MULTI-LEVEL APPLICATION-CENTRIC PROFILING OF UO INTERNET TRAFFIC

by

NATHAN KOGA

A THESIS

Presented to the Department of Computer Science
in partial fulfillment of the requirements for the degree of
Bachelor of Science

June 2024

An Abstract of the Thesis of

Nathan Koga for the degree of Bachelor of Science
in the Department of Computer Science to be taken June 2024

Title: Multi-level Application-centric Profiling of UO Internet Traffic

Approved: *Dr. Reza Rejaie*
Primary Thesis Advisor

Characterizing different aspects of exchanged traffic between an organization and the Internet provides valuable insights for the organization to determine how network resources are utilized and help identify potential malicious activity and performance bottlenecks. However, the huge volume and complexity of Internet traffic make such a profiling effort inherently challenging, as identifying an important event or pattern is essentially akin to finding a needle in a haystack.

In this thesis, we profile multiple aspects of exchanged traffic between the UO campus and the Internet using flow-level traffic data. Our main goal is to efficiently identify and summarize some of the key flow-level features of UO traffic that represent normal/typical behavior. This, in turn, enables us to quickly determine whether a single flow or an aggregate group of flows (e.g., all flows associated with a particular application) exhibits any abnormal behavior. To this end, our profiling follows a top-down approach in characterizing UO traffic by starting from aggregate analysis, classifying flows into main categories, and then “zooming into” main categories to gain more insight into each group. This strategy enables us to define a signature at each level for each category of flows.

We present the results of our multi-level profiling of UO traffic and investigate whether meaningful/stable signatures for distinguishing normal and abnormal behavior at each level can be identified.

Acknowledgements

I would like to thank my mentors Chris Misa and Professor Reza Rejaie for all their help and guidance in enabling and enhancing this research. I have learned a lot about a field that was originally unknown to me, and I am grateful to the ONRG team for being willing to allow me the opportunity to learn and grow through this experience.

Additionally, I am incredibly grateful for all the support and opportunities that have been provided to me through the University of Oregon Computer Science Department, and the many peers and faculty members that I have met who have inspired me throughout my study of Computer Science.

Lastly, I would like to thank my family, especially my parents, who have supported me throughout my life in all my endeavors. Their guidance and example have brought me to where I am today.

This work was supported, in part, by a grant from the National Science Foundation (NSF) Research Experiences for Undergraduates (REU) program.

Table of Contents

List of Figures

List of Tables

Chapter 1: Introduction

In recent years, the Internet has come to play an increasingly important role in modern life, continuously evolving and offering new services and opportunities for people worldwide. To put a number to this reality, Cisco's Annual Internet Report of 2023 said, "Globally, the total number of Internet users is projected to grow from 3.9 billion in 2018 to 5.3 billion by 2023" [1].

As the digital landscape evolves, computer networks have become increasingly complex as a result to satisfy the demands for more data from more users. As such, it is more important than ever for organizations to manage their networks to ensure efficient, secure, and reliable access to the Internet for users. Given the huge volume and variety of information that can be present in network traffic, it is important to utilize efficient and effective methods that are well-suited to handle this task. One such strategy that can help organizations in this effort is to employ network monitoring techniques to analyze the traffic patterns of a network. Through network monitoring, one can gain crucial visibility and insight into the characteristics of a given network, and from this, the standard modes of operation of a network can be defined. When presented with this context, an organization may use this information to influence future design decisions regarding its network. Additionally, they may reason that any abnormalities or deviations from understood patterns can indicate potential problems in the network. Such problems range from network mismanagement and performance bottlenecks, all the way to malicious attacks on the network.

Some main questions regarding the standard modes of operation of a network that may be answered through observing network traffic include the following:

- What are the typical traffic patterns of Internet traffic at each level?
- What transport protocols are most commonly used in our network?

- What are the common ports and identifiable applications that use our network?
- What are the characteristics of significant autonomous systems that interact with our network?
- Do these subsets of Internet traffic display recognizable characteristics that can define standard use?

To this end, this thesis utilizes unsampled NetFlow data collected by routers at the border between the University of Oregon (UO) campus network and the rest of the Internet. These routers, located at the edge of the university, offer a key vantage point for analyzing all incoming Internet traffic and all outgoing traffic originating from the university. In our thesis, we are primarily interested in characterizing how external entities interact with our network, therefore we focus our observation on incoming flows from the Internet. As such, we will filter through incoming NetFlow flows, analyzing snapshots taken over seven Wednesdays from the past year in the context of a few key flow-level profiling metrics to characterize incoming UO network traffic.

This investigation takes a top-down approach to profiling the university's network traffic at varying levels of depth. We begin by conducting an aggregate-level analysis of all incoming traffic from each chosen day to characterize the main observable features of incoming traffic as a whole. Then, we fine-tune our analysis by focusing on specific sub-categories of this traffic in hopes of defining signatures for each of these levels. This starts by zooming in on the most dominant transport protocol and delving deeper into specific ports representing that subset of web traffic. Finally, we focus on autonomous systems that communicate over these ports.

Prior studies in the field of computer networks have employed NetFlow to observe and monitor networks and systems, similarly, using aggregates of NetFlow data to observe the prevalence of specific protocols and applications or observe how external content providers interact with a network. While these studies utilize statistical analysis to make their conclusions, they often overlook the opportunity to characterize Internet traffic temporally. Furthermore, some investigations focus on analyzing Internet traffic at a single level of depth.

In this thesis, we aim to utilize statistical and temporal analysis to characterize Internet traffic, doing so in a top-down approach, zooming in at multiple levels of depth such that we can characterize Internet traffic at each resolution.

The rest of this thesis is organized as follows:

- In Chapter 2, we cover essential background information, including a detailed explanation of the dataset and methods used for data collection and processing.
- Chapter 3 presents the primary investigation, showcasing the findings and discussing key takeaways at each level of analysis.
- Chapter 4 reviews prior research in this field and highlights the contributions of our study to existing works.
- We conclude with Chapter 5, which provides a summary of our key findings and discusses their significance and discusses possible directions for future research.

Chapter 2: Dataset and Method

This section presents the necessary background for this thesis, describing data collection methods, datasets utilized, and techniques used for data processing.

2.1. NetFlow

In this thesis, we employ NetFlow as our means of network monitoring. NetFlow is a protocol developed by Cisco with its foundation based on flow exporting. A flow is defined as "a set of packets passing an Observation Point in the network during a certain time interval. A packet is defined as belonging to a flow if it completely satisfies all the defined properties of the Flow" [2]. For example, the "traditional *'5-tuple'* Flow Key" defines flows based on the following properties: source and destination IP addresses, source and destination transport ports, and transport protocol. As data packets are observed, they are aggregated into corresponding flows. Flows also accumulate additional metadata within their record, such as the type of service (ToS) of the flow, the timestamps of the first and last packets of that flow, and the total summation of bytes and packets delivered through that flow.

At the University of Oregon, our campus routers are NetFlow-enabled. This means that these routers take packets that pass through and group them into flows and aggregate this flow data throughout the day. The campus routers, after every five-minute window of the day, will take all flows seen within that window and export them using the *nfcapd* binary file format for the University to store and analyze in the future.

Even though these files are stored in binary, and are therefore very efficient and compact files, the sizes of these files are still quite substantial. Each 5-minute file takes up around 60~90 MB of storage during quiet hours, all the way up to 150 MB of storage during peak hours (depending on the router). Summing up both border routers, there is approximately 50 GB worth of NetFlow data to analyze per day, accounting for the billion-plus total flows that the university may observe on any given day.

Fortunately, the *nfdump* project is a powerful toolset with the capability to read and process these NetFlow records that have been exported to the *nfcapd* format. While *nfdump* offers a Command Line Interface (CLI) program that can take in sequences of NetFlow records and display information regarding queries for top statistics for numbers of flows, bytes, or packets seen in a certain time frame, they also offer a framework built using the C programming language for programmers to write their own procedures. Using this framework, we can process and filter through NetFlow records with very precise queries to extract whatever flow-level information we need for a given profiling metric. For example, we can plot a cumulative distribution function of the length of incoming flows observed by the UO network in a day.

2.2. Method

To analyze these datasets, we procedurally load and observe all flows found in each 5-minute snapshot during the desired time window. Characteristics that define each flow are compared to our profiling filters: relevant flows pass through our filters and are aggregated into the proper data structures as defined in our process, while all other flows that got filtered out are ignored.

In our investigation, we start by classifying flows as “incoming” (arriving at the UO network) or “outgoing” (leaving the UO network), such that we can partition and collect data by direction. Though many flows are bi-directional (where both parties communicate back and forth), the UO network creates separate flows for each direction — allowing us to filter accordingly.

For most flows, the “ToS” bitfield included in each flow record alongside the defining ‘5-tuple’ is enough to determine the direction of a flow. If the lowest order bit is active (i.e., the bitwise operation ToS & 1 returns true), the destination IP address is anonymized. If the second lowest order bit is active, then the source IP address is anonymized.

As determined by the UO network, the only anonymized IP addresses are internal IP addresses originating from the UO, save for a predefined list of 11 un-anonymized UO IP addresses. From this rule, we can determine flow direction with the following conditional logic:

- Incoming flows: The source IP address is un-anonymized and not found in the list of un-anonymized UO addresses, as well as the destination IP address is either anonymized or found in the list.
- Outgoing flows: The source IP address is anonymized or found in the list of un-anonymized UO addresses, as well as the destination IP address is non-anonymized and isn’t found in the list.

Any flows that don’t match these patterns are filtered out and not considered for our analysis. For example, at this highest level, we partitioned 514 million total flows read on 04/10/24 into 208 million incoming and 304 million outgoing flows.

After the direction of a flow has been determined, all other relevant attributes of that flow are used to determine whether we consider that flow for further analysis. In particular, the flow's transport protocol and source port, found from the ‘5-tuple’ flow key, are key attributes used to identify what group a flow belongs to.

The transport protocol of a flow indicates whether the flow is using Transmission Control Protocol (TCP), User Datagram Protocol (UDP), or another protocol as a means of communication. These transport protocols are well-suited to handle different types of data transfers, and it is important to understand the prevalence of these protocols in a network. TCP is a connection-based protocol that is a reliable but slower way to transport data, compared to UDP, which is a connectionless protocol that is faster but less reliable [3].

The port numbers associated with a flow are important to identify which specific service or application is used in a flow. A few well-known ports are assigned to specific application-layer protocols, such as port 443 for Hypertext Transfer Protocol Secure (HTTPS) and port 80 for Hypertext Transfer Protocol (HTTP). Port numbers can also be used to identify specific applications as well, such as Spotify (port 4070), or Apple Push Notification Services (port 5223).

Specifically with the University of Oregon routers, flows are additionally enriched with data that records the source and destination Autonomous System Number (ASN) of each flow. Each ASN is a unique number assigned to an Autonomous System (AS) and describes a collection of IPs that are owned by a single organization. Since flows at the UO are enriched with ASN information, specific content providers that communicate with the UO network can be identified. For instance, we know that flows with a source ASN of 16509 are related to Amazon.

Beyond these attributes, any additional information tracked with each flow record (such as the start and end times of a flow, or the total number of bytes and packets sent), further informs how each flow is handled and counted.

2.3. Dataset

This analysis focuses on the traffic observed by the UO campus border routers in the past school year, starting in the fall of 2023 up to the spring of 2024. As a baseline, flows are only initially counted (regardless of direction) if they counted more than three packets in their flow record. This decision was made to minimize the noise of the dataset by disregarding irrelevant traffic that may be recorded in these small flow records. Such irrelevant traffic includes failed connection attempts or network scanning packets. For example, On April 10th, 2024, without this initial filter, we counted 1.37 billion total flows (compared to 514 million) for only an increase of 0.2 terabytes of information. By adding this initial filter, our dataset is better set up to provide insight into the characteristics of these more meaningful network connections.

Snapshot	Unique Flows		Bytes		Unique Source IPs (Incoming)	
	Total	Incoming	Total	Incoming	Source	Destination
10/11/23	518.9 M	206.4 M	107 TB	78 TB	1.03 M	0.51 M
11/08/23	504 M	202.6 M	120.6 TB	89.4 TB	1.07 M	0.5 M
12/06/23	411.2 M	171.7 M	107.8 TB	76.8 TB	1.02 M	0.33 M
01/17/24	350.8 M	149.8 M	104.7 TB	76 TB	0.93 M	0.3 M
02/14/24	508.3 M	207.3 M	104.6 TB	76.2 TB	1.14 M	0.49 M
03/13/24	529.3 M	217.3 M	113 TB	81.1 TB	1.04 M	0.48 M
04/10/24	514.7 M	208.2 M	126.9 TB	92.5 TB	1 M	0.49 M

Table 1: Overview of Observed Flow-level Features over Selected Daily Snapshots

Table 1 summarizes the primary flow-level features observed from selected Wednesdays in the past year while school was in session. These features include the total and incoming number of flows and bytes, along with the total number of source and destination IP addresses observed for incoming connections. Since the focus of this investigation is to characterize how external entities from the Internet interact with our network, Table 1 explicitly includes the features of incoming flows. In each incoming flow, the source IP address belongs to an external entity, while the destination IP address is an internal IP address belonging to the UO network.

