlow record, which are not recorded, are necessarily smaller than the active timeout interval. Therefore, in order to reconstruct flows from NetFlow records, the fixed time threshold τ is set to be at least as large as the NetFlow active timeout interval. The five tuples constitute the *flow IDentifier (flow ID).*

The fixed time threshold phrase is required because a TCP connection can be held open for a long duration, but only carry packets in intermittent bursts. For example, with HTTP1.1, a TCP connection is held open while a Web client accesses a Web server. If the first downloaded Web page has multiple images located on the same Web server, then each of those images will be downloaded on the same TCP connection. Since the Web client software parses the HTML page and automatically sends out GET requests for the images, these inter-GET time gaps will be short. On the other hand, when human user input (e.g., a mouse click) is required to generate GET requests, there could be large "think-time" gaps.

Multiple sets of GET request bursts (consisting of GET requests generated automatically by the Web client), and their responses, could thus occur on the same TCP connection, and will hence share a flow ID. But packets related to each such set is likely be parsed out as a separate flow given the time threshold in our definition of a *flow*. Effectively, if the time gap between the last-packet timestamp in one NetFlow record r, and the first-packet timestamp in the next NetFlow record with the same flow ID exceeds the threshold τ, then a flow is said to have terminated with NetFlow record r, and a new flow started with the next NetFlow record.

4.2.2 α NetFlow records and β NetFlow records

As described in section 2.1, a NetFlow record r is said to be an α NetFlow record if $o_r \geq H$, where H is a size threshold. We define β NetFlow records as *non-α NetFlow records*.

4.2.3 α flow and γ flow

In Chapter 3, we used the term "α flow" to represent any flow that had at least one α NetFlow record (as defined in Section 2.1). In this chapter, we use the term γ flow to characterize any flow that has at least one α NetFlow record. We redefine the α flow term to be a γ flow whose size and rate exceed specified thresholds. This change was made because the new algorithm, presented in this chapter, allows us to compute the total size and total duration (from which average rate can be determined) of each α flow. Since α flows were defined informally in Chapter 1 to be large-sized, high-rate flows, this new algorithm allows us to provide a more formal definition of "α flow" with specified thresholds for size and rate. Therefore, we coined the new term γ flow to characterize flows that have at least one α NetFlow record, and redefined the term α flow.

4.2.4 Other notation

Other notation used in this chapter is presented in Table 4.1.

Table 4.1: Notation

i	per-day index
j	flow-identifier (ID) index
k	γ-flow index
r	NetFlow-record index
$\mathbf{F}_i$	set of NetFlow records
$\mathbf{A}_i$	set of α NetFlow records (size $> H$)
$\mathbf{W}_i$	set of unique flow IDs ω_r for records $r \in \mathbf{A}_i$
$\mathbf{B}_i$	set of β NetFlow records r whose flow IDs $\omega_r \in \mathbf{W}_i$
$\mathbf{C}_{ij}$	set of NetFlow records r, s.t. $\omega_r = j$, for $j \in \mathbf{W}_i$
$\mathbf{E}_{ijk}$	Subset of $\mathbf{C}_{ij}$: records of a single γ flow
N_{ij}	Number of γ flows
S_{ijk}	Size of γ flow
D_{ijk}	Duration of γ flow
ρ	packet sampling rate (e.g., 1/1000)

4.3 Algorithm of reconstructing flows from NetFlow records

We developed an algorithm for combining information from multiple NetFlow records to determine the size, duration, and average rate of α flows. Using the notation in Table 4.1, the main steps of the algorithm are listed below:

1. From each day's set of NetFlow records, $\mathbf{F}_i$, determine sets $\mathbf{A}_i$, $\mathbf{W}_i$, and $\mathbf{B}_i$ using the size threshold H.

2. For each day i, the set $\mathbf{A}_i \bigcup \mathbf{B}_i$ is divided into disjoint subsets, $\mathbf{C}_{ij}$, $1 \leq j \leq |\mathbf{W}_i|$.

3. Order the records in each set $\mathbf{C}_{ij}$ by sorting on the first-packet timestamp (earliest-to-latest). The ordered set of records are $r_1, r_2, \cdots, r_{|\mathbf{C}_{ij}|}$.

