Evaluation of the Effectiveness of Small Aperture Network Telescopes as IBR Data Sources

Submitted in fulfilment
of the requirements of the degree of

Doctor of Philosophy of Science

of Rhodes University

Stones Dalitso Chindipha

Grahamstown, South Africa
December 2021

Abstract

The use of network telescopes to collect unsolicited network traffic by monitoring unallocated address space has been in existence for over two decades. Past research has shown that there is a lot of activity happening in this unallocated space that needs monitoring as it carries threat intelligence data that has proven to be very useful in the security field. Prior to the emergence of the Internet of Things (IoT), commercialisation of IP addresses and widespread of mobile devices, there was a large pool of IPv4 addresses and thus reserving IPv4 addresses to be used for monitoring unsolicited activities going in the unallocated space was not a problem. Now, preservation of such IPv4 addresses just for monitoring is increasingly difficult as there is not enough free addresses in the IPv4 address space to be used for just monitoring. This is the case because such monitoring is seen as a 'non-productive' use of the IP addresses. This research addresses the problem brought forth by this IPv4 address space exhaustion in relation to Internet Background Radiation (IBR) monitoring.

In order to address the research questions, this research developed four mathematical models: Absolute Mean Accuracy Percentage Score (AMAPS), Symmetric Absolute Mean Accuracy Percentage Score (SAMAPS), Standardised Mean Absolute Error (SMAE), and Standardised Mean Absolute Scaled Error (SMASE). These models are used to evaluate the research objectives and quantify the variations that exist between different samples. The sample sizes represent different lens sizes of the telescopes. The study has brought to light a time series plot that shows the expected proportion of unique source IP addresses collected over time.

The study also imputed data using the smaller /24 IPv4 net-block subnets to regenerate the missing data points using bootstrapping to create confidence intervals (CI). The findings from the simulated data supports the findings computed from the models. The CI offers a boost to decision making. Through a series of experiments with monthly and quarterly datasets, the study proposed a 95% - 99% confidence level to be used. It was known that large network telescopes collect more threat intelligence data than small-sized network telescopes, however, no study, to the best of our knowledge, has ever quantified such a knowledge gap. With the findings from the study, small-sized network telescope users can now use their network telescopes with full knowledge of gap that exists in the data collected between different network telescopes.

Acknowledgements

Throughout this research study, I have received a great deal of support and assistance from several individuals. Foremost, I would like to give glory to God, who has been a constant presence and guide when all seemed to be lost. I owe Him everything as far as this research is concerned, and none of this would be possible without Him.

As far as individuals go, firstly, I would like to thank my supervisor, Professor Barry Irwin, whose expertise was invaluable in helping me understand the parameters of this research. His knowledge base of the datasets used and insightful feedback pushed me to sharpen my thinking and brought my work to a higher level. A token of appreciation to Dr Alan Herbert for his technical support as my co-supervisor. You really came through for me at a time when I needed a well-balanced life. Your holistic approach to supervision kept me sane.

The PhD journey would not have been taken if not for my then-girlfriend, now wife, Akhona Ayabonga Chindipha. She saw a future that I did not see, opportunities that I could not have considered. She has had more faith in me than I did in my abilities. When I hear her talk of her husband, I often think she talks of someone I do not know. Thank you for your patience, companionship, prayers and encouragement when I needed it the most. In addition, I would like to thank my family, particularly, my mother for her prayers, faith in me, wise counsel and sympathetic ear. You were always there for me.

I want to acknowledge all my colleagues from the Red Lab for the help and guidance they gave to problems I faced. I want to thank the Rhodes University Computer Science Department for providing the working environment that I needed to carry out this research.

Finally, I could not have completed this dissertation without the support of Professor Tony Booth, who offered me funding to carry out this study. Further financial support came from the Distributed Multimedia Centre of Excellence at Rhodes University. The author acknowledges that opinions, findings and conclusions or recommendations expressed here are those of the author and that none of the above-mentioned sponsors accepts liability whatsoever in this regard.

Contents

List of Figures

List of Tables

List of Acronyms

ACK	Acknowledge
AMAPS	Absolute Mean Accuracy Percentage Score
CI	Confidence Interval
CAIDA	Centre for Applied Internet Data Analysis
CERT	Computer Emergency Response Team
CIDR	Classless Inter-Domain Routing
CPS	Cyber-Physical Systems
DPORT	Destination Port
DSTIP	Destination Internet Protocol Address
DDoS	Distributed Denial of Service
IANA	Internet Assigned Numbers Authority
IBR	Internet Background Radiation
ICCP	Inter-Control Center Communications Protocol
ICMP	Internet Control Message Protocol
IDF	Inverse Document Frequency
IoT	Internet of Things
IP	Internet Protocol
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6
IQR	Interquartile Range
IR-TMT	Information Retrieval and Text Mining Techniques
ISP	Internet Service Provider
JD	Jaccard Distance
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error

MASE	Mean Absolute Scaled Error
MSE	Mean Square Error
PRNG	Pseudo Random Number Generator
RIRs	Regional Internet Registries
RMSE	Root Mean Square Error
SAMAPS	Symmetric Absolute Mean Accuracy Percentage Score
SANS	SysAdmin, Audit, Network, and Security
SMAE	Standardised Mean Absolute Error
SMASE	Standardised Mean Absolute Scaled Error
SMAPE	Symmetric Mean Absolute Percentage Error
SRCIP	Source Internet Protocol Address
SYN	Synchronize
TF	Term Frequency
TF-IDF	Term Frequency-Inverse Document Frequency
TCP	Transmission Control Protocol
UDP	User Datagram Protocol

1

Introduction

Internet Background Radiation (IBR) is defined as non-productive data packets on the Internet, which target unused IP addresses, or ports where there is no network device set up to receive them (Cooke *et al.*, 2004; Pang *et al.*, 2004; Wustrow *et al.*, 2010; Guillot *et al.*, 2019). In theory, no traffic should ever arrive at such an IPv4 address, and so such traffic is marked as an anomaly and thus recorded and analysed (Hunter *et al.*, 2013; Guillot *et al.*, 2019; Richter and Berger, 2019). IBR data is typically collected by devices known as network telescopes (Czyz *et al.*, 2013; Irwin, 2013; Richter and Berger, 2019; Torabi *et al.*, 2020). The basis of a network telescope is to monitor these IP address blocks on networks that have no services running on them (Hunter *et al.*, 2013; Fachkha *et al.*, 2017; Bou-Harb *et al.*, 2018). The value of network telescopes has been dealt with thoroughly by other researchers to the point that its significance to cybersecurity research cannot be overemphasised (Moore *et al.*, 2004; Irwin, 2013; Bou-Harb *et al.*, 2018). Often this traffic shows evidence of either malicious activity or poor configuration (Pang *et al.*, 2004; Nkhumeleni, 2014; Bou-Harb *et al.*, 2014; Richter and Berger, 2019). The poor configuration could either be temporary or permanent (Nkhumeleni, 2014; Fachkha *et al.*, 2017).