From these snapshots, we can see that the number of incoming flows accounts for roughly 41% of total daily flows and that the incoming number of bytes accounts for around 73% of the total daily volume of data seen by the border routers. Additionally, there are many more source IPs (from external entities) than destination IPs (in the UO network), at a ratio of around 2:1 up to 3:1.

2.4. Data Processing

When processing such large datasets, more consideration must be put into the time and spatial complexity of employed algorithms. While queries to calculate running sums over a given snapshot of traffic can be computed in linear time and with no additional space, queries that depend on the uniqueness of data are more computationally expensive than others and take much more memory to compute. Therefore, some queries require special considerations for their implementation. For instance, the query to determine the number of unique internal IPs associated with each external address can have an immensely large upper bound on the amount of memory required to store this information, given how many unique IP addresses may be seen. To circumvent this issue of storing too much in memory at once, we can pre-process all the flows by accumulating all relevant information into subsets and partitioning the total of the data into smaller files first, then handling each file individually. While that specific query went unused in this thesis, it was part of the initial investigation into characterizing NetFlow data and may be utilized for future analysis.

In processing these daily datasets, the primary factor that impacts the speed of data processing (beyond the sheer quantity of observed total flows) is the technique used to store unique elements. For example, when computing the number of unique IP addresses seen per five-minute window, the efficiency of the data structure used for storing these unique elements is critical. In this investigation, we utilized the "*unordered_map*" and "*unordered_set*" objects from the C++ standard library. While the default hashing functions used by these objects were sufficient to insert and count these unique elements, the use of specialized hashing functions for storing elements like unique IP addresses may increase the speed of computation in future investigations. This would be due to an improved distribution of hashed values, which goes hand in hand with a reduced likelihood of collisions when inserting these elements into a data structure.

For each query, we observe around 50 GB of *nfcapd* binary files associated with each date, and an additional 15 GB of an early portion of the next day to account for any active flows only seen after terminating the following day. For this amount of data, we can process any profiling query regarding incoming flows in around 15 minutes for simpler accumulation (such as the number of incoming bytes per window) or up to 30 minutes when the tracking of unique elements is required.

Chapter 3: Analysis

The primary goal of this thesis is to discuss what information that can characterize a network can be extracted from analyzing traffic data at varying scopes. To achieve this, we discuss three primary levels of analysis that will be conducted on snapshots of the UO campus network.

First, we will perform an aggregate-level analysis of all incoming traffic to examine the prevalence of specific transport protocols that make up the sum of incoming Internet traffic.

Using insights gained from aggregate-level analysis regarding the most prevalent transport protocol, we will perform two port-level analyses of the flows that make up the majority of that portion of incoming web traffic. At this level, we observe the traffic patterns of major application-level protocols and major applications (identifiable by port) associated with the most prevalent transport protocol.

Lastly, our lowest-level analysis observes the traffic patterns of major ASes, identifiable by ASN, that make up the largest subset of traffic from the previous level.

3.1. Aggregate-Level Analysis

At the top level, we aim to identify any characteristics that can be deduced by observing an entire day's incoming traffic. This traffic will be grouped based on the transport protocols used by each flow.

Traffic Volume Analysis. For this first analysis, we are focused on observing the amount of incoming data per 5-minute window, categorized into TCP, UDP, and other flows. Our first such observation is from Figure 1, detailing the volume of incoming TCP, UDP, and other data over each 5-minute window for a few selected daily snapshots. Here, we see that the volume of TCP traffic dominates over the other transport protocols. This is true for all moments of the day, including quiet hours which take place around 2 am to 8 am, and during peak hours, which are most commonly around noon to 4 pm. Additionally, traffic that belongs to neither TCP nor UDP makes up such a small portion of traffic that it is not visible on any snapshots.

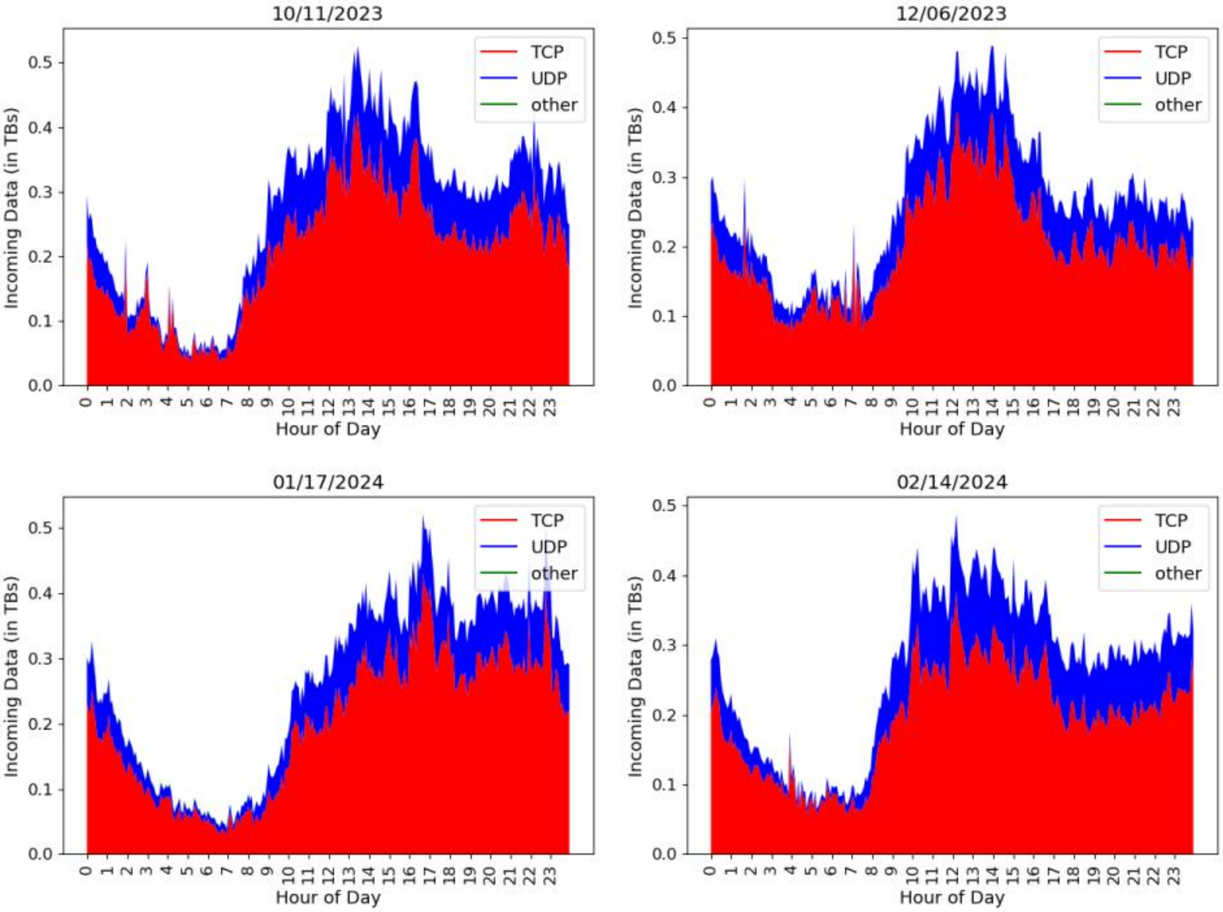

Figure 1: Incoming aggregate traffic volume (in TB) per 5-minute window, split into TCP, UDP, and Other flows

Flow Arrival Analysis. Figure 2 depicts the number of newly arriving TCP, UDP, and other flows per 5-minute time window from selected daily snapshots. This figure additionally confirms TCP's prevalence in the UO network, showing that TCP traffic accounts for up to an order of magnitude more arriving flows than all non-TCP traffic combined at any given moment.

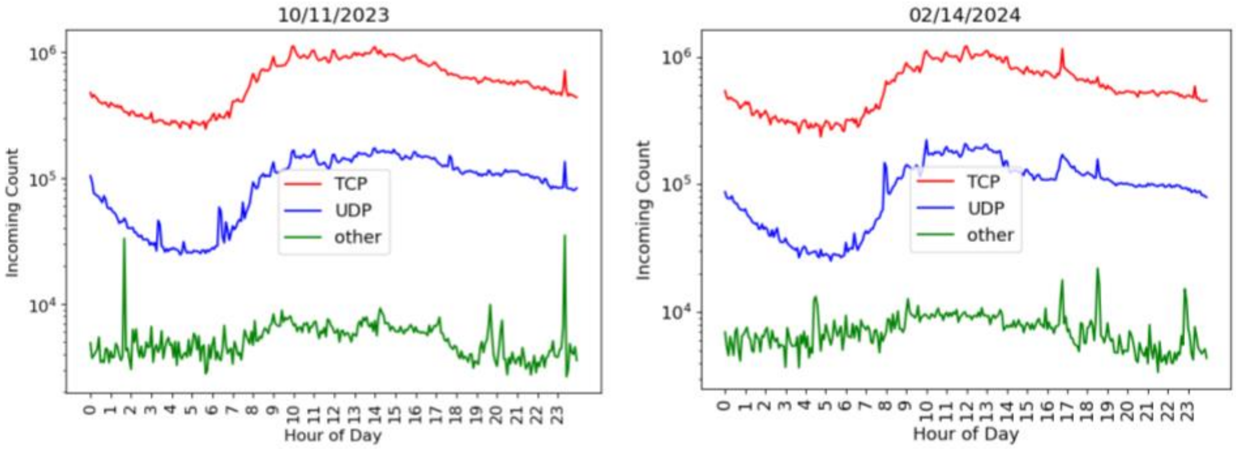

Figure 2: Temporal pattern of newly arriving TCP/UDP/Other flows per 5-minute window

Figure 3 provides a more detailed look into the composition of newly incoming flows observed during each day. Unlike Figure 2, which provides a direct comparison of the number of new flows per flow subset, Figure 3 provides additional insight into traffic characteristics of the newly incoming flows of each subset. It shows the number of unique external source IP addresses (both IPv4 and IPv6), IPv4 /24 prefixes, and ASNs that make up these newly arriving connections. By including these metrics, we can reason that sudden changes such as a surge in the number of unique IP addresses could indicate potential anomalies.

IPv4 /24 prefixes, a subset of the IPv4 address space focusing on the first 24 bits of an address, are included as an additional level of observation. Here, we prioritize IPv4 connections over IPv6 connections because IPv4 addresses account for around 70% of the unique incoming source IP addresses on both 10/11/2023 and 02/14/2024.

Overall, this figure shows that, for each of these views, the arrival rate remains largely stable, and that the counts for each of these levels are similar across both dates.

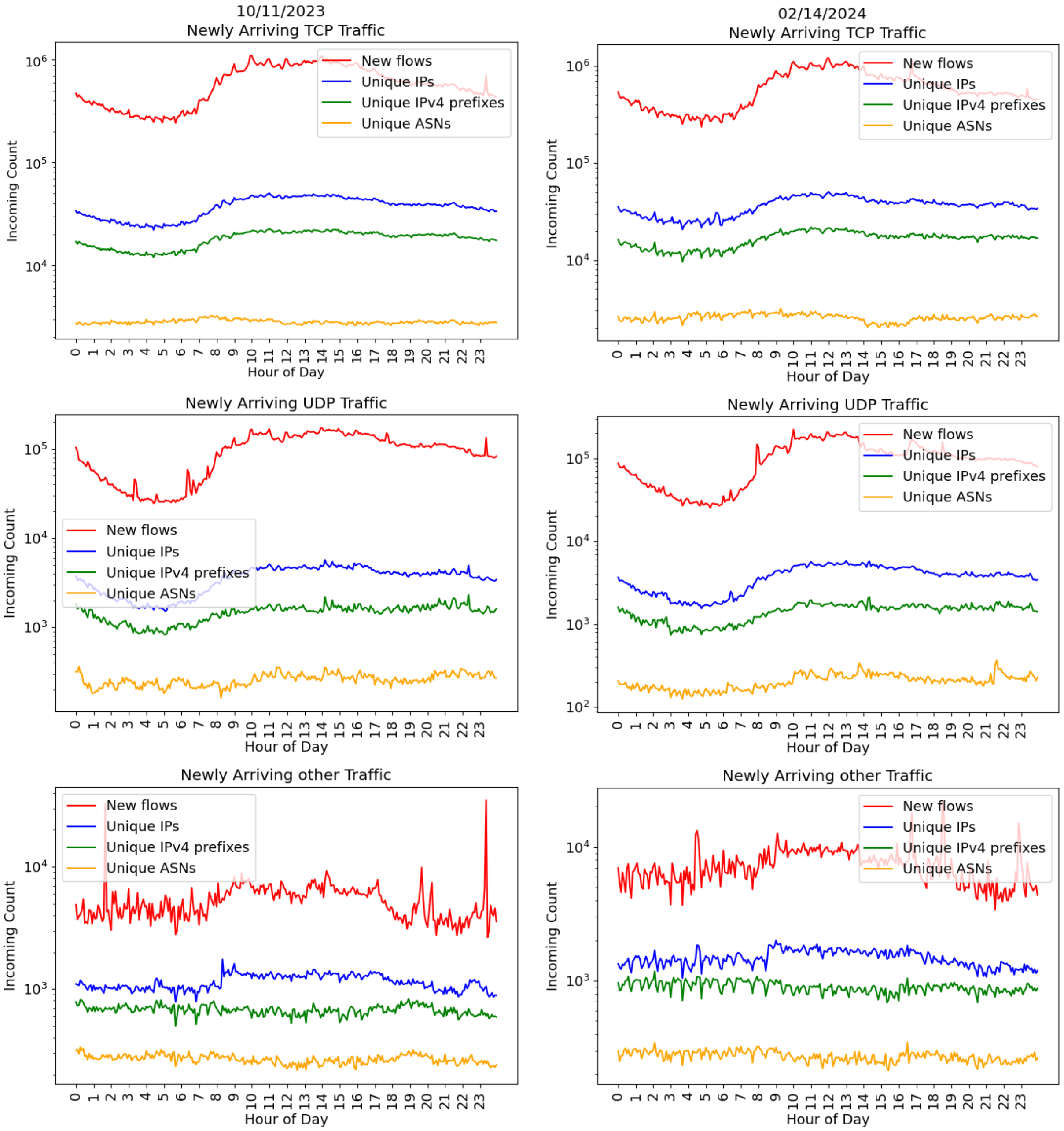

Figure 3: Temporal pattern of incoming source IP, IPv4 /24 prefix, and ASN counts of TCP/UDP/Other flows per 5-minute window

Flow Size Analysis. Figure 4 shows the distribution of observed flow sizes for incoming TCP, UDP, and other flows. From this figure, we see that UDP flows tend to have the largest volume of the flow subsets. Additionally, all three subsets of aggregate traffic are quite consistent and stay in their respective ranges over all snapshots.