4. Divide each set $\mathbf{C}_{ij}$ into disjoint subsets $\mathbf{E}_{ijk}$, $1 \leq k \leq N_{ij}$ such that a consecutive set of NetFlow records $\{r_n, r_{n+1}, \cdots, r_{n+u}\} \in \mathbf{E}_{ijk}$ iff

$$\begin{aligned} f_{r_{m+1}} - l_{r_m} &\leq \tau \qquad n \leq m < n+u \\ f_{r_n} - l_{r_{n-1}} &> \tau \qquad \text{for } n \neq 1 \\ f_{r_{n+u+1}} - l_{r_{n+u}} &> \tau \qquad \text{for } n+u \neq \left|\mathbf{C}_{ij}\right| \end{aligned} \tag{4.1}$$

5. A γ flow $k, 1 \leq k \leq N_{ij}$, appearing on day i with flow-ID $\omega_j \in \mathbf{W}_i$, and consisting of NetFlow records $\{r_n \cdots, r_{n+u}\} \in \mathbf{E}_{ijk}$, is characterized by

$$\begin{aligned} \text{Size } S_{ijk} &= \left(\frac{1}{\rho}\right) \sum_{m \in \mathbf{E}_{ijk}} o_m \\ \text{Duration } D_{ijk} &= l_{r_{n+u}} - f_{r_n} \\ \text{Av. rate } R_{ijk} &= \frac{S_{ijk}}{D_{ijk}} \end{aligned} \tag{4.2}$$

Starting with each day's set of NetFlow records ($\mathbf{F}_i$), the *first step* is to find the subset of α NetFlow records ($\mathbf{A}_i$), from which the set of unique flow IDs ($\mathbf{W}_i$) is extracted. Using these flow IDs, a second pass through set $\mathbf{F}_i$ is executed to find all β NetFlow records (set $\mathbf{B}_i$) for the γ flows observed on day i. The goal of this first step is to reduce the number of NetFlow records from which to extract α flows.

The *second step* creates sets $\mathbf{C}_{ij}$ consisting of all the α and β NetFlow records corresponding to each γ flow ID j. Since these $\mathbf{C}_{ij}$ sets are extracted from the disjoint sets of α ($\mathbf{A}_i$) and β ($\mathbf{B}_i$) NetFlow records, the records in each $\mathbf{C}_{ij}$ need to be sorted by the first-packet timestamp before flows can be reconstructed. This is the *third step*.

The *fourth step* is to divide the NetFlow records in each set $\mathbf{C}_{ij}$ into multiple subsets, each of which consists of a set of consecutive records belonging to a single γ flow. Recall from Section 4.2, that if a time gap threshold is exceeded between the last-packet timestamp l_r of one NetFlow record r and the first-packet timestamp f_{r+1} of the next NetFlow record $(r+1)$, the flow is considered to have terminated with record r, and a new flow begun with the next record. There is potential for a small gap between l_r and f_{r+1} for two consecutive records r and $(r+1)$ because of packet sampling. Therefore, as long as this gap is less than a time-threshold τ, the consecutive NetFlow records are considered to belong to the same flow. Using k as the index for γ flows, the subsets of $\mathbf{C}_{ij}$ are denoted $\mathbf{E}_{ijk}$, all of which share the same flow ID j in their appearance on day i (see Table 4.1).

The *final step* is to add up the bytes in the NetFlow records of each γ flow to determine the size of the flow and multiply by the reciprocal of the packet sampling rate ρ.

Duration is computed by finding the time difference between the last-packet timestamp of the last NetFlow record and the first-packet timestamp of the first NetFlow record in each set $\mathbf{E}_{ijk}$. Average rate is computed by dividing flow size by flow duration.

As an example, consider the NetFlow records shown in Table 4.2. The first two columns show the number of packets, and cumulative number of bytes, in the sampled packets of the NetFlow record. The next five columns, source and destination IP addresses, source and destination transport-layer port numbers, and protocol type field, constitute the flow ID ω (see 2.1). The source and destination IP addresses were anonymized and hence the numbers shown in Table 4.2 are not in the expected 4-byte format. The timestamps (TS) are in UTC format. For example, the first-packet TS of the first NetFlow record is 1304269790.137; UTC time 1304269790 corresponds to Sun, 01 May 2011 17:09:50 GMT [27]. The last three digits 137 corresponds to milliseconds. In this example, τ was set to 60 sec. The gap between the last-packet TS of the first NetFlow record and the first-packet TS of the next NetFlow record is 889.798 sec; as this gap is greater than τ (1 min), the second NetFlow record of Table 4.2 represents the start of a new flow. This flow had $(95+6=101)$ sequential NetFlow records with inter-record gaps less than τ. For example, the gap between the first two records of the 101-record flow is only 180 ms. Similarly, the gap between the last-packet TS of the last record of the 101-record flow and the first-packet TS of the last record in Table 4.2 is 40665.873 sec, which is well above τ.

Table 4.2: Example NetFlow records observed for one γ flow ID in one day; TS: Timestamp; dur: duration (sec)

pkts	bytes	src IP	dst IP	src port	dst port	prot.	first-pkt TS	last-pkt TS	dur (sec)
Previous flow's last NetFlow record									
481	683020	6853	6840	20886	62362	6	1304269790.137	1304269820.122	29.98
Next flow (has 101 NetFlow records)									
173	245660	6853	6840	20886	62362	6	1304270709.920	1304270749.856	39.93
251	356420	6853	6840	20886	62362	6	1304270750.036	1304270809.975	59.93
247	350740	6853	6840	20886	62362	6	1304270810.282	1304270869.675	59.39
There were 95 other NetFlow records with inter-record gaps less than τ									
230	326600	6853	6840	20886	62362	6	1304276573.971	1304276633.668	59.69
234	332280	6853	6840	20886	62362	6	1304276634.016	1304276693.903	59.88
61	86620	6853	6840	20886	62362	6	1304276694.116	1304276704.044	9.92
Next flow's first NetFlow record									
57	80940	6853	6840	20886	62362	6	1304317369.174	1304317391.838	22.66

4.4 Validation of the algorithm

4.4.1 Method

To validate the algorithm presented in Section 4.3, we devised the following method using operational, not experimental, datasets.