Given the fact that there are no legitimate hosts in an unused address block (Polakis *et al.*, 2011; Irwin, 2012; Czyz *et al.*, 2013; Torabi *et al.*, 2020), traffic must originate as the result of poor configuration, back-scatter from spoofed source addresses, or scanning from worms and other probings (Cooke *et al.*, 2004; Wustrow *et al.*, 2010; Richter and Berger, 2019). What makes IBR very critical is its ability to provide an early-warning detection mechanism for new threats and attacks (Harder *et al.*, 2006; Bou-Harb *et al.*, 2014; Chatziadam *et al.*, 2014; Fachkha and Debbabi, 2016).

The reason for the growing interest in the field of network telescopes, among other threats, is partially attributed to the changing threat landscape (Shannon and Moore, 2004; Hunter *et al.*, 2013; Bou-Harb *et al.*, 2016). The change in the threat landscape is caused by an increase in self-propagating malicious viruses such as Witty Worm (Chen and Bridges, 2017; Fachkha *et al.*, 2017; Torabi *et al.*, 2020), Conficker Worm, GoBrut and Jokeroo ransomware (Irwin, 2013; Zhang *et al.*, 2015; Thomas and Galligher, 2018; Ochieng *et al.*, 2019; McElhinney and Curran, 2020) and the Mirai Botnets (Bertino and Islam, 2017). This change in the threat landscape makes IBR an effective method for analysing and quantifying Internet security phenomena.

1.1 Background

Levin and Schmidt (2014) explained that the modern Internet has long relied on the existence and use of IPv4 addresses. With fast-paced technology and new devices needing IP addresses, exhaustion of the IPv4 address blocks globally has made it nearly impossible for organisations to gain access to large blocks of IPv4 addresses to use for perceived 'unproductive' purposes (Arlot and Celisse, 2010; Irwin, 2011; Durand *et al.*, 2011; Nyirenda-Jere and Biru, 2015; Dainotti *et al.*, 2016; Beeharry and Nowbutsing, 2016). In their work, both Levin and Schmidt (2014) and Dainotti *et al.* (2016) stated that IP addresses are allocated according to regions and because of that, some regions have already exhausted their allocated lot of IPv4 addresses. According to Mamushiane *et al.* (2021), among other researchers, Africa is among those continents whose IPv4 blocks have run out and adoption to IPv6 is still proving to be difficult.

Dainotti *et al.* (2014, 2016) confirmed through two separate studies that the commercialisation of IPv4 addresses and the emergence of the Internet of Things (IoT) has rapidly led to the exhaustion of IPv4 addresses. Due to this development, priority is given to production systems that provide a service, thus reducing available IPs for threat intelligence gathering. This has left most organisations with little to no option of using their

IP addresses for purely threat intelligence gathering purposes through the use of network telescopes. Of interest to this study is how IPv4 have been used to threat intelligence gathering and other security applications. With a practical solution presented, the use of IPv4 can continue to provide this valuable feedback to the security community. IPv4 address block usage also has significant information security applications, among which include supporting detection of address squatting, informing host reputation systems, and active measurement experiment design (Chatziadam *et al.*, 2014; Dainotti *et al.*, 2016). With the exhaustion of IPv4 addresses, passive measuring of security threat activities has been one of the activities that have been negatively affected (Chatziadam *et al.*, 2014; Dainotti *et al.*, 2014).

1.2 Problem Statement

This research study provides an evaluation study to address the IPv4 shortage by showing that it is possible to model a /24 IPv4 network block using smaller subnet samples obtained from a /24 IPv4 address block. These smaller samples represent different sizes of the smaller network telescope. This addresses the problem brought forth by the exhaustion of IPv4 address blocks in relation to IBR's 'unproductive' use. The problem in this case being the inability of network telescope users to successfully configure a small-sized network telescope and quantify how representable their small network telescopes are to the larger network telescopes. It is the lack of research studies and tools that can quantify the data collected from the small-sized network telescopes and compare it to the perceived larger network telescopes.

There are a number of questions being addressed by this research study but at the top of this list is the number of unique Destination IP (DSTIP) addresses a user needs. This study is designed to evaluate the least possible number DSTIP addresses that a user needs to use in setting up a small-sized network telescope. The aim of setting up a small-sized network telescope is to assess how representative the events and behaviour identified in a larger network telescope can be observed in small-sized network telescope. By sub sampling the baseline data (/24 IPv4), this study will create samples which are representative of small-sized network telescopes. The identified subnet sample (which represents a smaller network telescope) ought to offer a reasonably high degree of accuracy and a high level of confidence in the data to the user. Those samples that offer high confidence and high accuracy scores will be given a high priority as best representatives

of the baseline. With such scores the study will offer recommendations on which of the available samples is the best fit.

In order to address this problem, this research study investigates the origin of the network traffic, Destination IP (DSTIP) address and Destination ports (DPORT) from a /24 IPv4 net-block which acted as the benchmark dataset for this study. The source host's IP addresses are referred to as Source IP (SRCIP) throughout this study.

1.3 Research Goals

This research was conducted with the overall objective of evaluating the effectiveness of small-sized network telescopes as Internet Background Radiation data source. The researchers are fully aware that a sample of the baseline dataset will not fully replace the baseline data. However, finding an alternative that offers a high level of confidence in the samples drawn out from the baseline would be a better fit as compared to completely eliminating the use of such powerful technology in threat intelligence gathering. In other words, working with a smaller network telescope is better than not using one at all. This study aims to quantify such differences in the data collected by a smaller network telescope and a large one. Essentially the primary question was: if a sample is taken from a baseline dataset, how representable can the sample be in relation to its baseline data? In order to achieve this, this study primarily investigates the following points:

1. Assess if there is a continual direct relationship between the number of unique source IP (SRCIP) addresses observed against the number of unique destination IP (DSTIP) addresses used to collect data when normalised.

2. Compute the time frame needed to acquire specific proportions of the unique source IP addresses from the baseline data. Compare these with the proportions that each sample contains.