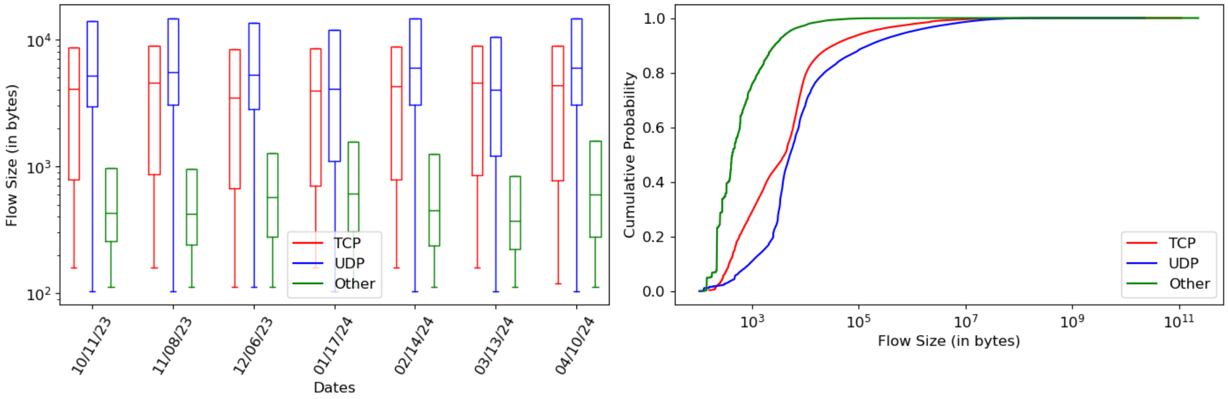

Figure 4: Summary distribution of flow size for incoming TCP/UDP/Other flows across all snapshots (left) and CDF of flow size for incoming TCP/UDP/Other flows in a sample snapshot of 10/11/2023 (right)

Flow Duration Analysis. Figure 5 shows the distribution of observed flow durations for incoming TCP, UDP, and other flows. This figure illustrates that TCP and UDP connections are much shorter than the other connections observed, with TCP traffic tending to be slightly longer on average than UDP on most days.

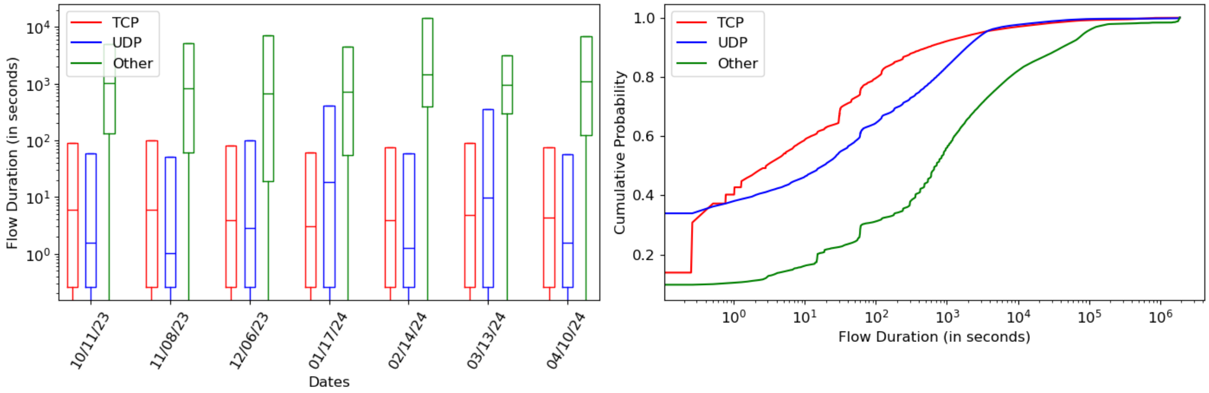

Figure 5: Summary distribution of flow duration for incoming TCP/UDP/Other flows across all snapshots (left) and CDF of flow duration for incoming TCP/UDP/Other flows in a sample snapshot of 01/17/2024 (right)

Flow Throughput Analysis. Figure 6 shows the distribution of observed throughputs of incoming TCP, UDP, and other flows. With the above observations, Figure 6 helps illustrate that UDP flows have higher throughput than TCP flows primarily due to the larger flow size, and shorter duration flows. Additionally, when compared to UDP and TCP traffic, these “other” flows exhibit incredibly low throughput, often three to four orders of magnitude less than UDP and TCP traffic, on average.

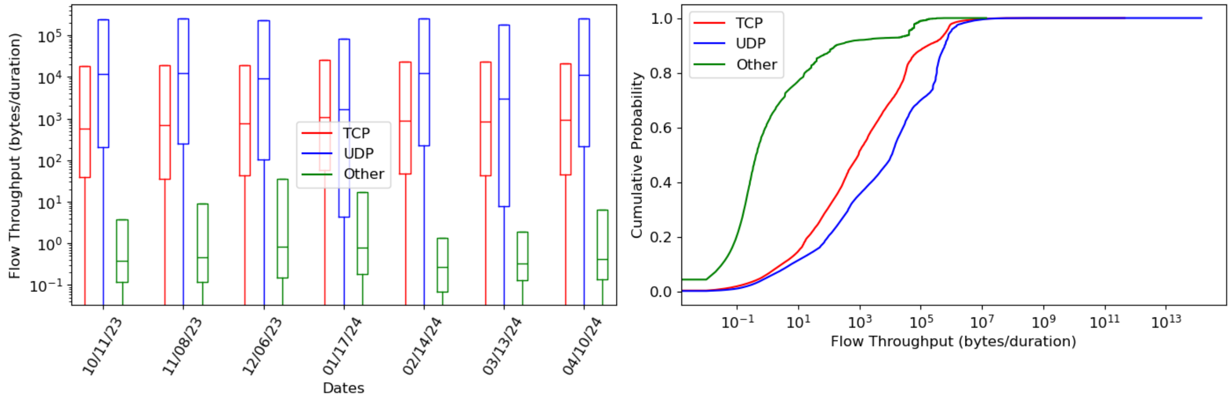

Figure 6: Summary distribution of flow throughput for incoming TCP/UDP/Other flows across all snapshots (left) and CDF of flow throughput for incoming TCP/UDP/Other flows in a sample snapshot of 04/10/2024 (right)

Key Takeaways:

From this highest level of aggregate analysis, we observe the following characteristics of incoming aggregate traffic:

- TCP flows make up a significant portion of incoming network traffic at all given points of the day while the “other” traffic category is so miniscule that it is never visible on any graphs. (Figure 1)
- Additionally, Figure 1 shows that the network is least active from 3 am through 8 am, and usually most active around 12 pm to 3 pm, though these peak hours are more variable.

- TCP flows make up a significant portion of incoming network traffic primarily due to the substantial number of new connections made each day. (Figure 2)
- This can be confirmed by observing that TCP flows do not report the highest average flow size; rather, UDP flows are the largest on average. (Figure 4)
- Of the three observed subsets of incoming flows, UDP flows have the highest throughput, while the "other" flows have the lowest throughput. (Figure 6)
- The high throughput of UDP flows is due to larger flow sizes (Figure 4) over shorter durations (Figure 5).
- The distribution of IP addresses, IPv4 prefixes, and ASNs that make up these newly incoming flows are stable over time and consistent across both observed dates. This indicates that these metrics follow expected and predictable patterns. (Figure 3)

3.2. Aggregate TCP Analysis

At this next level, we aim to identify any characteristics that can be deduced by observing the ports utilized by the largest subset of data found in the aggregate traffic analysis from Section 3.1. As such, we observe the characteristics of ports most used in TCP traffic.

Figure 7 details the total number of incoming flows per daily snapshot, divided into the most used TCP source ports observed in each snapshot. Here, we see HTTPS (port 443) accounting for most of the incoming flows, followed by HTTP (port 80), Internet Message Access Protocol (IMAP, port 993), and Apple Push Notification Service (APNS, port 4070).

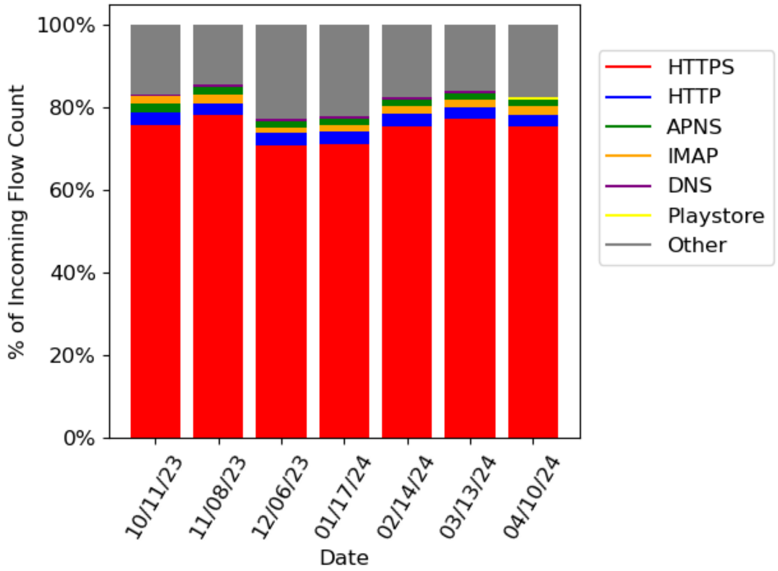

Figure 7: Stacked plot of most frequently used TCP source ports in incoming flows

Based on Figure 7, our analysis at the aggregate TCP level is broken down further into two sub-categories: web traffic analysis, which will feature the HTTPS and HTTP transfer protocols, and application-level analysis, which will feature APNS, IMAP, and additionally Spotify. Observations from both analyses are presented separately, beginning with the web traffic analysis.

3.2.1. Web Traffic Analysis

In our first investigation at this level of analysis, we aim to identify any characteristics that can be deduced by observing HTTPS, HTTP and other TCP flows that make up incoming TCP traffic. This investigation is titled “Web Traffic Analysis” given that the HTTP/S protocols are the primary protocols used for communication between web browsers and websites.

Traffic Volume Analysis. To observe what ports are used most frequently by TCP flows, Figure 8 describes the volume of incoming HTTPS, HTTP, and other TCP data over each 5-minute window of four selected daily snapshots. We find that HTTPS flows dominate incoming traffic across all snapshots, with HTTP flows being the second most seen.