Step 1: Obtain GridFTP usage logs [6] from an operational data transfer node: GridFTP usage logs were obtained from dedicated data transfer nodes at the National Energy Research Scientific Computing (NERSC) center for the period, Apr. 22 to June 30, 2012. The usage logs include the following information for each transfer: remote end's IP address, size in bytes, start time of the transfer, and transfer duration.

Step 2: Find corresponding NetFlow records from an ESnet router: Next, since NERSC is a customer of ESnet, and ESnet has located one of its routers at NERSC, i.e., a provider-edge (PE) router, we obtained NetFlow records from this PE router for the same time period. For each GridFTP usage log entry, using the source and destination IP addresses and the start and end time of the corresponding transfer, our software finds matching NetFlow records.

Step 3: Find additional NetFlow records with the same flow IDs: Using the unique 5-tuple flow IDs from the per-day set of matched NetFlow records obtained in Step 2, a second pass was executed to find all NetFlow records corresponding to these 5-tuple flow IDs even if the time intervals of these records (first-packet TS, last-packet TS) were outside any GridFTP-transfer time intervals. These NetFlow records were required to determine whether our size/rate estimation algorithm could correctly identify the GridFTP transfers as single flows.

Step 4: Characterize flows: From the sets of NetFlow records found in steps 2 and 3, we executed the algorithm described in Section 4.3 to characterize γ flows.

Step 5: Recreate "sessions" from GridFTP transfer logs: The prior analysis [13] showed that most GridFTP transfers occur in sessions, i.e., multiple file transfers on the same TCP connection. The `-fast` option of GridFTP when invoked to move files in a directory will result in all files being transferred on the same TCP connection. The GridFTP sending process sends multiple files concurrently. All transfers to the same destination with overlapping durations are included in a single session. A gap value of up to 10 ms was allowed when grouping transfers into sessions. Also, the log entry shows the number of parallel TCP streams used for a transfer (which is set by users with the `-p` option). Since large datasets are typically moved using the `-p` option, we included only those transfers that used more than 1 parallel TCP stream. All transfers within each session had the same number of parallel TCP streams.

Step 6: Accuracy computation: For each GridFTP session that exceeded size and rate thresholds (5 GB and 667 Mbps), we found multiple γ flows whose start and end times fell within the GridFTP session duration. There were multiple γ flows because of the use of parallel TCP streams. The γ-flow sizes were added to find the total size before comparing with the GridFTP session size. The average duration across all the γ flows corresponding to a GridFTP session was determined and compared with the GridFTP session duration. Size (duration) accuracy is defined as the ratio of the size (duration) estimated by our algorithm from the NetFlow records to the size (duration) reported in the GridFTP usage logs.

4.4.2 Results

Table 4.3 shows the results of our validation procedure. Both duration accuracy and size accuracy for these high-rate large-sized flows were close to 100%. Size accuracy can be greater than

Table 4.3: Results of algorithm validation using GriFTP logs

No.	Log dur. (s)	Est. dur. (s)	D-acc (%)	Log size (GB)	Est. size (GB)	S-acc (%)
1	195.3	194.2	99.4	52.4	51.9	99.0
2	158.9	156.2	98.3	34.4	33.2	96.7
3	190.2	187.7	98.7	34.4	34.3	99.9
4	157.8	155.4	98.5	34.4	35	101.7
5	6516	6466.3	99.2	6.2	6.6	105.5
6	7696.8	7695.8	99.9	6.2	6.3	101.3
7	73.94	72	97.4	5.8	6.1	105.5

100% because the NetFlow packet sampling process could have caught more packets of a particular transfer than 1-in-1000.

4.5 Characterization of α flows observed in ESnet Traffic

The set of NetFlow records from four ESnet routers collected over a 7-month time period, May-Nov. 2011 used for the HNTES evaluation was reused in this work to characterize γ and α flows. After presenting the results generated by applying our algorithm to the NetFlow data in Section 4.5.1, the implications of these findings are discussed in Section 4.5.2.

4.5.1 Results

Four sets of results are presented:

1. aggregate characteristics of γ flows and α flows
2. statistics about three characteristics: size, rate, and duration, of γ flows and α flows
3. number of α flows as a function of the size and rate thresholds, and
4. persistency measure: number of γ flows and α flows created between the same source and destination.