3. Identify how accurate a small-sized network telescope lens is at representing /24 IPv4 network telescope. This is in terms of the representativeness of the threat intelligence data collected by each network telescope. This will be done by comparing and contrasting the results computed from the data samples (which represent a smaller lens) with that of the /24 IPv4 baseline dataset.

4. Evaluate the differences that exist when the IPv4 addresses in the network sensors are randomly selected compared to when the IPv4 addresses are selected in 'traditional' blocks aligning to contiguous CIDR subnetting.

1.4 Research Approach

This research was conducted with the overall objective of evaluating small aperture network telescopes as a tool for threat intelligence gathering using IBR data. The basis for this research was formed by understanding that relatively small-sized network telescopes have been previously deployed for threat intelligence gathering (Harder *et al.*, 2006; Wustrow *et al.*, 2010; Benson *et al.*, 2015; Zeghache and Yacine, 2020). Other researchers have presented this work but what has not been presented yet, to the best of our knowledge, is how representative these small-sized network telescopes are at mimicking key attributes present in the baseline network telescope. In this research study, baseline network telescopes were configured to using a /24 IPv4 address block. More details about the work done on small network telescope can be found in **Section 2.3**.

The final study is data-driven, as such it presents the results of analysis using data collected over three months from January - March 2021. The three months were collected from a series of distinct network telescopes, which formed both monthly and quarterly analyses in the data analytics chapters. All the data used was from three of the five network telescopes maintained by the Rhodes University Security Networks Research Group (SNRG). The data is summarised using data dictionaries in **Section 4.1**. Tools and techniques were developed for the processing and analysis of the data.

During this research, exploratory data analysis was performed on the telescope data to understand the composition of the datasets. Due to the nature of the research questions being addressed, the study required the use of sampling techniques. Thus, random and sequential sampling techniques were used to come up with data samples. The size of the IPv4 subnets for the network telescopes was used to determine the different sizes of the data sample. Included in this research are the mathematical models designed to quantify the differences between the baseline datasets and the data samples drawn from them. The models were inspired by research done by Hyndman and Koehler (2006) in their work on forecasting time series data. The work done by Hyndman and Koehler (2006) is presented in **Section 3.9**. This research found limitations in using the models that Hyndman and Koehler developed, and as such, derivations were made to suit the needs of this research.

Apart from sampling the /24 IPv4 address blocks, the study also imputed data using the smaller samples of /24 IPv4 address blocks to regenerate the missing data points using a technique called bootstrapping. The study focused on simulating the samples of IBR data that were taken from the baseline datasets. The reason for this approach was to attempt to reproduce the baseline dataset from its data samples using bootstrapping. An explanation on the rationale of this research approach is presented in **Section 5.1**. Different samples were used to bootstrap IBR data and confidence intervals derived from these simulations by using the mean as the data's population parameter of interest. Firstly, data were sampled into different subnet equivalents (sample sizes that mimic the size of subnets in IPv4 address blocks). In order to reproduce the number of data points found in the baseline datasets, an hour was used to represent a single data point (see **Section 5.2**). Thus, the number of data points was equated to the number of hours found in the dataset under study. Where the monthly dataset is used, the baseline datasets have 744 data points if the monthly data contains 31 days. The same approach was used for quarterly datasets. More details are presented in **Section 5.5**.

The different subnet equivalents (data samples) were treated using these data points to simulate the same number of data points. This was done to mimic a scenario where a user would not have access to baseline data. The study attempts to quantify the levels of confidence each bootstrap sample would give to the data user (read network telescope user) should the user happen to have only such a handful of destination IP to be used for passive monitoring of the network.

The mathematical models developed in this study proved to have multiple applications outside the information security field. So **Chapter 7** of this study is dedicated to exploring some of those practical applications in other fields. However, considering that this research study was conducted with the intended use in networks and security space, the discussions in this document focus on the use of mathematical models to quantify the differences in IBR data. The testing on the performance of the mathematical models happened in four phases. Firstly, the study validated the need for the proposed changes to the already existing as the underlying conditions with which Hyndman and Koehler (2006) did not accommodate such requirements. This required testing the model against Hyndman and Koehler's model using the same data. Secondly, the models were tested against monthly datasets, which were later followed by quarterly datasets. All the models and the test conducted can be found in **Chapter 6** where more details are presented.

1.5 Research Scope and Limitations

The scope and limitations of this research was inherent in the data that was used. The network telescope sensors are a set of passive network monitoring devices designed to record both active and passive IBR. This study used network telescopes that were configured to do passive monitoring i.e. only receive incoming traffic without responding to it. The analyses and evaluations in this research were limited to TCP Internet traffic. The choice of focusing on TCP was arrived at due to its large volume collected in all of the network telescopes under study. The study has presented an exploratory data analysis of UDP traffic but no further analysis is done on this protocol. The data for UDP, ICMP and GRE protocols is presented but due to its small volume as compared to TCP, it was not used for any evaluation to address the research questions. In addition, this study did not accommodate traffic from IPv6 address blocks as at the time of study the network telescopes did not collect sufficient data to be used. Thus all analysis done was based on IPv4 address blocks with /24 net-blocks being used as the baseline data for all network telescopes.

1.6 Document Conventions

In the remainder of this document, as a general rule, a number of conventions can be seen. The conventions apply to certain words, segments of text and numbers are represented with different fonts, sizes and formatting to emphasise the presence of a keyword, phrase, equation (model) or name that has been used sequentially throughout the document. This section describes those conventions used as a guide.

Models: Mathematical models are centred and given equation numbers on the right-hand margin of the page that they are used in. For example:

$$E = mc^2 \tag{1.1}$$

Number formatting: Numbers in this document are rounded to two significant figures after the decimal point. A thousand separator used is a comma (,) and decimals are given after a period (.). For example: **123,456.89**

IP Address: This research focuses primarily on IPv4 addresses. As such, when the term IP address is used, it refers to IPv4 addresses. The source and destination IP addresses are abbreviated as follows:

- Source IP: SRCIP
- Destination IP: DSTIP

Sample naming: The size of a data sample is defined by the number of unique destination IP addresses found in the specified subnet and their names are presented in italics. Sequential samples have the name subnet following the size of the sample they represent while random samples have the name subnet equivalent following the size of the sample they represent. Preceding the size of the sample for sequential sampling is forward-slash (/) as commonly used in Classless Inter-Domain Routing (CIDR). On the other hand, preceding a random sample is a slash with a subscript e ($/_e$). For example:

- Sequential sample of size 128: */25 subnet*
- Random sample of size 128: $/_e$*25 subnet equivalent*