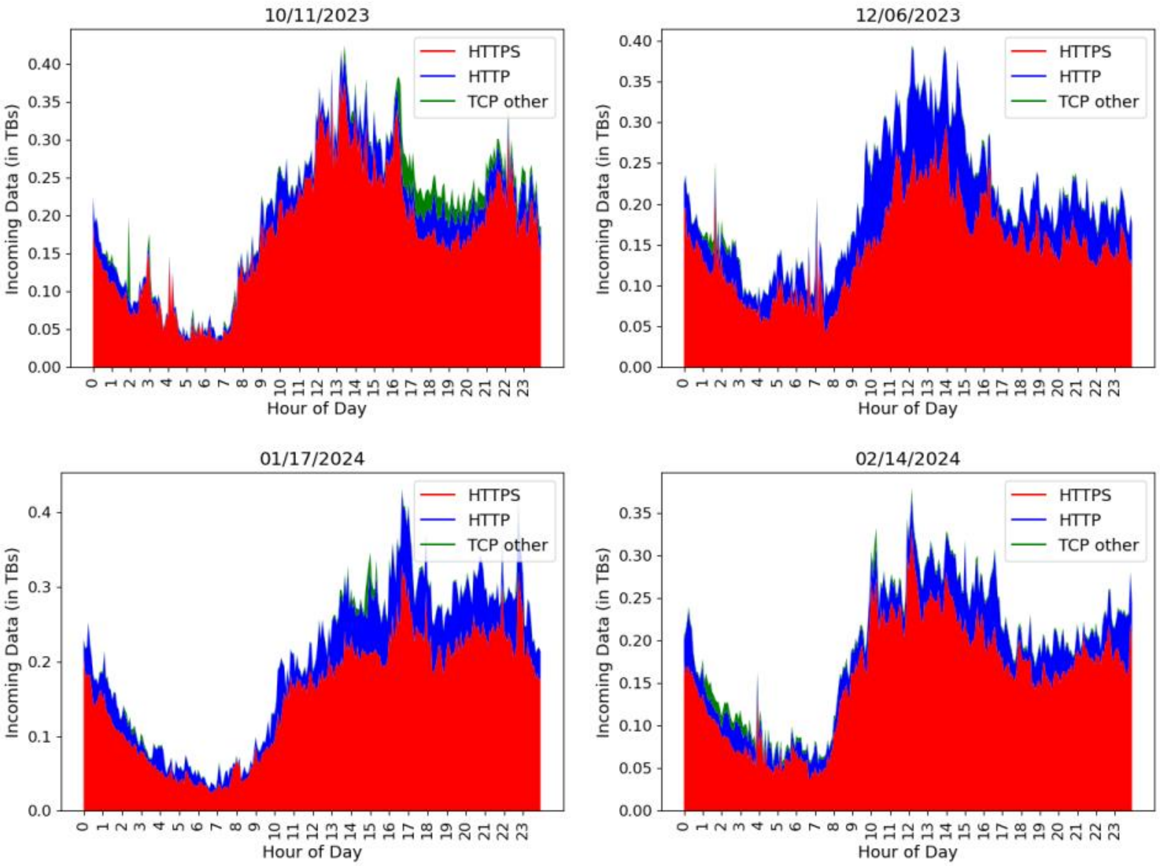

Figure 8: Incoming TCP traffic volume (in TB) per 5-minute window, split into HTTPS, HTTP, and Other TCP flows

Flow Arrival Analysis. Figure 9 shows the number of newly arriving HTTPS, HTTP, and other TCP flows per 5-minute time window from selected daily snapshots. This figure additionally confirms the prevalence of HTTPS in incoming TCP traffic by showing that HTTPS traffic accounts for a large majority of new incoming flows in each 5-minute window. On the other hand, while there are more “other TCP” flows than HTTP flows at all times of the day, these flows don’t carry as much data, as Figure 8 shows that these miscellaneous flows contribute very little to the overall volume of incoming TCP traffic.

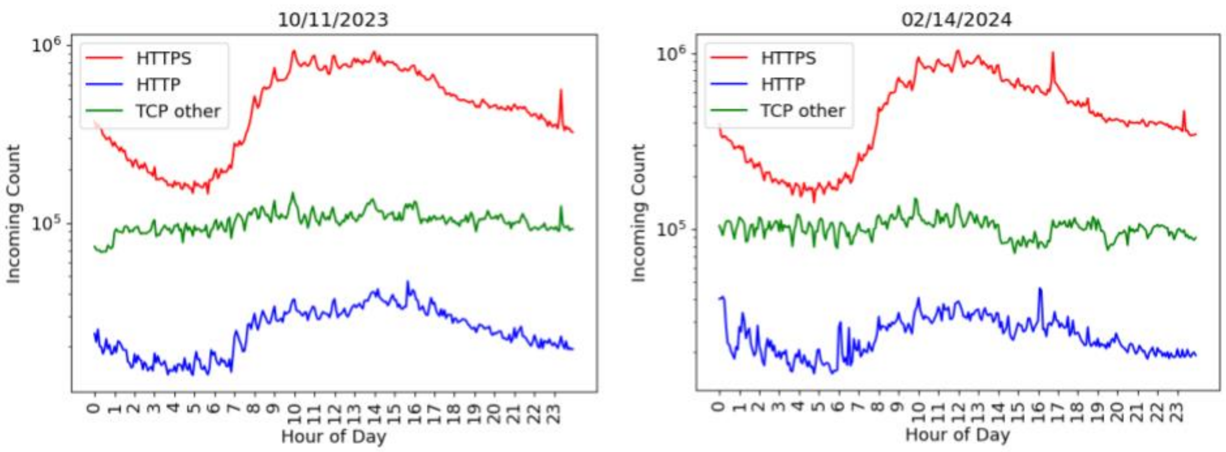

Figure 9: Temporal pattern of newly arriving HTTPS/HTTP/Other TCP flows per 5-minute window

Figure 10 provides a more detailed look into the composition of newly incoming HTTPS, HTTP, and other TCP flows observed during each day by showing the number of unique external source IP addresses, IPv4 /24 prefixes, and ASNs that make up these newly arriving connections.

For Figure 3, we focused on IPv4 /24 prefixes, as IPv4 connections as a whole account for approximately 70% of unique incoming source IP addresses on both 10/11/2023, and 02/14/2024.

Overall, this figure shows that these measures of uniqueness remain largely stable for HTTPS flows, while HTTP and other TCP flows exhibit more fluctuation. This fluctuation is consistent across all 3 levels of uniqueness. Despite these variations, the overall counts at each level are similar on both dates.

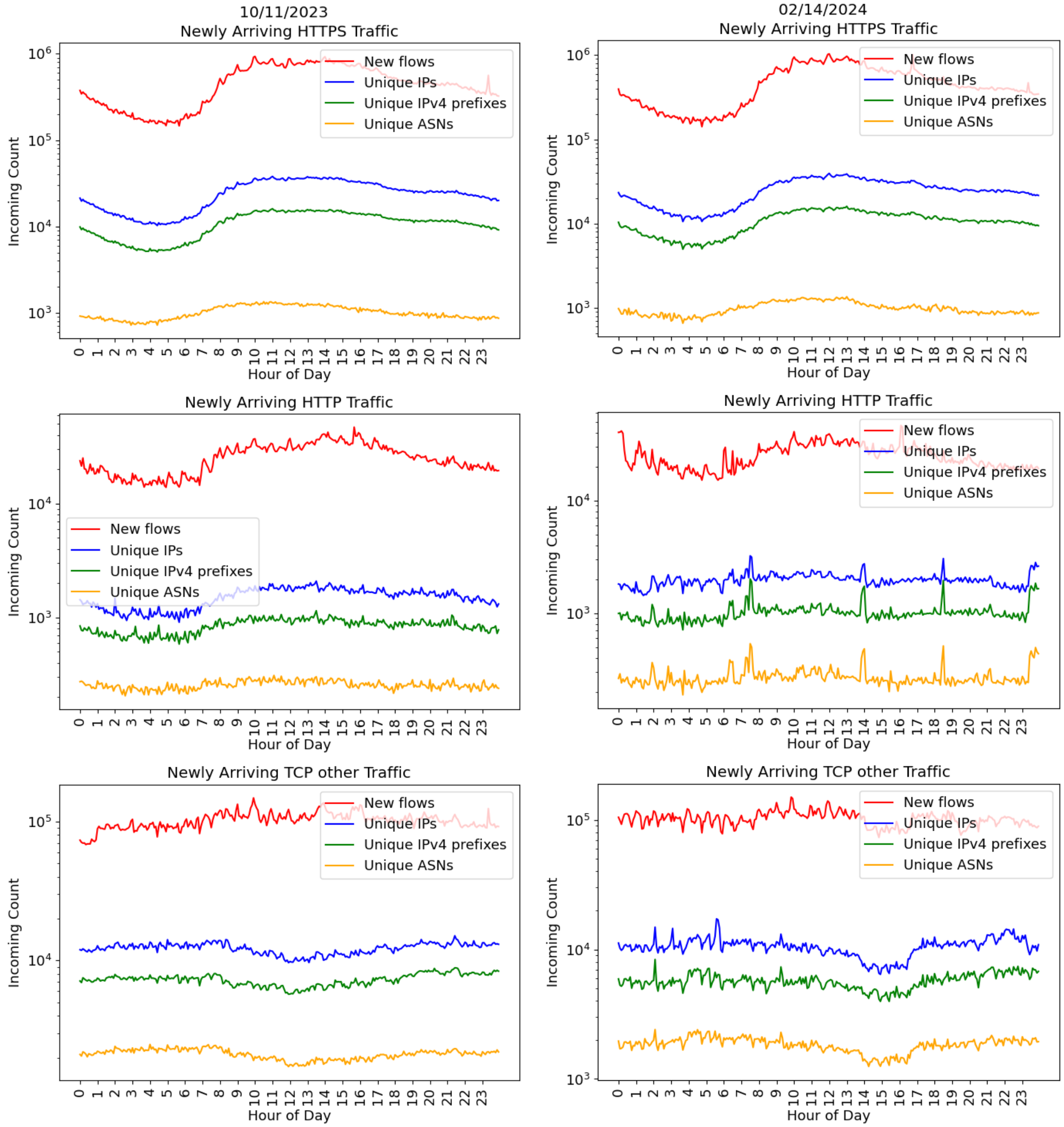

Figure 10: Temporal pattern of incoming source IP, IPv4 /24 prefix, and ASN counts of HTTPS/HTTP/Other TCP flows per 5-minute window

Flow Size Analysis. Figure 11 illustrates the distribution of observed flow sizes of incoming HTTP, HTTPS, and other TCP flows.

From this figure, we can see that the flow sizes are consistently much larger for HTTPS, while HTTP and other TCP ports have similar flow sizes.

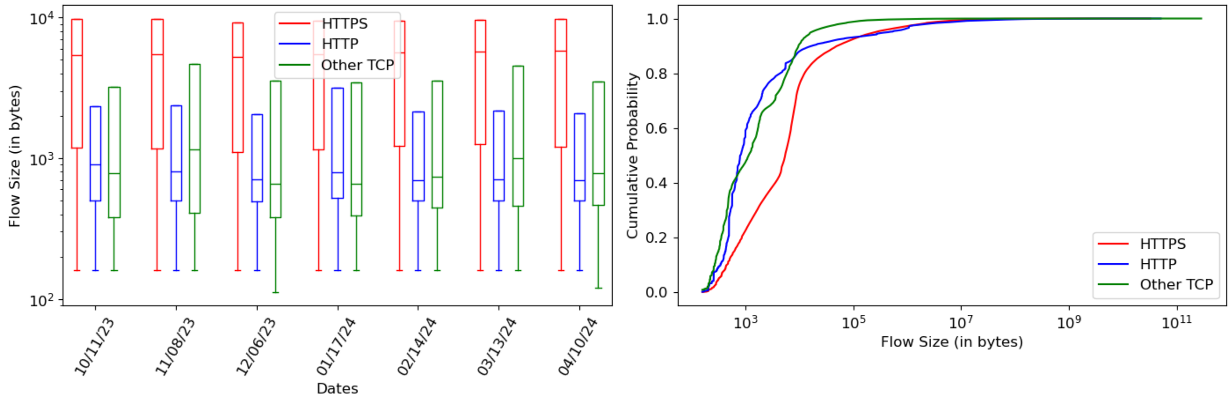

Figure 11: Summary distribution of flow size for incoming HTTPS/HTTP/Other TCP flows across all snapshots (left) and CDF of flow size for incoming HTTPS/HTTP/Other TCP flows in a sample snapshot of 11/08/2023 (right)

Flow Duration Analysis. Figure 12 observes the distribution of flow durations of incoming TCP flow subsets. Here, we see that HTTPS flows tend to last the longest of all TCP flow subsets.

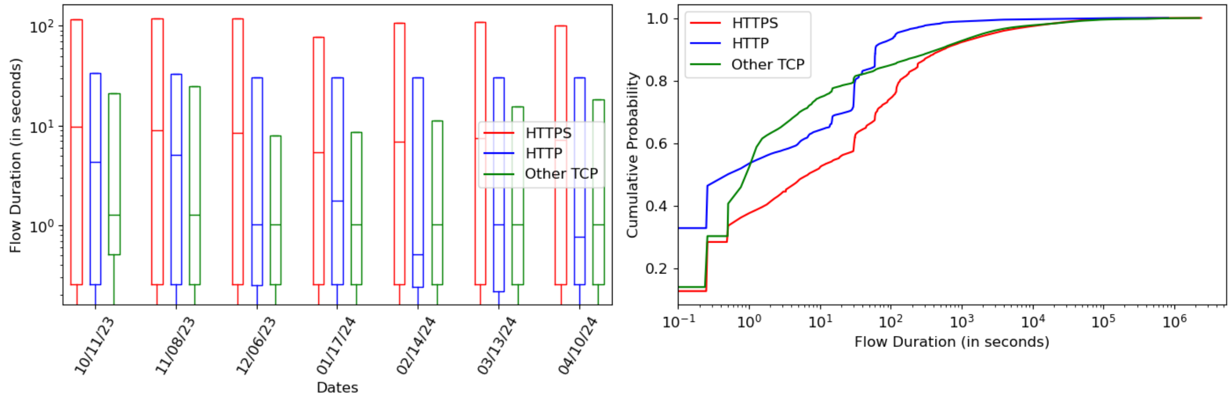

Figure 12: Summary distribution of flow duration for incoming HTTPS/HTTP/Other TCP flows across all snapshots (left) and CDF of flow duration for incoming HTTPS/HTTP/Other TCP flows in a sample snapshot of 02/14/2024 (right)

Flow Throughput Analysis. Our last analysis from this section is centered around Figure 13, which shows the distribution of observed flow throughputs of incoming TCP flow subsets. Here, we see that the average throughput between all types of TCP traffic is similar. This means they deliver data at similar rates, and their main differences lie in their flow sizes (Figure 11) and durations (Figure 12).