Table 4.4: Aggregate data on γ and α flows; across 214 days

	Router-1	Router-2	Router-3	Router-4
No. of γ flows	28685	27963	2516	212
No. of unique γ flow IDs	19365	26939	2455	212
No. of unique /32 src-dst pairs gen. γ flows	1479	1611	193	158
Max. no. of per-day γ flows corr. to a single γ flow ID	33	56	6	1
No. of α flows	916	9538	986	16
No. of unique α flow IDs	834	9043	943	16
No. of unique /32 src-dst pairs gen. α flows	95	419	89	14

Table 4.5: Size in MB of γ flows; across 214 days

	Router-1	Router-2	Router-3	Router-4
Min	1001	1001	1005	1010
1st Qu.	1149	1540	4050	1203
Median	1275	2869	4360	1532
Mean	2513	9046	17540	3612
3rd Qu	1701	8768	21380	3772
90%	2761	16600	54115	5774
99%	12909	92012	104356	26389
99.9%	229727	288797	180138	100460
Max	633300	811600	233600	112800
CV	5.20	2.56	1.40	2.43
skewness	25.35	12.56	2.37	10.09

Aggregate characteristics of γ flows (H was set to 1 GB) and α flows (using a size threshold of 5 GB and rate threshold of 100 Mbps) at each of the routers across the observation period of 214 days are listed in Table 4.4. The second row shows the number of unique γ flow IDs observed, while the third row lists the number of unique source-destination pairs that generated γ flows, in the 214-day period. The fourth row represents the maximum number of per-day γ flows corresponding to a single γ flow ID. Multiple γ flows could have resulted from a TCP connection being held open

Table 4.6: Rate in Mbps of γ flows; across 214 days

	`Router-1`	`Router-2`	`Router-3`	`Router-4`
Min	11.7	3.6	34.6	49.2
1st Qu.	160.9	147	117.6	130.9
Median	199.3	181.9	132.6	156.4
Mean	245.2	230.9	159	182.7
3rd Qu.	258.9	252.1	159.2	195.8
90%	403	363	264	275
99%	881	944	503	649
99.9%	1711	993	953	755
Max	5154	5757	979	776
CV	0.71	0.72	0.56	0.61
skewness	7.36	3.95	3.82	2.86

Table 4.7: Duration in sec of γ flows; across 214 days

	`Router-1`	`Router-2`	`Router-3`	`Router-4`
Min	4.2	8.0	9.5	12
1st Qu.	41.8	60.9	190.9	54.9
Median	54.2	121.1	272	94.3
Mean	122.8	414.2	1098	235.6
3rd Qu.	73.6	398.9	1169	227.6
90%	118.5	977.2	3655.7	349.3
99%	639.9	3942.1	6183.3	1460.3
99.9%	17055.9	11751.4	12854.4	8697.6
Max	32460	31910	13940	9978
CV	7.39	2.34	1.50	3.18
skewness	23.76	10.33	2.32	10.99

for a long duration with gaps between flows as explained in Section 4.2. The last three rows present aggregate information about α flows.

Statistics for three characteristics of γ flows: size, rate, and duration, are presented in Tables 4.5, 4.6, and 4.7. These tables are independent, e.g., the largest-sized flow is not the same as the highest-rate flow.

Table 4.8 presents results from a sensitivity analysis of the number of α flows to the size and rate thresholds.

Table 4.8: Sensitivity to size-rate threshold: No. of α flows

size	rate	`Router-1`	`Router-2`	`Router-3`	`Router-4`
5GB	200Mbps	496	4475	201	3
10GB	100Mbps	526	5460	726	3
10GB	150Mbps	399	4121	297	1
10GB	180Mbps	375	3037	124	0
10GB	200Mbps	357	2443	92	0
50GB	200Mbps	19	505	28	0
80GB	500Mbps	0	20	0	0

Finally, we characterized the persistency with which source-destination pairs generated γ flows and α flows. Figs. 4.1 and 4.2 plot the cumulative distribution function (CDF) of the numbers of γ flows and α flows per source/destination pair for `router-2`, `router-3` and `router-4`. The plots for `router-1` have been omitted because they overlapped significantly with those of `router-2`. Recall that `router-1` and `router-2` are PE routers that capture flows corresponding to downloads from DOE labs, and hence have similar numbers of flows.

4.5.2 Discussion of the results

The results presented in the previous section are discussed below in three groupings. *First*, we discuss the numerical values themselves to understand the range of sizes, rates, durations, and frequencies, of γ flows and α flows. *Next*, we compare the characteristics of flows observed at the different routers. *Finally*, an example application is described to demonstrate usage of this characterization of α flows.