Network Telescope naming: All network telescopes are named after the first octet with which their address block represents. This is followed by a forward slash (/) and the value of 8 to show that the naming represents a /24 IPv4 address blocks. The names are presented in bold throughout the document. The other details of the IP addresses in the network telescope are hidden for privacy. For example, using /24 IPv4 address blocks naming convention a network telescope to 146 will be presented as: **146/8**

Telescopes within 196/6 are named differently because Rhodes has more than one telescope within this range. Thus to distinguish them, an additional feature is added to the naming convention. This is presented as: **196-A/8**

Dataset naming: Datasets are named after the network telescope from which they were collected followed by the month and year of data collection. Sometimes datasets are described after the network telescope they were collected from when dealing with monthly analysis. For example, a dataset belonging to **146/8** network telescope collected in January 2021 will be presented as: **146/8-012021**

Bootstrap Sample naming: Bootstrap samples are named after the network telescope from which the data was collected followed by the name of their subnet equivalent. Between the name of the network telescope and the name of the subnet equivalent, there is a dash (-). For example bootstrap sample belonging to **146/8** network telescope collected from a $/_e24$ subnet equivalent in January 2021 will be presented as: **146/8-012021- $/_e$24**

When presenting tables or figures belonging to a quarterly analysis, the following naming convention is used: **146/8-2021 - [Jan - Mar]:**

URLs: When a URL pertaining to websites or organisations is mentioned, it is given as a footnote on the page of mention. Firstly, the URLs used provided up to date information on the topic that could not be found in published articles. The second reason was the need to minimise the break in the flow of the document, and to allow readers quick access to the information pertaining to such work.

1.7 Document Structure

The remainder of this document is structured as follows:

Chapter 2 presents the first half of the literature review. The intention of this part of the literature review is to provide the reader with a good understanding of the background material upon which this research is based. The research is data-based and before diving into the nature of the data, this chapter explores the tools with which the data was collected. This is where the background material about network telescopes and the progress made since their inception is introduced. The chapter explores why network telescopes are an integral part of information security and the implications that the growth of the Internet of Things (IoT) has had on its usability. The chapter also explains the exhaustion of IPv4 addresses and the implication of this on the use of network telescopes. Internet Background Radiation (IBR) is the name given to the data collected by network telescopes.

Chapter 3 discusses the second part of the review of literature related to this research. An exploration of the statistical tools which have been used to process the data are explained here. In this chapter, the author explores tools like bootstrapping, which is a data simulation statistical technique needed when one would want to evaluate the confidence intervals given a scenario where a data user does not have complete datasets. Thus by using the data parameters, a user is able to reproduce a dataset that has similar attributes

as the original dataset. Since the network telescope data is of a time series nature, the chapter also explored time series analysis. Another tool discussed in this chapter is the art of mathematical modelling which forms one of the main building blocks of this research.

Chapter 4 focuses on the data that is used for this research. In this chapter, the reader gets to know the sources of the data and where it was collected. The study used network telescope data collected from Rhodes university. The data is split into two categories: monthly and quarterly. For the monthly and quarterly analysis, data from three network telescopes is used and its characteristics are also discussed in this chapter. The chapter also explores the sampling techniques that were used to process the data into samples. Since the study evaluated the effect of the size of network telescopes, sampling forms a critical part of this study.

Chapter 5 is where the actual result-oriented data analysis begins. Two different bootstrapping techniques are explored in this chapter and a comparison of which bootstrapping technique is ideal for IBR data is discussed here. The research approach used to bootstrap IBR is also discussed in this chapter. The results, discussions and recommendations of bootstrapping also compose this chapter.

Chapter 6 introduces the models that have been developed in response to the research question. In this chapter, four models built on the models that were developed by Hyndman and Koehler (2006) are presented. The chapter justifies the need for the models and how they work. The underlying assumptions with which the models should operate are also explained. The tests conducted on the models are evaluated using the monthly and quarterly data sets. The data are split further into sequential and random sampling and the results and discussion are computed and explained. The limitations of the model are also explained in this chapter, which further pays attention to the feasibility of sampling the datasets whose answer helps to address the primary research question.

Chapters in **Chapter 7** presents consolidated findings and recommendations from the research study. In this chapter, practical applications of the developed models are presented. The chapter makes it clear that the fields presented in it are just a sample of the many fields in which the models can work. Practical applications of bootstrapping IBR data are also presented in this section.

The document concludes the research with a discussion of the results achieved during this study in **Chapter8.** The focus of this chapter is tying the results achieved to the goals set at the start of this research. The chapter concludes with a discussion of future development and research that could come from this study.

2

Literature Review: Network Telescopes

This chapter introduces the reader to, and facilitates a better understanding of the parameters with which this research was conducted. The chapter begins by introducing network telescopes and how they collect IBR data in **Section 2.1**. This is followed by the discussion on IBR Data in **Section 2.2**. Considering that the research was inspired by the desire for people to be able to use small-sized network sensors to collect data, the study looked at related work regarding the feasibility of using small-sized network telescopes and how others have done it in **Section 2.3**. There was a need to explain how useful the IBR data has been as a source of threat intelligence data, thus **Section 2.4** explains some of the use cases of IBR data and how the field of Network security has drawn insights from it. **Section 2.5** expands more on the exhaustion of IPv4 addresses and the extent to which the depletion of the IPv4 address blocks reached. This is a concept that was introduced in **Chapter 1**. **Section 2.6** starts by looking at time series analysis because IBR data fits in this category. The study considers the significance of time series analysis and issues that come with using it. The chapter closes with a summary in **Section 2.7**

2.1 Network Telescopes

Traditionally, the operation of network telescopes has required larger contiguous block of IP addresses (Moore, 2002; Moore *et al.*, 2004; Bailey *et al.*, 2005; Fachkha and Debbabi, 2016). These blocks can be as large as /8 or /16 net-blocks (Benson *et al.*, 2015; Richter and Berger, 2019). The IP address blocks ought to be contiguous because such a network setup helps to understand which range of IP addresses are being targeted the most (Moore *et al.*, 2004). Running a contiguous block of IP addresses makes it easier to configure than when the IP addresses are spread around (Moore *et al.*, 2004; Irwin, 2011; Fachkha and Debbabi, 2016). This network telescope setup is proving hard to emulate currently because organisations are increasingly under pressure to utilise fully the address blocks they have (Durand *et al.*, 2011; Richter *et al.*, 2015).