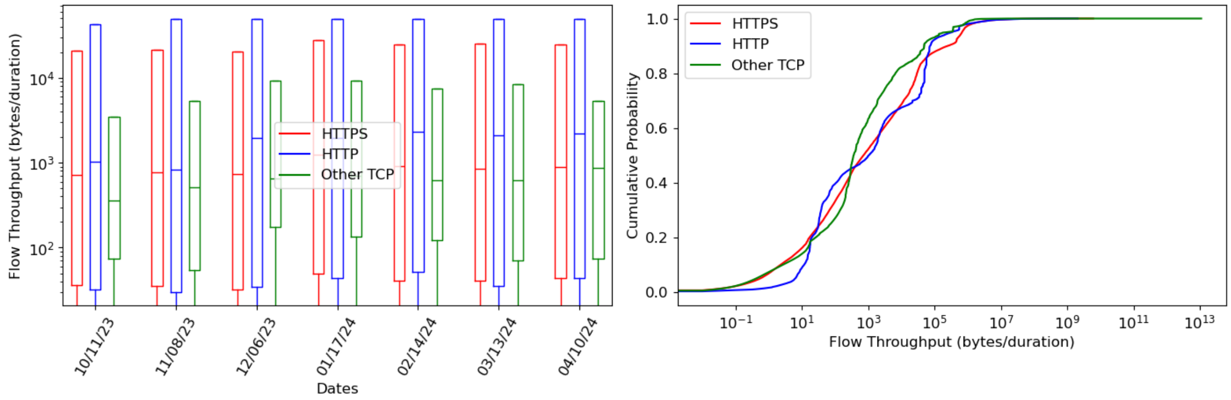

Figure 13: Summary distribution of flow throughput for incoming HTTPS/HTTP/Other TCP flows across all snapshots (left) and CDF of flow throughput for incoming HTTPS/HTTP/Other TCP flows in a sample snapshot of 10/11/2023 (right)

Key Takeaways:

With these observations of application-layer protocols (HTTPS and HTTP), we can conclude the following about TCP traffic.

- We can see similar daily patterns from Figure 8 as above in Figure 1 regarding activity at peak vs. quiet hours. This makes sense, considering TCP traffic makes up most of the incoming aggregate traffic.
- HTTPS makes up a significant portion of new incoming network connections at this level. (Figure 9)
  - Additionally, HTTPS sends the most data (Figure 11) over longer periods (Figure 12) when compared to the other subsets at this level.

- Even with the varying flow sizes and durations of the subsets of TCP traffic, the throughput is similar among these groups, meaning they deliver comparable amounts of data in terms of speed. (Figure 13)
    - By this, we conclude that HTTPS flows tend to deliver more bytes than all other TCP traffic due to the longer flow durations observed.
- The distribution of IP addresses, IPv4 prefixes, and ASNs observed from the new flows are stable over time for HTTPS, while they are more variable for HTTP and other flows.
    - The counts of each metric are consistent across both dates, which indicates that these metrics follow predictable patterns in terms of volume, though HTTP and other TCP flows are more unpredictable. (Figure 10)

3.2.2. Application-Level Analysis

Our second analysis at this level aims to observe characteristics of specific applications identifiable by port. It is important to note that HTTPS and HTTP traffic cannot be included in this investigation, as these ports do not belong to any specific application. Rather, many applications communicate using these protocols [4].

From Figure 7 above, we saw that Apple Push Notification Service (APNS) and Internet Message Access Protocol (IMAP) are the most used ports beyond HTTPS and HTTP.

APNS is a cloud service utilizing port 5223 over TCP that allows third-party apps to deliver notifications to Apple devices, and IMAP, using port 993 over TCP, enables email clients to retrieve mail from email servers using SSL/TLS encryption [5].

Additionally, Spotify is included in our analysis. Utilizing port 4070 over TCP, observing Spotify, the most popular online music streaming service, may offer important insight into this class of applications.

Traffic Volume Analysis. To start our analysis of these applications' characteristics, Figure 14 shows the volume of incoming application data over each 5-minute window of four selected snapshots. Our first observation is that IMAP is the application that accounts for most of the incoming traffic by volume at all points of each day. IMAP can also be seen to exhibit behaviors of sending large bursts of data without following any particular pattern. This trend can be explained by the nature of IMAP as an application; IMAP, as a service that exists to retrieve emails for a user, is heavily human driven, which results in a significant variance in traffic volume from day to day, depending on the demand.

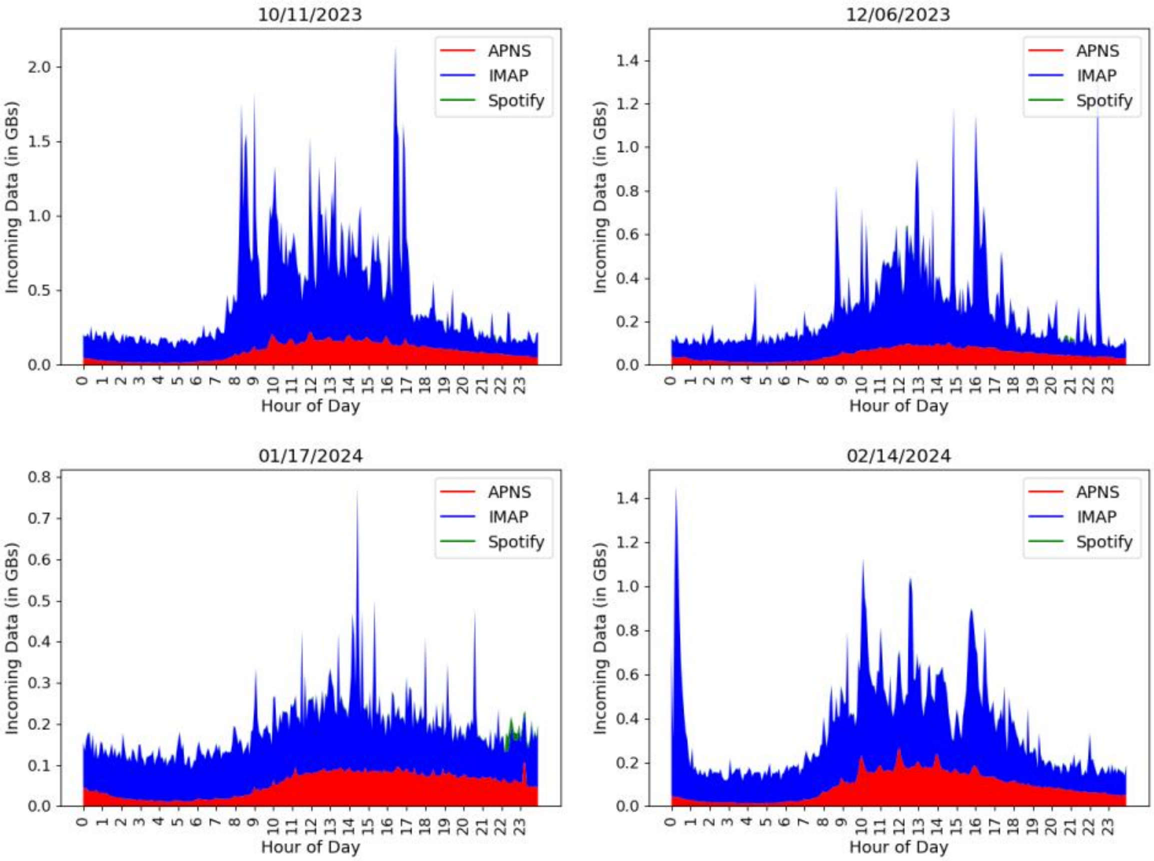

Figure 14: Incoming port-based application traffic volume (in GB) per 5-minute window, split into APNS, IMAP, and Spotify flows

Flow Arrival Analysis. Figure 15 shows the number of newly arriving flows from each application per 5-minute time window from selected daily snapshots. The main observation is that there are somewhat similar numbers of new APNS and IMAP connections made per 5-minute window, though the arrival rate of IMAP flows fluctuates significantly. These peaks in newly arriving flows do not necessarily correspond to peaks observed in Figure 14, so it must be that IMAP flows have a larger range of observed flow sizes.

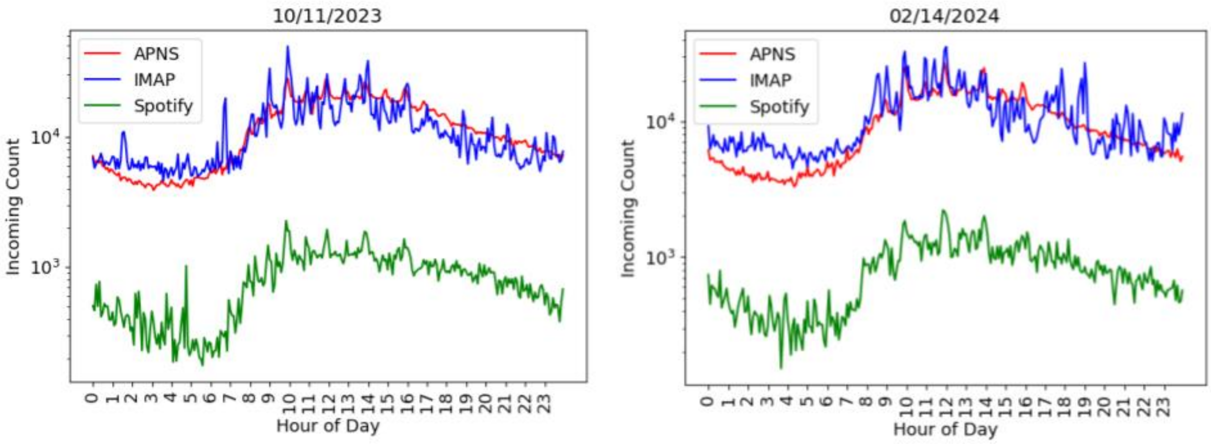

Figure 15: Temporal pattern of newly arriving APNS/IMAP/Spotify flows per 5-minute window

Flow Size Analysis. Figure 16 displays the distribution of observed flow sizes of incoming traffic from APNS, IMAP, and Spotify. When we look at Figure 16, we can see that the flow sizes of APNS and IMAP flows are consistent across multiple snapshots and have very similar upper bounds in flow sizes, while the sizes of Spotify flows can vary significantly.

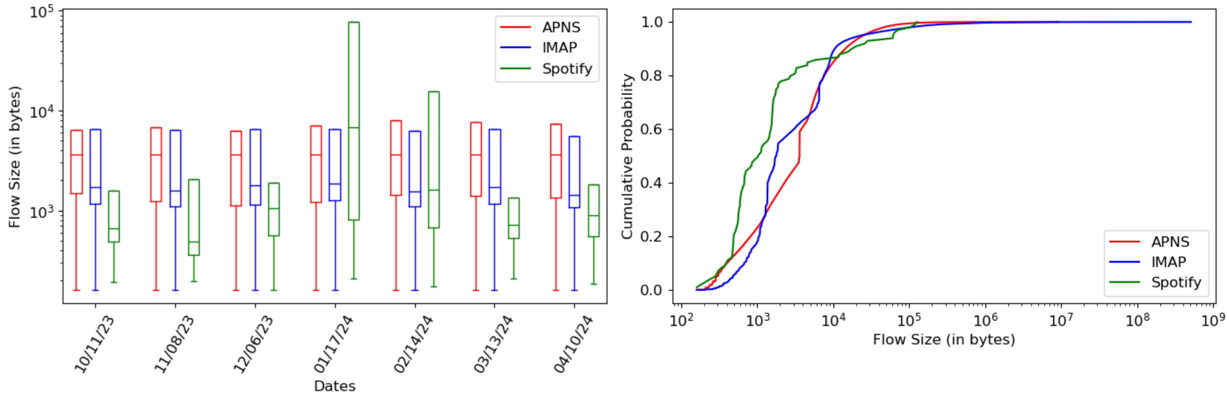

Figure 16: Summary distribution of flow size for incoming APNS/IMAP/Spotify flows across all snapshots (left) and CDF of flow size for incoming APNS/IMAP/Spotify flows in a sample snapshot of 12/06/2023 (right)

Flow Duration Analysis. Figure 17 shows the distribution of flow durations of incoming traffic from APNS, IMAP, and Spotify flows. This investigation into flow duration reveals distinct characteristics of each application: APNS flows consistently last much longer than the other applications (at least an order of magnitude longer). IMAP flows are very short-lived, and Spotify is also likely to exhibit shorter-lived flows, though we see more variance in the upper bound of this Application.