Numerical values:

The difference between the number of γ flows, and number of unique γ flow IDs (rows 1 and 2 in Table 4.4) occurs because of two possibilities: the same 5-tuple values were used on two different days, or a given flow ID was reused in multiple flows within the same day. The latter is characterized in the fourth row. Most γ flow IDs have only single γ flows in a given day, but there are a few occasions when multiple γ flows have been observed on the same day for a given γ flow

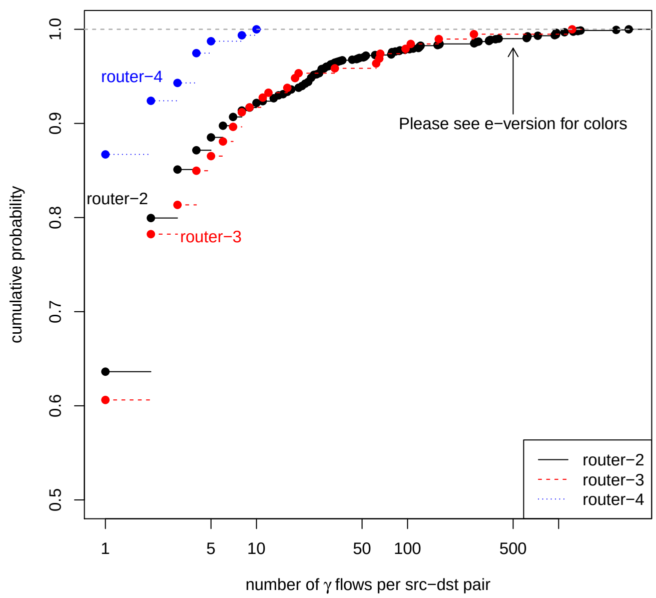

Figure 4.1: CDF of number of γ flows per src/dst pair across 214 days for `router-2`, `router-3`, `router-4` (`router-1` plot overlaps closely with the `router-2` plot and is hence omitted)

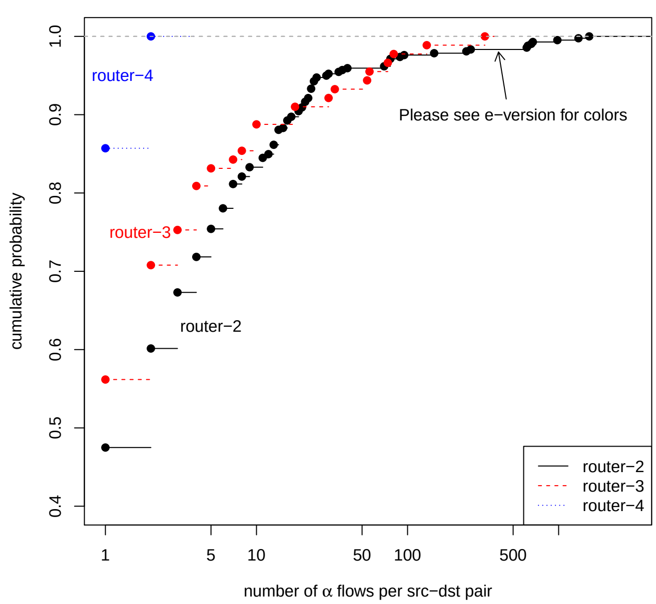

Figure 4.2: CDF of number of α flows (> 5 GB, > 100 Mbps) per src/dst pair across 214 days for `router-2`, `router-3`, `router-4`

ID. As many as 56 γ flows were observed for a single five-tuple ID in one day (at `router-2`) as shown in Table 4.4.

Across the 214-day period, of all the flows observed at the four routers, the largest-sized flow was 811.6 GB (max row of Table 4.5) and the highest-rate flow enjoyed an aggregate rate of 5.76 Gbps (max row of Table 4.6), both of which were downloads passing through PE router `router-2`. The largest-sized flow had a rate of 301 Mbps, and the fastest flow size was 7.14 GB. The longest flow lasted 32460 sec (more than 9 hours) passing through `router-1`, during which time 370 GB was moved (max row of Table 4.7).

At the lower end, rates as low as 3.6 Mbps were observed, also at `router-2`. This particular γ flow moved 1.9 GB, which means it lasted about 4181 sec (more than an hour).

Since there is a significant gap between the 3^{rd} quartile values, and the maximum values, Tables 4.5 and 4.6 show a few more quantiles in the fourth quarter. Using the number of γ flows provided in Table 4.4, we see that the 99.9% value of 229.73 GB implies that only 28 flows in the size range (229.73 GB, 633.3 GB) entered `router-1` from its connected DOE lab. Similarly, the 99.9% rate value for γ flows passing through `router-2` was still less than 1 Gbps (even though the maximum rate for this router was 5.76 Gbps). This implies that only 27 flows out of the 27963 observed γ flows (flows larger than 1 GB with a rate $>$ 133 Mbps) enjoyed (average) rates higher than 1 Gbps during the 7-month period.

Skewness is defined as μ_3/σ^3, where μ_3 is the third moment and σ is the standard deviation. The coefficient of variation (CV) and skewness values were lower for rates than for sizes, as seen in Tables 4.5 and 4.6. This was expected since file sizes have heavy-tailed distributions [21].