Network telescopes can be configured to either actively monitor the threat activities in the unallocated blocks or set up for passive monitoring (Bailey *et al.*, 2006; Hunter, 2018). With network telescopes configured for active monitoring, the telescopes are designed to enable them to probe more information from the source host to ensure that a TCP three-way handshake occurs (Bailey *et al.*, 2006; Hunter, 2018). This is just to establish a connection, after which traffic is one way. This configuration gives them an edge in threat intelligence gathering in that they are able to receive application-level data from the first packet that may lead to a better understanding of an exploit attempt (Cooke *et al.*, 2004; Bailey *et al.*, 2005, 2006). TCP uses a three-way handshake to establish a reliable connection. The connection is full-duplex, and both sides synchronize (SYN) and acknowledge (ACK) each other before any packet transmission is done (Nagai *et al.*, 2018; Dang *et al.*, 2018). The exchange of these flags to establish a connection is performed in three steps: SYN, SYN-ACK, and ACK as shown in **Figure 2.1**. However, in this research study, the primary focus was on unidirectional traffic only as it relates to IBR.

On the other hand, network telescopes configured for passive monitoring, like the one in this research study, are only capable of receiving incoming traffic, thus are unable to record TCP based exploit data or details of misconfigured application requests (Bailey *et al.*, 2006). However, due to the availability of UDP and ICMP packets in their dataset, passive telescopes are also capable of recording threats that do not need a three-way handshake (Harder *et al.*, 2006; Hunter, 2018; Bou-Harb *et al.*, 2018). DDoS attacks can also be detected in passive network telescopes (Harder *et al.*, 2006). The telescopes used in this study were configured to do passive monitoring, i.e. they do not accommodate the TCP three-way handshake. **Figure 2.2** shows a basic network telescope setup.

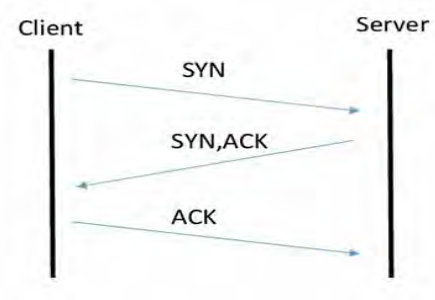


Figure 2.1: TCP Three-way Handshake

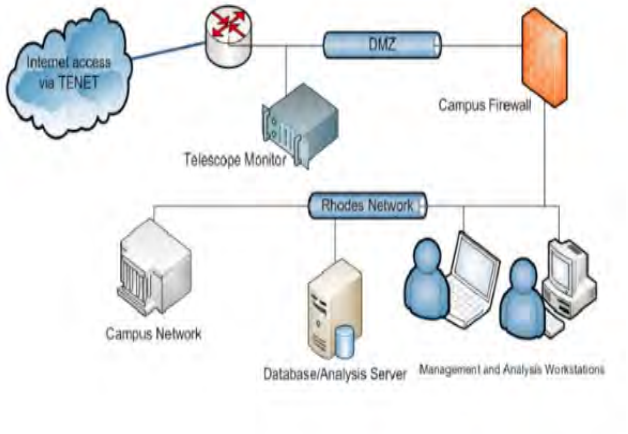


Figure 2.2: Basic Network Telescope Setup (Irwin, 2011)

The primary focus in **Figure 2.2** should be drawn to the fact that the network telescope sensors are located outside the firewall to ensure that incoming traffic is not filtered. What is common between active and passive monitoring is that, in each case, routable, yet unused Internet Protocol (IP) addresses are used (Bou-Harb *et al.*, 2018; Torabi *et al.*, 2020). Secondly, traffic destined to these inactive hosts has been observed to contain suspicious and unsolicited activities such as random scans of vulnerable systems on the other networks which are often used for Internet reconnaissance activities (Bou-Harb *et al.*, 2018; Torabi *et al.*, 2020).

The reason for the growing interest in the field of network telescopes is partially attributed to the changing threat landscape which brought about self-propagating malicious viruses (Irwin, 2011; Bertino and Islam, 2017; Bou-Harb *et al.*, 2018; Torabi *et al.*, 2020). The aforementioned were initially observed in IBR data when it was analysed, proving the ability of IBR data to forewarn threat activities in a network (Shannon and Moore, 2004; Irwin, 2012; Bou-Harb *et al.*, 2016; Bertino and Islam, 2017). The change in the threat

landscape has made IBR data even more valuable as it provides a unique way of looking at the existing datasets and provides an early warning mechanism for new threats and attacks (Harder *et al.*, 2006; Torabi *et al.*, 2020). The use of network telescopes, as a means of collecting IBR data and analysing network telescope traffic, has been adopted by security experts to understand the evolution of network threats and various potential malicious activity (Wustrow *et al.*, 2010; Irwin, 2011; Fachkha *et al.*, 2012; Fachkha and Debbabi, 2016; Bou-Harb *et al.*, 2016, 2018). This was done in order to justify why it is essential to have such a set-up for threat intelligence gathering. The reasons include its ability to provide global network security events that are difficult to monitor using the traditional node or end to end measurements (Moore *et al.*, 2004; Irwin, 2011; Torabi *et al.*, 2020).

Bou-Harb *et al.* (2018); Piotr *et al.* (2019); Torabi *et al.* (2020) also explained that network telescopes allow users to be forewarned of the threats that could be targeting their production networks Network telescopes expose malicious threats targeting an organisation and poor security configuration that could otherwise be missed. However, their major drawback is that they are costly to maintain, especially when large contiguous net-blocks are used to generate network telescopes and they can be polluted by misconfiguration traffic (CAIDA, 2017; Bou-Harb *et al.*, 2018). These costs, in terms of fees to Regional Internet Registries (RIR) and Internet Service Provider (ISP), hold true for both IPv4 and IPv6 since acquiring more IP addresses requires more funds and the management thereof.

As far as past research studies are concerned, it has been established that having large address blocks is vital for monitoring globally scoped events (Moore, 2002; Cooke *et al.*, 2004; Fachkha and Debbabi, 2016; Torabi *et al.*, 2020). Other researchers have also reported that distributed sensors provide more addresses that increase visibility, which in turn broadens coverage of threats (Yegneswaran *et al.*, 2004; Bailey *et al.*, 2005; Irwin and Nkhumeleni, 2015; Bou-Harb *et al.*, 2018). According to Chatziadam *et al.* (2014), a distributed network telescope can be defined as network telescope set up that consist of many network telescope sensors that reside on remote networks capturing traffic from the unallocated address blocks and relaying it back to a central server so that it can be classified and analyzed. **Figure 2.3** shows an example of a distributed network telescope set up.