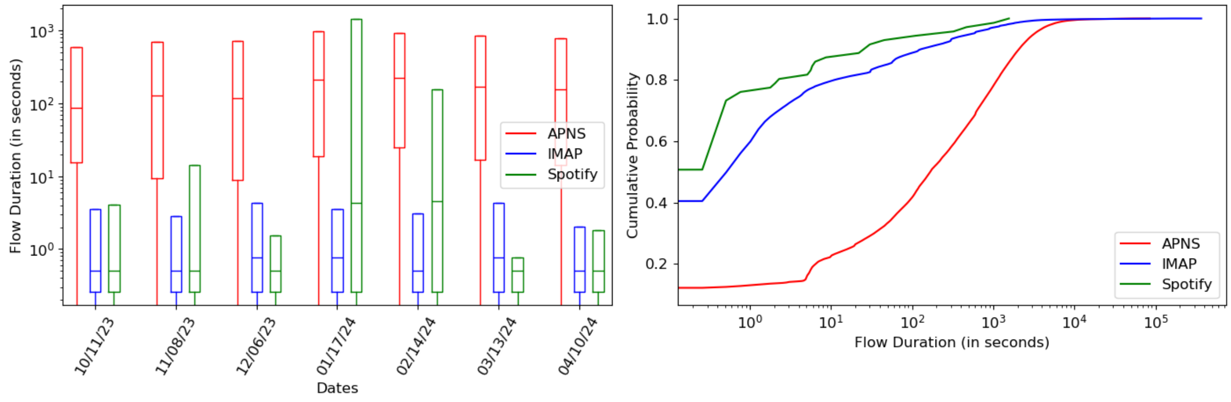

Figure 17: Summary distribution of flow duration for incoming APNS/IMAP/Spotify flows across all snapshots (left) and CDF of flow duration for incoming APNS/IMAP/Spotify flows in a sample snapshot of 03/13/2024 (right)

Flow Throughput Analysis. Lastly, Figure 18 shows the distribution of observed flow throughputs of incoming APNS, IMAP, and Spotify flows. This figure shows that the throughput of APNS and IMAP flows stay consistent over the snapshots, with APNS favoring low throughput and IMAP favoring high throughput. In contrast, Spotify's throughput varies from day to day, though flows can exhibit high throughput like IMAP in certain snapshots.

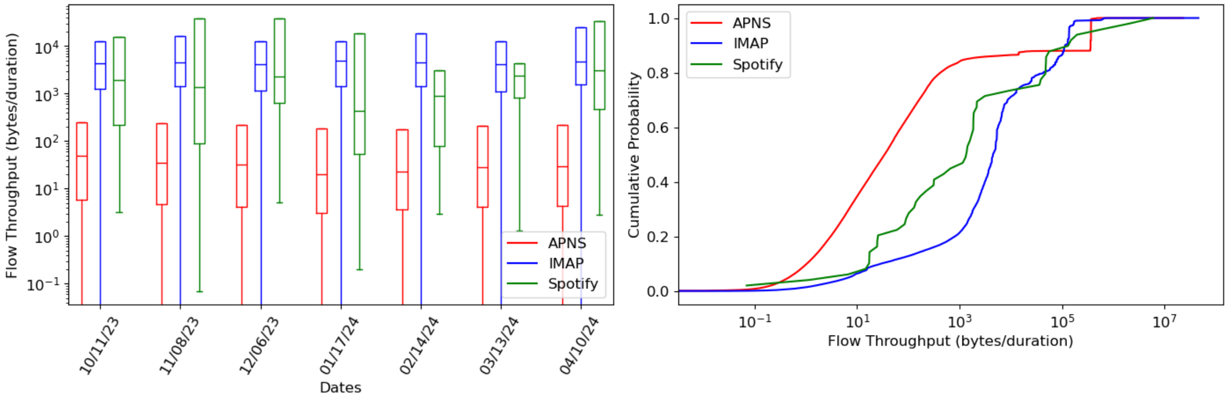

Figure 18: Summary distribution of flow size for incoming APNS/IMAP/Spotify flows across all snapshots (left) and CDF of flow size for incoming APNS/IMAP/Spotify flows in a sample snapshot of 11/08/2023 (right)

Key Takeaways:

Through observing specific port numbers found in incoming NetFlow data, we can conclude the following about port-identifiable applications found in TCP traffic:

- IMAP accounts for the most incoming data of the three observed applications. However, this data sent by IMAP exhibits lots of variance, sending data in large bursts, rather than consistently like APNS. (Figure 14)
  - There is no significant correlation between peaks in incoming IMAP data by volume (Figure 14) and the rate of newly arriving flows (Figure 15), so it must be that IMAP flows have a larger variance of observed flow sizes as well.

- Despite the differences in flow duration (Figure 17) and throughput (Figure 18) for each application, the flow sizes for these applications are similar, especially APNS and IMAP. (Figure 16) Therefore, these applications primarily differ by their tendency to:
    - Prefer longer, sustained communication (APNS).
    - Prefer shorter, high-throughput communication (IMAP and Spotify).
- Specific applications such as streaming services may have variable demands, and associated flows are expected to exhibit a more significant variance of flow-level characteristics on a day-to-day basis.
- Overall, when observing specific applications identifiable by port, we can see the variable demands of human-driven traffic more clearly through these large bursts of data.
    - However, like previously observed patterns of incoming data (Figure 14) and new flows (Figure 15), traffic at this level follows similar peaks in activity at noon and quiet hours starting at 2 am.

3.3. AS Traffic Analysis

From the application-layer protocol analysis of TCP traffic, we observed that HTTPS is the dominant application-layer protocol utilized by the network. With this information, we can go a level deeper into HTTPS traffic, observing which autonomous systems utilize HTTPS to communicate with UO network users. This can be done by observing the Autonomous System Number (ASN) associated with each incoming HTTPS flow.

To choose which autonomous systems will be observed, we took an initial look into what ASNs are seen to deliver the most information to the UO network.

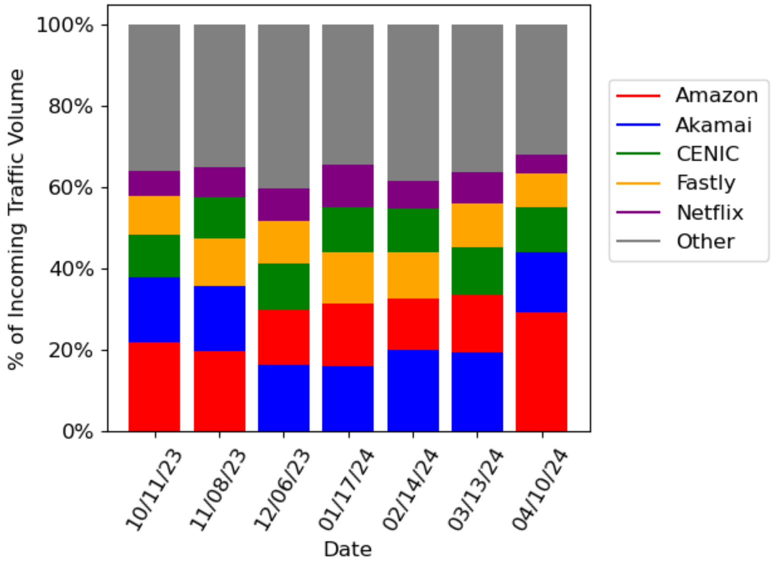

Figure 19: Stacked plot of incoming HTTPS traffic by total volume, split by top ASNs

From our findings in Figure 19, we selected three notable autonomous systems. The chosen autonomous systems are Akamai Technologies (ASN 20940), Amazon (ASN 16509), and Netflix (ASN 2906).

Netflix and Akamai were chosen as they appear as the most prevalent content delivery network (CDN) providers on our network. CDNs are a network of geographically dispersed servers that cache Internet content close to the end user. This better proximity allows for quicker access to Internet content through better latency and reduced load times [6].

On the other hand, Netflix was chosen as the most prevalent online streaming service. Unlike the previously listed CDNs, ASN 2906 is directly related to serving users with Netflix data and may exhibit unique traffic patterns regarding the serving of streamed content.
Comparing and contrasting the characteristics of CDNs and streaming services at this level will offer valuable insights into how different autonomous systems utilize the UO network.

Traffic Volume Analysis. Our first observation at this level focuses on Figure 20, which shows the volume of incoming traffic from Akamai, Amazon, and Netflix over time. From this figure, we see that there is not always a dominant flow type at this level, unlike the previous levels of analysis. Akamai sends more data overall in certain snapshots, while Amazon sends more data in others. However, Akamai and Netflix both seem to send data more consistently, while Amazon has a very high variance in terms of bytes sent, showing large bursts and peaks of sent data, much like IMAP in the previous investigation.

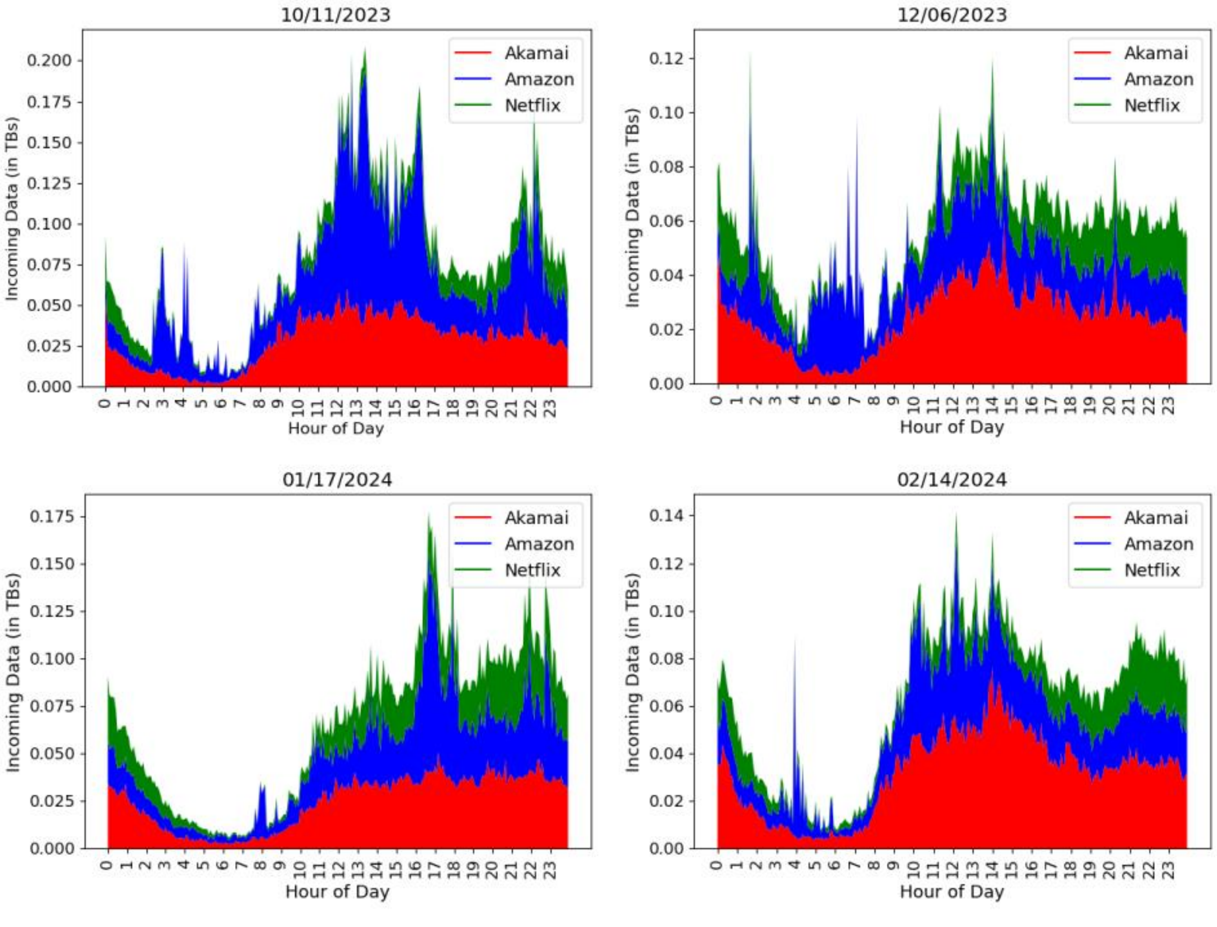

Figure 20: Incoming ASN-based traffic volume (in TB) per 5-minute window, split into Akamai, Amazon, and Netflix flows

Flow Arrival Analysis. Figure 21 shows the number of newly arriving flows from each AS for each 5-minute window over selected daily snapshots, along with a CDF from a sample snapshot. When observing how many new connections happen throughout the day, we see that Akamai and Amazon make connections at very similar rates. Given that the arrival rates of these flows do not indicate as obvious peaks as Figure 20 above, it may be the case that the characteristics of Amazon flows (such as byte size and duration) may be more varied, and depend a lot on the demand.