Table 4.8 shows that the number of α flows falls quickly as the size-rate threshold is increased, which is to be expected. Nevertheless, the absolute numbers are interesting to note. Router `router-2` connects ESnet to a supercomputing center, which explains that even at the high per-flow thresholds of 80 GB and 500 Mbps, 20 α flows were observed.

Comparison between flows observed at different routers:

As seen in Table 4.4, there were many more γ flows in downloads from DOE labs than uploads to DOE labs (since downloads were observed at `router-1` and `router-2`, while uploads were observed at `router-3` and `router-4`). Also, more source-destination pairs engaged in transfers larger than 1 GB for downloads than uploads.

As seen in Tables 4.5 and 4.6, γ flows for downloads from DOE labs were larger in size and higher in rate. Uploads to DOE labs, observed at `router-3` and `router-4` were considerably slower, with the maximum rate reaching only 776 Mbps at the commercial peering router `router-4` and only 979 Mbps at the REN-peering router `router-3`. Maximum flow sizes were also smaller. Table 4.7 shows that the longest downloads were longer than the longest uploads, but most γ flows are short in duration.

A comparison of the number of α flows across the 4 routers from Table 4.8 shows a difference between the two PE routers. While `router-1` is a PE router connected to large national DOE lab, the significant research projects at this lab are in a single science discipline. In contrast, PE router `router-2` connects to a national scientific supercomputing center that is used by scientists from many disciplines. This explains the larger numbers of α flows for `router-2` when compared to `router-1` as seen in Table 4.8.

Finally, Figs. 4.1 and 4.2 show that uploads through the commercial peering router `router-4` were considerably fewer (maximum values of 10 γ flows and 2 α flows) than through the other routers. A comparison of the red (`router-3`) and black (`router-2`) plots shows the former plots ending before the latter plots. The maximum number of γ-flow and α-flow uploads per source-destination pair for `router-3` were 1229 and 325, respectively, while at `router-2`, the numbers for γ-flow and α-flow downloads per source-destination pair were 2913 and 1596, respectively. The maximum γ-flow and α-flow downloads per source-destination pair at `router-1` were 2860 and 445, respectively. The ninety percentile numbers for γ flows per source-destination pair were 39, 7, 7.8 and 2 for the four routers in sequence, and the numbers for α flows per source-destination pair were 11.6, 19, 18 and 1.7. Therefore, less than 10% of the source-destination pairs generated large numbers of repeated γ flows and α flows, which makes it somewhat easier for operators to provide better services (higher rates, lower variance) for these particular source-destination pairs.

Example application:

Consider the source-destination pair that generated the largest numbers of γ flows and also the largest number of α flows across the 214-day period. The particular source-destination IP address pair that generated these maximum number of flows was (2888,7128) using the anonymized addresses[1]. Since all these flows were between the same source and destination, and there were no network upgrades during the data-collection period, the bottleneck link rate and round-trip time were approximately the same, and all flow sizes are greater than 1 GB, which means TCP's Slow Start period could not have had a major influence on the average rate. Nevertheless, in the 2913 γ-flow set, 75% of the flows experienced less than 161.2 Mbps while the highest rate experienced was 1.1 Gbps (size: 3.5 GB). Similarly, in the 1596 α-flow set, 75% of the flows experienced less than 167 Mbps, while the highest rate experienced was 536 Mbps (size: 11 GB). Such information would allow the provider to initiate diagnostics to determine the causes of lower rates.

4.6 Conclusions

This work demonstrated that it is feasible to determine the size, duration, and rate, of high-rate, large-sized (α) flows from NetFlow records in spite of low packet sampling rates, e.g., 1-in-1000. The algorithm proposed here can form the basis of a network management system for characterizing α flows. Example applications include special traffic-engineering of α flows (since they have the potential to degrade service quality of real-time flows), offering users who generate α flows diagnostic support to determine causes of low throughput or high throughput variance, and identifying BGP misconfigurations that cause α flows to enter a provider's network on a less-preferred route. The algorithm was validated using independently collected usage logs from application servers. We executed our algorithm on actual NetFlow records from 4 ESnet routers collected over a 7-month period. Individual flows moving datasets as large as 811 GB and at rates as high as 5.7 Gbps were observed. Some source-destination pairs were found to repeatedly create α flows. An analysis of the rates of the 1596 repeated α flows created by one pair showed considerable variance, with minimum rate of 100 Mbps, maximum rate of 536 Mbps, and a coefficient of variation of 30%.

[1]For privacy reasons, the actual addresses are not published, but are stored in our data archives for retrieval if needed.