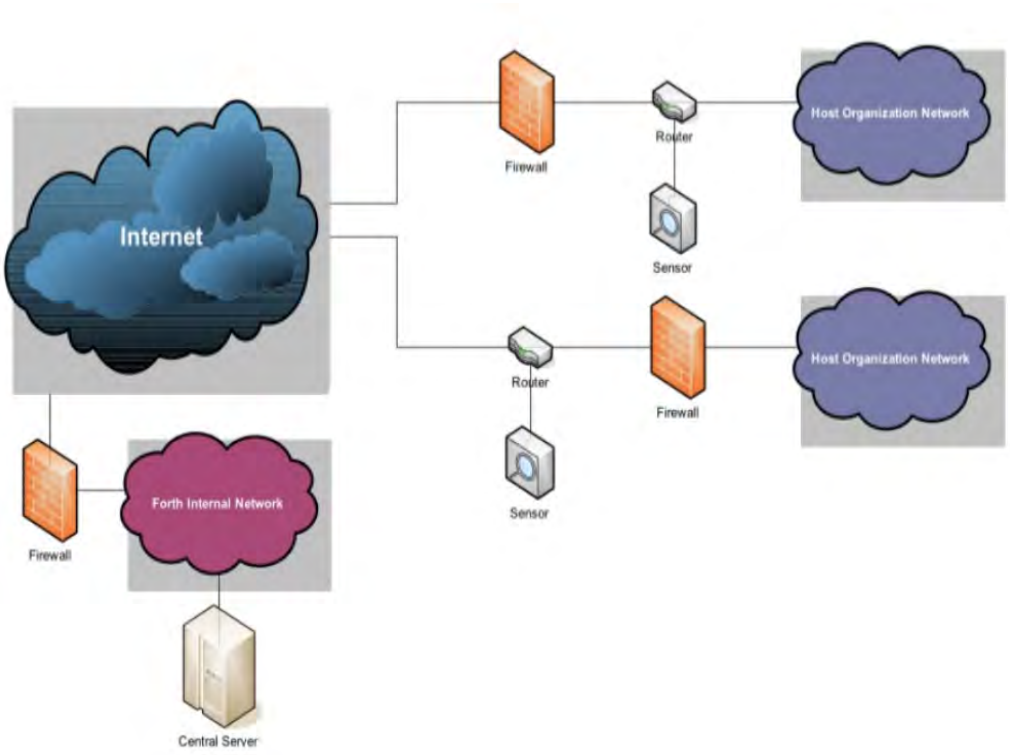


Figure 2.3: Distributed Network Telescope Setup (Chatziadam *et al.*, 2014)

A distributed network telescope set-up means that more DSTIP addresses have been used, typically on a diverse set of DSTIP ranges, leading to increased visibility. This, in turn, improves detection time, duration precision, and helps in revealing the events happening in the address blocks under observation (Chatziadam *et al.*, 2014; Fachkha and Debbabi, 2016). The set up shown in **Figure 2.3** shows one sensor outside the firewall while another sensor is inside the network telescope. This is dependent on the network telescope user preference and the objective to be met by such a set up. It should be noted that having network telescope sensors inside the firewall means that the traffic will be filtered according to the firewall configuration which could clear out some threats and thus less traffic collected.

Having more telescope sensors entails that the network telescope user sets network telescopes that can operate in different address blocks of the network and thus maximises the amount of data collected (Chatziadam *et al.*, 2014; Irwin and Nkhumeleni, 2015). However, the cost of running such operations, maintaining the infrastructure (operation and storage), and hosting many IP addresses on a network is quite exorbitant. For instance, in 2017 alone, the Centre for Applied Internet Data Analysis (CAIDA) spent over $3,500,000 maintaining their network telescope systems (CAIDA, 2017). In addition to this, there is the shortage of IPv4 addresses which has been explained in detail in **Section 2.5**

2.2 Internet Background Radiation Data

IBR data has proved to be useful in gathering threat intelligence data for over a decade through the gathering of unsolicited network traffic (Polakis *et al.*, 2011; Irwin, 2012, 2013; Fachkha and Debbabi, 2016). As explained in **Chapter 1**, IBR data consists of non-productive data packets on the Internet, which target unused IP addresses, or ports, where there is no network capable-device set up to receive them. A lack of hosts that are connected to any network in these unallocated blocks of IP should imply zero traffic. However, if traffic is registered in this region, then it can only be attributed to poor configuration, back-scatter from spoofed source addresses, or scanning from worms and other probings (Cooke *et al.*, 2004; Bailey *et al.*, 2005; Irwin, 2013). Worth mentioning is that the IBR data collected by network telescopes is usually unidirectional. This is to say that IBR normally collect incoming traffic and do not respond (Dainotti *et al.*, 2014). The word radiation in IBR is used because IBR data contain persistent traffic that originates from many sources distributed all over the world (Wustrow *et al.*, 2010; Fachkha and Debbabi, 2016). This known attribute is critical in that it enables IBR data to provide a valuable source for Internet situational awareness from a global perspective.

With the coming in of Internet of Things (IoT), IBR data is even more critical as recent research showed that there are newly targeted ports observed in IBR that indicate emerging IoT malware/botnet (Fachkha *et al.*, 2017; Torabi *et al.*, 2020). Among these newly target ports include destination port range 19328–19622 which according to Torabi *et al.* (2020) had no known vulnerabilities at the time of their study. Other ports included in the study they conducted include ports running Web Services on Devices API (WSDAPI) which runs on port 5358 (for both TCP and UDP), Server Message Block (SMB) which runs on 445/TCP, Remote Desktop Protocol which runs on port 3389 for both TCP and UDP, and CPE WAN Management Protocol (also known as CWMP or TR-069)which runs on 7547/TCP.

The increasing number of cyber attacks that target IoT devices because they are used for data collection, monitoring, and information sharing, illustrate the rise of malware tailored towards IoT devices (Bou-Harb *et al.*, 2016; Torabi *et al.*, 2020). The primary objective of such tailor-made malware is to exploit vulnerable IoT devices (Bou-Harb *et al.*, 2016). In such cases, IBR data has been used to detect and characterize emerging IoT malware/botnets because of its ability to provide an Internet-scale perspective of IoT devices and their unsolicited activities over a period of time (Fachkha and Debbabi, 2016; Bou-Harb *et al.*, 2016; Torabi *et al.*, 2020).