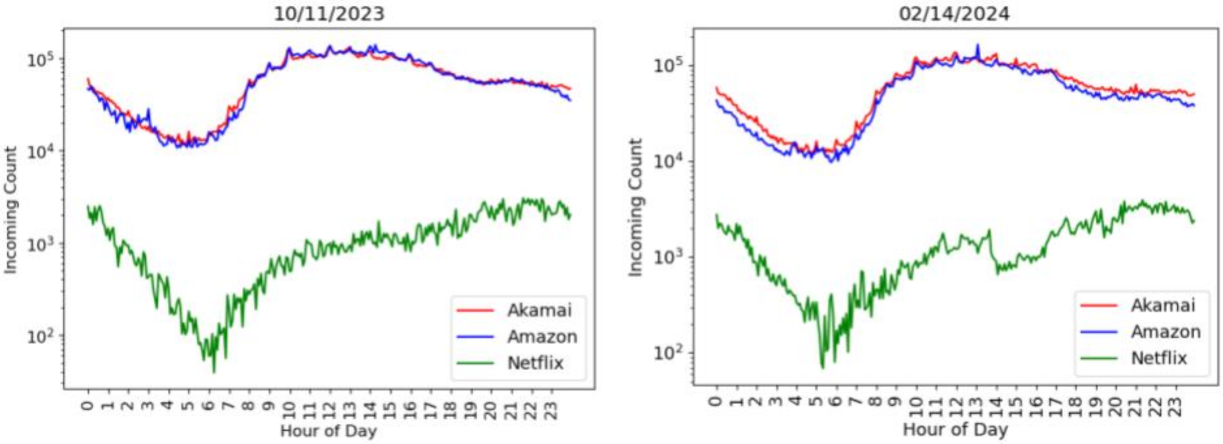

Figure 21: Temporal pattern of newly arriving Akamai/Amazon/Netflix flows per 5-minute window

Figure 22 provides a more detailed look into the composition of newly incoming Akamai, Amazon, and Netflix flows observed during each day by additionally showing the number of unique external source IP addresses and IPv4 /24 prefixes that make up these newly arriving connections. This view gives a detailed look into the difference in the size of the IP address space reserved for and utilized by each AS. Amazon uses around an order of magnitude more unique IP addresses than Akamai, and two orders of magnitude more than Netflix.

Though Akamai and Amazon both make a similar number of connections at all points of the day, we see that this traffic comes from a much more focused set of IP addresses from Akamai.

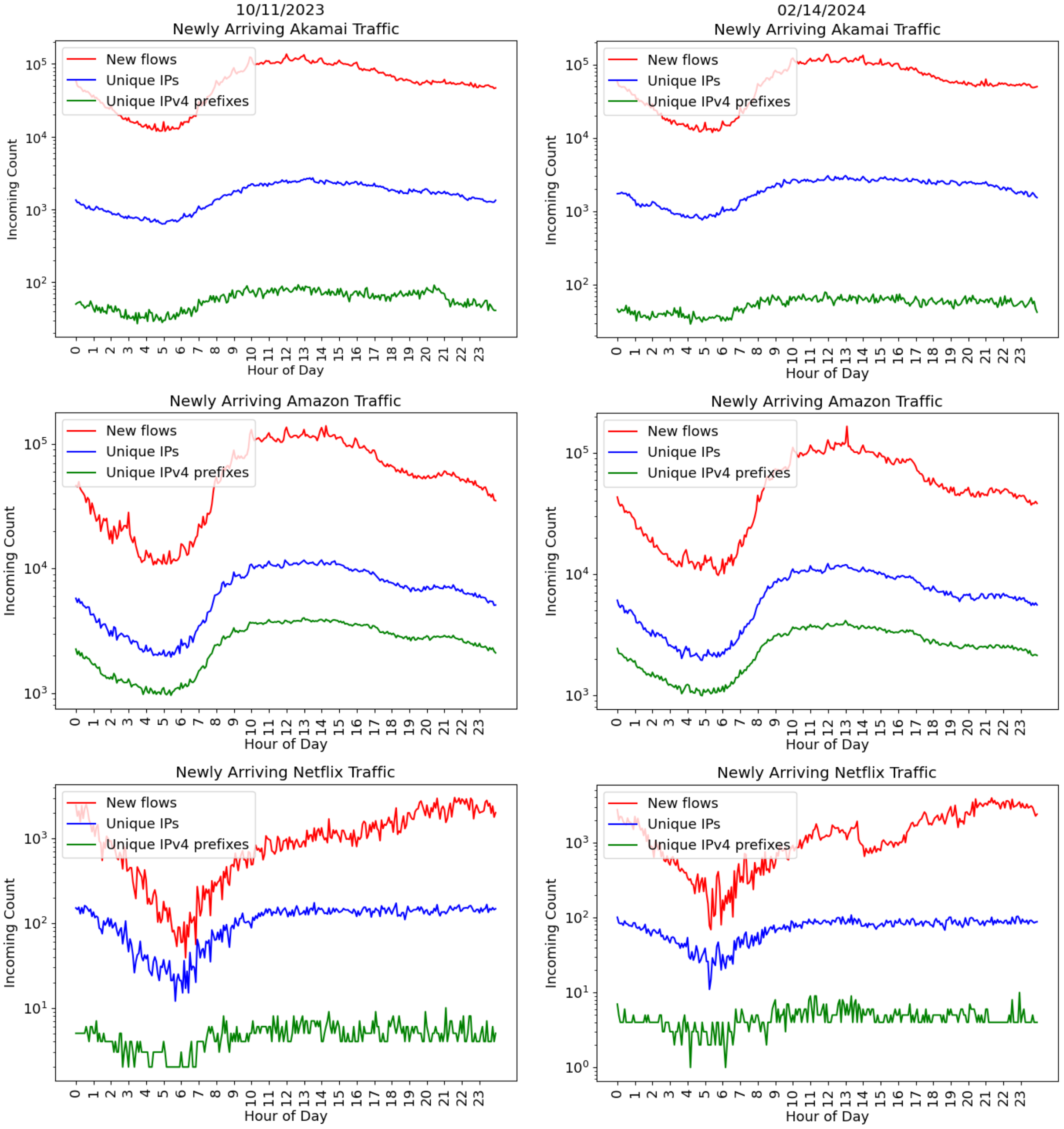

Figure 22: Temporal pattern of incoming source IP and IPv4 /24 prefix counts of Akamai/Amazon/Netflix flows per 5-minute window

An additional observation from Figure 22 is regarding the small number of IPv4 addresses and prefixes associated with incoming Netflix traffic. This smaller number of IP addresses and prefixes (compared to all other groups of flows) allows us to easily investigate the locality of flows to IP addresses and IP prefixes. This concept of locality refers to the tendency for flows to be focused to specific IP addresses or prefixes, where high locality implies each IP address or IP prefix is associated with many flows.

While Figure 22 shows that there is around an order of magnitude gap between each line for Netflix, further examination into the specific IP addresses used by Netflix shows a more detailed look into the high degree of locality among flows across IP addresses. More specifically, a majority (approximately 70%) of flows are associated with 12 source IP on both 10/11/2023 and 02/14/2024. Furthermore, these flows tend to have a higher mean duration (two to four minutes on average) and throughput (0.9 to 1.6 million bytes per second), which suggests that these specific IP addresses are primarily responsible for video streaming.

This view into locality can be further generalized by observing the IPv4 /24 prefixes. At this level, out of the 35 to 40 total unique IPv4 /24 prefixes used by Netflix on 10/11/2023 and 02/14/2024, two IPv4 prefixes together account for around 93% of incoming flows, with similar ranges of mean values of durations and throughputs as in the IP address view.

These findings suggest that certain IP addresses and IP prefixes serve different roles in content distribution by providers. For instance, certain IP addresses or prefixes having higher average durations and throughputs may indicate use for video streaming, in the case of Netflix.

Flow Size Analysis. Figure 22 shows the distribution of observed flow sizes of incoming flows from each AS over each snapshot, along with a CDF from a sample snapshot. Figure 22 shows that Akamai and Amazon have very similar median flow sizes and very consistent ranges as well. Netflix flows, on the other hand, send much more data, up to multiple orders of magnitude more data.

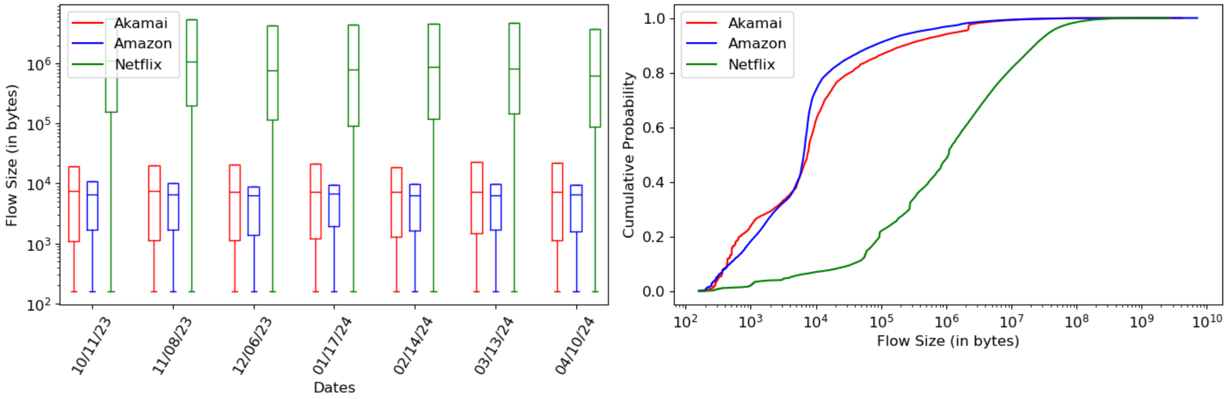

Figure 23: Summary distribution of flow size for incoming Akamai/Amazon/Netflix flows across all snapshots (left) and CDF of flow size for incoming Akamai/Amazon/Netflix flows in a sample snapshot of 10/11/2023 (right)

Flow Duration Analysis. Continuing the investigation, Figure 23 displays the distribution of incoming flow durations among the different ASes. From this figure, we see that Akamai flows are the shortest on average, compared to Amazon and Netflix. Amazon and Netflix have similar upper bounds in their flow durations, though Netflix flows last the longest on average.

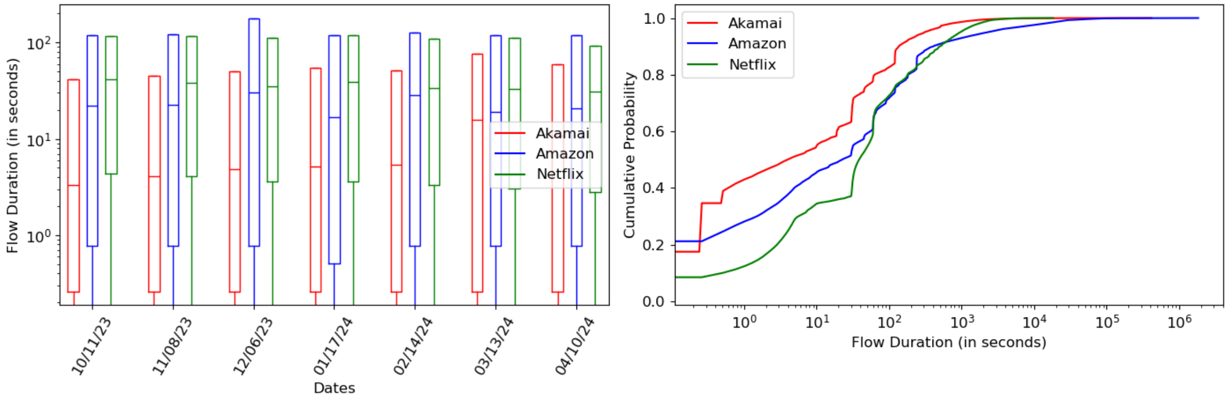

Figure 24: Summary distribution of flow duration for incoming Akamai/Amazon/Netflix flows across all snapshots (left) and CDF of flow size for incoming Akamai/Amazon/Netflix flows in a sample snapshot of 11/08/2023 (right)

Flow Throughput Analysis. Finally, Figure 24 shows the distribution of observed flow throughputs of incoming Akamai, Amazon, and Netflix flows, along with a CDF from a sample snapshot. This figure shows that both CDNs have similar overall throughput, and Netflix has the highest throughput, with median values around an order of magnitude above the others.