Chapter 5

Conclusions and Future Work

This thesis presented an evaluation of Hybrid Network Traffic Engineering System (HNTES): we compared HNTES performance when using NetFlow records collected at four ESnet routers, and offered explanations for observed differences. The results showed that HNTES effectiveness was above 90% for NetFlow records collected at edge routers, which corresponded to file downloads from DOE laboratories, while the effectiveness was lower for the peering routers whose NetFlow records corresponded to file uploads. With further investigation, we found that uploads were less frequent and involved fewer source/destination pairs than downloads.

The thesis also described an algorithm for characterizing the size, duration, average rate, and frequency of α flows from NetFlow records. The algorithm was validated using independently collected usage logs from application servers. We executed the algorithm on actual NetFlow records from 4 ESnet routers collected over a 7-month period. Individual flows moving datasets as large as 811 GB and at rates as high as 5.7 Gbps were observed. Some source-destination pairs were found to repeatedly create α flows. An analysis of the rates of the 1596 repeated α flows created by one pair showed considerable variance, with minimum rate of 100 Mbps, maximum rate of 536 Mbps, and a coefficient of variation of 30%.

The findings of the research work have shown that our hypotheses are valid.

Future work items on HNTES evaluation include finding explanations for why HNTES effectiveness was lower for peering routers than for edge routers, and using size instead of α bytes in evaluating HNTES. The lower effectiveness could have been caused because of higher loads at the peering routers (which would influence effectiveness because of the 1-in-1000 NetFlow packet sampling rate), or it could be because uploads were less frequent than downloads. This work requires new data collection/procurement from ESnet. Future work items on α flow characterization include the extension of the algorithm to aggregate information from NetFlow records corresponding to parallel TCP flows since GridFTP users often use this feature (a group of parallel flows may cause the same adverse effects as a single high-rate α flow, and hence need to be identified) and application of these algorithms to NetFlow records collected at other ESnet routers and other providers' routers.

Acknowledgements

This work was finished with the help of our ESnet collaborator, Chris Tracy. This work was supported by the U.S. Department of Energy (DOE) grant DE-SC0007341 and NSF grants OCI-1038058, OCI-1127340, and CNS-1116081. The ESnet portion of this work was supported by the Director, Office of Science, Office of Basic Energy Sciences, of the U.S. DOE under Contract No. DE-AC02- 05CH11231.

I want to take this opportunity to thank my advisor Prof. Malathi Veeraraghavan, without whose guidance I could not complete my research work or the thesis. I also want to thank my committee members, Prof. Jack Davidson, Prof. Kevin Sullivan and Prof. Alfred Weaver for taking the time to review my thesis. I would also love to thank my parents and friends for their unconditional support. Life would be meaningless without them.

I sincerely wish all these wonderful people health and happiness all the time.

Bibliography

[1] N. Brownlee and K.C. Claffy. Understanding Internet traffic streams: dragonflies and tortoises. *Communications Magazine, IEEE*, 40(10):110 – 117, oct 2002.

[2] Valentn Carela-Espaol, Pere Barlet-Ros, Albert Cabellos-Aparicio, and Josep Sol-Pareta. Analysis of the impact of sampling on NetFlow traffic classification. *Computer Networks*, 55(5):1083 – 1099, 2011.

[3] N. Duffield, C. Lund, and M. Thorup. Estimating flow distributions from sampled flow statistics. *IEEE/ACM Transactions on Networking*, 13(5):933–946, 2005.

[4] ESnet. Available at `http://www.es.net`.

[5] Getting Started on the ITC Linux Clusters. Available at `http://www.uvacse.virginia.edu/getting-started-on-the-itc-linux-clusters/`.

[6] GridFTP. Available at `http://globus.org/toolkit/docs/3.2/gridftp/`.

[7] Fang Hao, M. Kodialam, T. V. Lakshman, and Hui Zhang. Fast, memory-efficient traffic estimation by coincidence counting. In *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, volume 3, pages 2080–2090 vol. 3, 2005.

[8] N. Hohn and D. Veitch. Inverting sampled traffic. *IEEE/ACM Transactions on Networking*, 14(1):68–80, 2006.

[9] N. Kamiyama and T. Mori. Simple and accurate identification of high-rate flows by packet sampling. In *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, pages 1–13, 2006.

[10] M. Kodialam, T. V. Lakshman, and S. Mohanty. Runs based traffic estimator (rate): a simple, memory efficient scheme for per-flow rate estimation. In *INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 3, pages 1808–1818 vol.3, 2004.

[11] Kun-chan Lan and John Heidemann. A measurement study of correlations of internet flow characteristics. *Computer Networks*, 50(1):46–62, 2006.

[12] Andrew Lake, John Vollbrecht, Aaron Brown, Jason Zurawski, David Robertson, Mary Thompson, Chin Guok, Evangelos Chaniotakis, and Tom Lehman. Inter-domain Controller (IDC) Protocol Specification, May 2008.