Most well-known computer worms, common threats and scanning activities have been investigated through IBR data (Benson *et al.*, 2013). Among these include Distributed Reflection Denial of Service (DRDoS) (Bou-Harb *et al.*, 2014), Witty worm (Shannon and Moore, 2004) Code Red and Slammer/Sapphire (Fachkha and Debbabi, 2016), Sality SIP scan botnet (Bou-Harb *et al.*, 2016) and Conficker worm (Irwin, 2012). These are among the most analysed threats which have been analysed using IBR data. In cases where there has been a lack of empirical data related to the widespread deployment of IoT devices IBR data has proved to be very useful in providing insightful feedback needed for decision making (Simmon *et al.*, 2013; Bou-Harb *et al.*, 2016). Even more importantly, IBR has provided details showing early stages of malware contamination (infection) on machines, thus offering an early warning detection mechanism to allow organisations to act in time and curb the detected infection as early as possible (Irwin, 2011; Chatziadam *et al.*, 2014; Fachkha and Debbabi, 2016; Bou-Harb *et al.*, 2016).

IBR has also been used to understand the security of Cyber-Physical Systems (CPS) protocols where an observation was made on the lack of interest in UDP-based CPS services, and the prevalence of probes towards the Inter-Control Center Communications Protocol (ICCP) and Modbus protocols (Fachkha *et al.*, 2017). Noteworthy is that CPS is heavily used in different industries, which include but are not limited to aerospace, automotive, energy, healthcare and manufacturing (Simmon *et al.*, 2013; Fachkha and Debbabi, 2016; Fachkha *et al.*, 2017). IBR related projects are found to monitor various cyber threat activities and are distributed in one-third of the global Internet (Fachkha and Debbabi, 2016). All of this demonstrates how significant IBR data has been as a source of valuable Internet-wide cyber threat intelligence. **Section 2.4** offers more insight on some of the use cases IBR data as a source of threat intelligence data.

Despite IBR data being of great value, not all organisations can afford to set it up and operate its infrastructure. Among the most significant reasons are the shortage of IP addresses in the IPv4 net-block and the cost of securing a large block of IP addresses for threat intelligence gathering (Cooke *et al.*, 2004; Durand *et al.*, 2011; Irwin, 2011; Fachkha and Debbabi, 2016). Organisations end up opting not to use the technique and allocate the IP addresses for production. This is where this study found a research gap and aims to address this shortage by offering an alternative. In this study, IBR data samples obtained from a range of /24 IPv4 address blocks were used to conduct tests and experiments in order to achieve the project's research objectives. Details of the data used can be found in **Chapter 4**

2.3 Related Work on Small Network Telescopes

As explained in **Section 2.1**, the cost of maintaining large blocks of IP addresses for the collection of IBR data remains a significant point of consideration. Though the exhaustion IPv4 address space affects the allocation of large blocks in this space, it does not affect IPv6. However, the cost of data collection with such network telescopes affects both IP addresses. Large datasets also implies the need for large storage resource and more time to make sense of the data. Having small sized network telescopes solves these problems. Thus in this work, the primary investigator used subnets of a /24 network telescope. This calls for a need to review literature to see how practical this is.

Past research (Harder *et al.*, 2006) has also shown that it is possible to configure a small-sized network telescope and collect threat intelligence data just as one would with a larger telescope . Harder *et al.* (2006) defined small network telescope as that which can accommodate a /24 net-block while Benson *et al.* (2015) varied between /16 net-block to a /8 net-block as the definition of a small sized network telescope. In a study conducted by Harder *et al.* (2006), it was established that virtually all traffic in a /24 IPv4 net-block, when monitored by a network telescope, was found in the top 100 destination IP addresses. This cements the notion that if properly sampled, the bigger net-block can virtually be represented by a smaller subset of the IP addresses of a bigger telescope. Another study conducted by Zeghache and Yacine (2020) showed that although large network telescopes are preferred when collecting threat intelligence, small network telescopes can also be configured to collect such kind of data. This was done in acknowledgement of the fact that large network telescopes are still better at collecting more threat intelligence data than small ones. However, they did not quantify the differences that exists in the threat intelligence collected between the large and small sized network telescope. Zeghache and Yacine (2020) used a /27 network telescope which accommodates 32 IP addresses.

Benson *et al.* (2015) conducted a series of experiments to assess how threat intelligence data collected by the network telescope is affected by the network telescope size. The theory presumed that by using smaller network telescopes, they should be able to observe fewer unique SRCIP addresses and even those observed should be less frequent as compared to a larger network telescope. This they did by varying the size of the network telescope lens between /16 to /8. Their data contained traffic that had Conficker's PRNG. In each of the sizes, they were able to find traits of some of the unique SRCIP addresses observed in the baseline dataset, but the volume of these unique SRCIP addresses changed. This in turn affected the overall volume of the traffic contained in each network telescope lens, which directly affects the volume of threat intelligence data.

Distributed network telescopes have also been used with small-sized network telescopes for observing different segments of the network with their output combined into one (Moore *et al.*, 2004; Irwin, 2013; Irwin and Nkhumeleni, 2015). This offers a collated view of the events across the observed network address blocks because it ensures that the small-sized telescopes cover different parts of the network. Another element that has allowed the configurations of small-sized network telescopes is that most of the dominant SRCIP addresses (SRCIP addresses that send the most traffic) send traffic to all DSTIP addresses in the network telescope (Richter and Berger, 2019). This ensures that, even if a small-sized network telescope is used, there is still a high chance of getting a significant amount of threat intelligence from the network traffic that they collect. Thus, by collecting data for longer periods, the discrepancies that may exist between large network telescopes and small network telescopes can be significantly reduced.

Even more significant are the scenarios where network telescopes have shown evidence of widespread localised scanning. This is to say that localised scanning are scenarios where threat actors target a specific individual network as explained by Richter and Berger (2019). In cases like these, the difference between large network telescopes and small-sized ones is significantly reduced since the threat actor is aiming at maximising the number of scans within the targeted network. In this way, it may not come as a surprise where small network telescopes end up containing traffic that is similar in terms of the number of unique SRCIP addresses and the kind of threat intelligence contained in it. The scenario is different when Internet-wide scans are used where threat actors are more interested in the overall state of vulnerable networks on the Internet. In such cases, large networks may contain more SRCIP addresses because they have a wider coverage than smaller network telescopes.