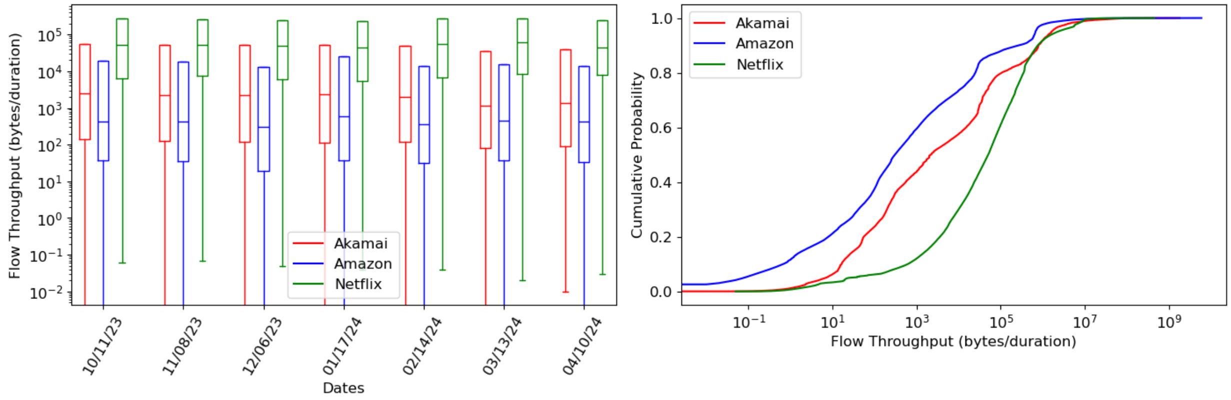

Figure 25: Summary distribution of flow throughput for incoming Akamai/Amazon/Netflix flows across all snapshots (left) and CDF of flow size for incoming Akamai/Amazon/Netflix flows in a sample snapshot of 11/08/2023 (right)

Key Takeaways:

From our autonomous system-level analysis, we observe the following:

- Unlike previous levels of traffic analysis, there is no consistently dominant flow type at this level. In terms of total volume, Amazon and Akamai often deliver data to the UO network in similar volumes (Figure 20).
- Akamai and Netflix display more consistent data transmission behaviors compared to Amazon, which shows high variance and bursty behavior. (Figure 20)
- Given that Akamai and Amazon make connections at very similar rates and only Amazon displays bursty behavior, it can be said that the characteristics of Amazon flows fluctuate based on demand. (Figure 21)

- The IP address space utilized by each AS is very different, with new Amazon flows having up to an order of magnitude more unique IP addresses than Akamai and two orders of magnitude more unique IP addresses than Netflix. (Figure 22)
- For Netflix in particular, we observed the locality between incoming flows, IPv4 addresses and IPv4 /24 prefixes, and found a high degree of locality at both levels:
    - At the IP address level, 70% of flows are associated with just 12 source IP addresses.
    - At the IPv4 /24 prefix level, 93% of flows are associated with just two prefixes.
    - These IP addresses and IP prefixes have high average durations and throughputs, which indicates that these specific IP groups are responsible for video streaming.
        - More generally, specific IP addresses and prefixes can be observed to play distinct roles in content distribution.
- Despite the differences in data volume, the throughputs of the CDNs are very similar overall. (Figure 25)
- Netflix flows can be characterized as having very high throughput over sustained periods. These flows are both longer lasting than flows from the CDNs (Figure 24) and carry significantly more data (Figure 23).
    - Of all previously observed subsets of incoming flows (including aggregate-level and port-level analysis), Netflix flows have the highest throughput. (Figure 25)
    - Overall, this shows that Netflix has much higher bandwidth requirements than other types of traffic and is one of the most demanding external entities that interact with the university network.

Chapter 4: Prior Work

Network measurement is often utilized to gain insight into some aspects of operational networks (e.g. routing [7], interconnections [8], geographic location [9]) which is not feasible through simulation-based studies (e.g. [10]). Traffic measurement, analysis and profiling is a subarea of network measurement that is commonly used to evaluate performance of the network, detect an anomaly or other related events, or provision resources.

Prior studies on traffic profiling that use NetFlow data can be broadly divided into two groups.

One such group of studies utilizes NetFlow by observing NetFlow data in aggregate. A study from the early 2000s used NetFlow data to identify the characteristics of top applications at the time, primarily looking at HTTP, FTP, and DNS, among others. By comparing traits such as the average throughput and flow size of these subsets of flows, they made conclusions about the similarities and differences in the observed characteristics of these applications [11]. Another study utilizes aggregate NetFlow data differently, observing the ASNs most prevalent in incoming Internet traffic to reveal which content providers interact with an organization the most. They then dive deeper into the specifics of traffic locality and achievable throughput of content providers through additional investigation beyond what NetFlow data alone can relay [12].

A second group of studies moves away from characterizing aggregate qualities and instead considers the unique characteristics of a flow, such as the ports, IP addresses, or sizes of the flow as part of its dataset. These studies use machine learning models to identify things such as “user traffic profiles” [4][13] and malicious attack patterns [14] through characteristics of flows such as packets per second and duration of individual flows, or consecutive groups of flows.

In this thesis, we expand upon the first listed group of studies and similarly utilize higher-level characteristics of NetFlow data in aggregate by introducing additional analysis into observed aggregate Internet traffic. While Lieu and Huebner [11] utilize NetFlow to make similar observations regarding the cumulative distribution functions of several flow-level characteristics, in the 20 years since that study was conducted, characteristics of Internet traffic have certainly changed significantly, and we aim to observe how trends have changed. On the other hand, Yeganeh [12] uses NetFlow in the context of content providers, extracting information regarding the effectiveness of major content providers through analysis of structure and locality. While both studies do well in informing about characterizing Internet traffic using statistical analysis, our work provides further context into incoming traffic by incorporating a temporal analysis along with statistical analysis. This additional strategy helps us define more qualities of an organization’s network traffic, primarily providing insight into temporal trends that were previously not observed.

We can observe trends occurring at the highest aggregate level, but at smaller and more precise levels as well. Through our investigation, we have found that, at these varying levels, temporal and statistical analysis can provide helpful insight into the effort to characterize network traffic.

Chapter 5: Summary

In this research, our primary goal was to explore the UO campus network and extract information regarding incoming Internet traffic at various levels. In doing this, we posed several key questions:

- What are the typical traffic patterns of Internet traffic at each level?
- What transport protocols are most used in our network?
- What are the common ports and identifiable applications that use our network?
- What are the characteristics of the autonomous systems that use our network the most?
- Do these subsets of Internet traffic display recognizable characteristics that can define standard use?

To answer these questions, we implemented a multi-level approach, analyzing incoming network traffic at multiple levels.

- For the highest-level analysis at the aggregate level, we analyzed all incoming flows to examine the prevalence of the specific transport protocols that make up this incoming Internet traffic, finding that TCP was the most prevalent.
- We then broke down the TCP traffic to observe the characteristics of the ports that make up TCP traffic.
    - We observed the qualities of the two popular application-layer protocols that are primarily used in web traffic, HTTPS and HTTP.
    - We zoomed in on specific applications that were identifiable by their port number, to gain insight into the characteristics of these specific applications.
- We found that HTTPS was the most popular port and broke down HTTPS traffic down further into the autonomous systems that utilize HTTPS for our final investigation.

Altogether, the results of this analysis give some insight into establishing some standard modes of operation of the UO network. From this work, the following characteristics of incoming network traffic at the UO have been established:

At the aggregate level, we have that:

- The network has the most demand from 12 pm to 3 pm and has the least demand from 3 am to 8 am.
- TCP traffic dominates the UO network at all points of the day.

When focusing on TCP web traffic, we see that:

- HTTPS traffic makes up a large majority of observed flows at this level.
- All TCP web traffic sends data at very similar rates, as seen by the similar throughput that HTTPS, HTTP, and other miscellaneous TCP flows share.
    - This means that HTTPS flows tend to deliver more bytes due to the flows lasting longer on average.

As for specific applications identifiable by port at this level, we see that:

- IMAP accounts for the most incoming data of the three observed applications, additionally exhibiting lots of variance, sending data in large bursts.
- The flow sizes of the three observed applications (APNS, IMAP, and Spotify) exhibit similar flow sizes. This helps us identify different classes of applications:
    - Applications that prefer longer, sustained communication (APNS).
    - Applications that prefer shorter, high-throughput communication (IMAP and Spotify).
- Since applications like streaming applications may have largely variable demands from day to day, associated flows can be expected to have more variance from day to day.

Lastly, from our investigation at the ASN level, for connections utilizing HTTPS, we find that:

- There is no consistently dominant flow subset at this level, as Amazon and Akamai are both seen to deliver comparable amounts of data on certain snapshots.
- Of the three ASes observed (Amazon, Akamai, and Netflix), Amazon flows are much more bursty and high variance, while Akamai and Netflix deliver data at more consistent rates.
- The throughputs of the CDNs (Amazon and Akamai) are very similar overall, despite their differences in data volume.
- Netflix, when compared to the CDNs (and all other observed subsets of incoming traffic in general), exhibits flows with the highest throughput.
    - This suggests that Netflix is one of the external providers that place the highest demand on the network with their connections.
- Amazon has the largest number of unique IP addresses of the observed ASes, with up to an order of magnitude more unique IP addresses per 5-minute window than Akamai, and two orders of magnitude more than Netflix.
- Netflix flows display a high degree of locality of flows to IP addresses and IPv4 /24 prefixes.
    - Two IPv4 prefixes are seen to account for around 93% of incoming connections on observed days, and 12 specific IP addresses account for around 70% of incoming connections on observed days.
    - Both groups are observed to have high average durations and throughputs, which indicates that these specific IP groups are primarily used for video streaming.

This research is crucial for organizations with networks that need to provide consistent and safe Internet connectivity to their users, such as the University of Oregon. By identifying characteristics such as which protocols are most widely used or how popular applications utilize the network, the organization has more insight into the characteristics of a network at these varying levels. With this information, the unique needs of a network can be better met to handle predicted volumes of traffic, while any significant deviations from standard trends can tip off administrators to potential attacks.

For instance, we have identified that the average Netflix flow has the highest throughput of all observed subsets of traffic. With this context, future flows with significantly large throughputs comparable to Netflix flows may indicate that this flow belongs to a similar class, such as another video streaming platform. On the other hand, if this flow is unidentifiable by port or ASN, or exhibits any qualities outside of the norm for an identified application, it may be indicative of a malicious flow attempting to overload the capacity of the network and can be investigated further.

In the future, this work can be expanded to further characterize specific applications and content providers by exploring the underlying causes for the difference in throughput observed for these different subsets of flows.

Alternatively, deeper investigation can be conducted on analyzing characteristics of IPv4 and IPv6 flows and their various IP prefixes. From these profiling efforts, one could potentially discover important information about characterizing a network, including the degree of locality of incoming subsets of Internet traffic.

Additionally, this work can be extended to see how effective this multi-level profiling strategy is in detecting days on which an attack was observed by a network. When a network is attacked by malicious entities, it would be valuable to observe which levels of depth are best able to display this information.

This work can also be repeated in the future to observe what characteristics of the UO network stay consistent and what characteristics change over time. Any observations from these changes can inform future decisions regarding network management.

Lastly, this work can be extended by exploring the possibilities of leveraging machine learning to see how effectively models can be trained to detect anomalies in the UO network and see which levels of analysis are most effective in this effort.

Bibliography

[1] Cisco. “Cisco Annual Internet Report (2018-2023) White Paper,” March 9, 2020. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html.

[2] B. Claise, B. Trammell, E. Zurich and P. Aitken, *Specification of the ip flow information export (ipfix) protocol for the exchange of flow information (rfc 7011)*, September 2013, [online] Available: https://tools.ietf.org/search/rfc7011.

[3] B. Gorman, “TCP vs UDP: Differences between the protocols,” February 23, 2023. [Online]. Available: https://www.avast.com/c-tcp-vs-udp-difference

[4] T. Bakhshi and B. Ghita, "User traffic profiling," *2015 Internet Technologies and Applications (ITA)*, Wrexham, UK, 2015, pp. 91-97, doi: 10.1109/ITechA.2015.7317376.

[5] C. Cohen, “What is Port 993,” November 12, 2023. [Online]. Available: https://www.cbtnuggets.com/common-ports/what-is-port-993

[6] Cloudflare. “What is a CDN?” [Online]. Available: https://www.cloudflare.com/learning/cdn/what-is-a-cdn/

[7] B. Yeganeh, R. Durairajan, R. Rejaie, W. Willinger, "How Cloud Traffic Goes Hiding: A study of Amazon’s Peering Fabric", Proceedings of the Internet Measurement Conference, pp.202-216, 2019, doi: 10.1145/3355369.3355602.

[8] R. Motamedi, B. Yeganeh, B. Chandrasekaran, R. Rejaie, B. M. Maggs, W. Willinger, "On Mapping the Interconnections in Today’s Internet," IEEE/ACM Transactions on Networking, vol. 27, no. 5, pp. 2056-2070, Oct. 2019, doi: 10.1109/TNET.2019.2940369.

[9] A. Rasti, N. Magharei, R. Rejaie, W.Willinger, “Eyeball ASes: from geography to connectivity,” IMC ’10: Proceedings of the 10th ACM SIGCOMM conference on Internet measurement, pp.192-198, November 2010, doi: 10.1145/1879141.1879165.

[10] S. Bajaj, L. Breslau, D. Estrin, K. Fall, S. Floyd, P. Haldar, M. Handley, A. Helmy, J. Heidemann, P. Huang, S. Kumar, S. McCanne, R. Rejaie, P. Sharma, S. Shenker, K. Varadhan, H. Yu, Y. Xu, and D. Zappala, “Virtual internetwork testbed: Status and research agenda”, Technical Report 98—678, USC Computer Science Dept, July 1998.

[11] D. Liu and F. Huebner, "Application profiling of IP traffic," 27th Annual IEEE Conference on Local Computer Networks, 2002. Proceedings. LCN 2002, Tampa, FL, USA, 2002, pp. 220-229, doi: 10.1109/LCN.2002.1181787.

[12] B. Yeganeh, R. Rejaie, and W. Willinger, "A view from the edge: A stub-AS perspective of traffic localization and its implications," 2017 Network Traffic Measurement and Analysis Conference (TMA), Dublin, Ireland, 2017, pp. 1-9, doi: 10.23919/TMA.2017.8002900.

[13] H. Jiang, Z. Ge, S. Jin, and J. Wang, “Network prefix-level traffic profiling: Characterizing, modeling, and evaluation,” 2010 *Computer Networks*, Volume 54, Issue 18, pp. 3327-3340, https://doi.org/10.1016/j.comnet.2010.06.013

[14] C. Kemp, C. Calvert, and T. Khoshgoftaar, "Utilizing Netflow Data to Detect Slow Read Attacks," *2018 IEEE International Conference on Information Reuse and Integration (IRI)*, Salt Lake City, UT, USA, 2018, pp. 108-116, doi: 10.1109/IRI.2018.00023.