[13] Z. Liu, M. Veeraraghavan, Z. Yan, C. Tracy, J. Tie, I. Foster, J. Dennis, J. Hick, Y. Li, and W. Yang. On using virtual circuits for GridFTP transfers. In *The International Conference for High Performance Computing, Networking, Storage and Analysis 2012 (SC 2012)*, pages 81:1–81:11, Nov. 10-16, 2012.

[14] Yi Lu, Mei Wang, B. Prabhakar, and F. Bonomi. ElephantTrap: A low cost device for identifying large flows. In *15th Annual IEEE Symposium on High-Performance Interconnects, 2007. HOTI 2007.*, pages 99–108, 2007.

[15] Tatsuya Mori, Masato Uchida, Ryoichi Kawahara, Jianping Pan, and Shigeki Goto. Identifying elephant flows through periodically sampled packets. In *Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*, IMC ’04, pages 115–120, New York, NY, USA, 2004. ACM.

[16] NetFlow. Available at `http://www.cisco.com/en/US/products/ps6601/products_ios_protocol_group_home.html`.

[17] Thuy T. T. Nguyen and Grenville J. Armitage. A survey of techniques for internet traffic classification using machine learning. *IEEE Communications Surveys and Tutorials*, 10(4):56–76, 2008.

[18] J. Paisley and J. Sventek. Real-time detection of grid bulk transfer traffic. In *Network Operations and Management Symposium, 2006. NOMS 2006. 10th IEEE/IFIP*, pages 66 –72, april 2006.

[19] Konstantina Papagiannaki, Nina Taft, Supratik Bhattacharyya, Patrick Thiran, Kav Salamatian, and Christophe Diot. A pragmatic definition of elephants in Internet backbone traffic. In *IMW '02 Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment*, pages 175–176, 2002.

[20] Junghun Park, Hsiao-Rong Tyan, and C-CJ Kuo. Internet traffic classification for scalable qos provision. In *Multimedia and Expo, 2006 IEEE International Conference on*, pages 1221–1224. IEEE, 2006.

[21] Vern Paxson and Sally Floyd. Wide-area traffic: the failure of Poisson modeling. *IEEE/ACM Transaction on Networking*, 3:226–244, 1995.

[22] perfSONAR. Available at `http://www.perfsonar.net/`.

[23] Dario Rossi and Silvio Valenti. Fine-grained traffic classification with netflow data. In *Proceedings of the 6th International Wireless Communications and Mobile Computing Conference*, pages 479–483. ACM, 2010.

[24] Matthew Roughan, Subhabrata Sen, Oliver Spatscheck, and Nick Duffield. Class-of-service mapping for qos: a statistical signature-based approach to ip traffic classification. In *Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*, pages 135–148. ACM, 2004.

[25] Shriram Sarvotham, Rudolf Riedi, and Richard Baraniuk. Connection-level analysis and modeling of nework traffic. In *ACM SIGCOMM Internet Measurement Workshop 2001*, pages 99–104, November 2001.

[26] Davide Tammaro, Silvio Valenti, Dario Rossi, and Antonio Pescap. Exploiting packet-sampling measurements for traffic characterization and classification. *International Journal of Network Management*, 22(6):451–476, 2012.

[27] Epoch & Unix Timestamp Conversion Tools. Available at `http://www.epochconverter.com`.

[28] Jorg Wallerich, Holger Dreger, Anja Feldmann, Balachander Krishnamurthy, and Walter Willinger. A Methodology for Studying Persistency Aspects of Internet Flows. *ACM SIGCOMM Communication Review*, 35(2), 2005.

[29] Zhenzhen Yan. Traffic Engineering in Packet/Circuit Hybrid Networks, Dec 2013. PhD dissertation.

[30] Zhenzhen Yan, C. Tracy, and M. Veeraraghavan. A hybrid network traffic engineering system. In *High Performance Switching and Routing (HPSR), 2012 IEEE 13th International Conference on*, pages 141 –146, june 2012.

[31] Zhenzhen Yan, Chris Tracy, Malathi Veeraraghavan, Tian Jin, and Zhengyang Liu. In-network identification of scientific data transfer flows for traffic engineering, 2013. Submitted to Journal of Network and Systems Management.

[32] Zhenzhen Yan, Malathi Veeraraghavan, Chris Tracy, and Chin Guok. On how to provision Quality of Service (QoS) for large dataset transfers. In *Proceedings of the Sixth International Conference on Communication Theory, Reliability, and Quality of Service (CTRQ)*, April 21-26 2013.

[33] M. Zadnik, M. Canini, A.W. Moore, D.J. Miller, and Wei Li. Tracking elephant flows in internet backbone traffic with an fpga-based cache. In *Field Programmable Logic and Ap-*

plications, 2009. FPL 2009. International Conference on, pages 640 –644, 31 2009-sept. 2 2009.

[34] Yu Zhang, Binxing Fang, and Yongzheng Zhang. Identifying high-rate flows based on bayesian single sampling. In *2010 2nd International Conference on Computer Engineering and Technology (ICCET)*, volume 1, pages V1–370–V1–374, 2010.