Moore *et al.* (2004) have shown that a large-sized network telescope leads to larger datasets, and thus, more unsolicited events that result in more detail of network events are observed. The study also proved that the size of a network telescope has a high impact on the observation of a specific event within a given time frame. This is to say that using smaller network telescopes presents a chance of missing out on other network events irrespective of where the small-sized network telescope is placed on the network. This was achieved by using probability statistics of observing an event on a single host in a given net-block (Moore *et al.*, 2004). What was established from the study conducted by Moore *et al.* (2004) was that the smaller network telescopes take a relatively longer time to observe specific events as compared to a larger telescope given the same likelihood of occurrence.

A separate study conducted by Irwin (2013) revealed that the observations found in smaller network telescopes when analysing the Conficker worm were similar to those observed in much larger network telescopes. In work done by Zhan *et al.* (2014), the authors acknowledged that while it is possible to configure small-sized network telescopes, substantially smaller network telescopes might not be as useful as a large telescope. This is a study that used network telescopes of sizes /8, /16 and /24 net-blocks. Thus the smaller the network telescope the higher the likelihood of missing other aspects of the network traffic. This study, however, did not quantify the volume of threat intelligence data that can potentially be missed when working with smaller network telescopes. Wustrow *et al.* (2010) also conducted a study in which it was observed that a small portion of destination or source address space were used i.e. most of this traffic is directed towards a small number of destinations in one-eighth of the overall DSTIP addresses.

To the best of the researcher's knowledge and based on literature reviewed, a comparison to quantify the gap in size as to how much a larger network telescope can be represented by a small-sized network telescope has not been published yet by other authors. This is what has been laid out in **Section 1.3** as one of the research objectives. What is known thus far is that a larger network telescope collects more threat intelligence data than a small-sized network telescope (Moore *et al.*, 2004; Yegneswaran *et al.*, 2004; Harder *et al.*, 2006; Pemberton *et al.*, 2007). Pemberton *et al.* (2007) observed that monitoring global traffic routed to a single host in a network leads to the same outcome (larger telescopes collecting more threat intelligence data than smaller ones).

It is on this notion presented by other researchers that this study opted to build by observing the sampled IP addresses and observing how long it would take to observe a specific proportion of the baseline dataset.

2.4 Network Telescope and IBR Data Use Cases

During this research study, work was done to see the viability and some good use cases of IBR data and network telescopes in general. This was a build-up of what other researchers have done relating to good uses of network telescopes and why they are significant to threat intelligence gathering. One of the experiments conducted by this research's primary investigator was evaluating the re-emergence of the SQL Slammer worm which did reappear in November and December 2016 (Chindipha and Irwin, 2017). This re-emergence was first reported in December 2016 by Check Point[1] researchers who confirmed that SQL

[1]https://research.checkpoint.com/2017/aprils-wanted-malware/

Slammer worm was back online targeting the same ancient flaw in Microsoft SQL server 2000 buffer overflow vulnerability. Using IBR data, the work done here, Chindipha and Irwin (2017) confirmed that Rhodes University's network telescopes had picked up spiked attributed to SQL slammer worm i.e. the IBR data detected SQL Slammer worm. Using UDP traffic, the signature of SQL Slammer worm was identified until March 2017. Considering that network telescopes act as early warning detection systems, it is possible that the re-emergence of SQL slammer worm could have been expected and measures put in place to mitigate it before any damage could be done. It is important to note that the damage caused by SQL slammer when it re-emerged was not as bad as it did in early 2000.

In another study conducted by Irwin (2012), it was observed that the pattern and behaviour seen by the effects of the Conficker worm were also reflected in the network telescope. Rhodes University network telescopes showed unusual volume of traffic that was not observed prior to the emergence of the Conficker worm. All this traffic was targeting destination port 445/TCP. The telescopes picked spikes of network traffic for a period of 14 months between August 2005 to September 2009. This was the same period when Conficker was in its prime. This is the same malware that exploited a vulnerability in the Microsoft RPC; a vulnerability that was also exploited by the Blaster and Welchia (Nachi) worms (Irwin, 2011).

Shannon and Moore (2004) used IBR data to measure the rate at which the Witty worm spread, which offered a global view of the spread of many Internet worms. In the process of doing this, Shannon and Moore (2004) showed how many computers were affected every two minutes, which later escalated to the pattern of spread every hour. Using such knowledge, measures were put in place to combat the spread. Such insight from the network telescope proved pivotal in understanding the worm's behaviour. Another study conducted by Harder *et al.* (2006) showed that IBR data gave a good understanding of Internet worm and virus Attacks on an active network. In this study (which has partly being explained in **Section 2.3**), results showed that traffic is not as random as it is thought to be as top 10 destination ports and top 100 DSTIP addresses account for virtually all the traffic collected. In addition to this, Fluid models which normally work with systems containing destination hosts that are prone to malicious software attacks have been improved, trained, and evaluated to understand worm behaviour using IBR data (Zou *et al.*, 2003). Code Red worm was one of the malware whose understanding has improved since network telescopes were acknowledged as a good source of threat intelligence (Zou *et al.*, 2003). Organisations and security companies such as the Computer Emergency Response Team (CERT), CAIDA, and SANS Institute have their own network

telescopes that they use to monitor the Internet and analyse unusual network traffic patterns (Zou *et al.*, 2005). Advice is then disseminated to organisations once their security experts identify the cause of such moments and appropriate solutions are offered.

IBR data has also been used successfully to characterise cybersecurity posture in a data-driven study. Cybersecurity posture can be defined as the strength of the cybersecurity controls and protocols for predicting and preventing cyber threats, and the ability to act and respond during and after an attack (Bahuguna *et al.*, 2020). The intention of a cybersecurity posture is to provide a high level indicator to general risk categories (Xu, 2019). Usually, such an evaluation is done using the status of an enterprise's networks, information, and systems based on information security resources (e.g., people, hardware, software, policies) and capabilities in place to manage the defence of the enterprise and to react as the situation changes. However, in this data-driven characterisation cybersecurity posture study, Zhan *et al.* (2014) conducted experiments and proved that the sweep-time is supposed to be characterised as a stochastic process rather than a random variable. Sweep-time was defined as the time taken for the majority of the network telescope DSTIP addresses to be probed at least once throughout the observation period. In the same study, it was observed that the total SRCIP addresses (which represent the number of attackers and attacks) observed by the network telescope is largely determined by the number of attackers from a single geolocation (city or country). In the same study, propositions were made to formalise the concept of cybersecurity posture from the number of victims (DSTIP addresses that are attacked), the number of attackers that are observed by the telescope, and the number of attacks that are observed by the telescope using a time series analysis.

Due to the nature of how network telescopes operate, i.e. acting as an early warning system, data collected from these network telescopes have been used to identify zero-day exploits and mitigate Advanced Persistent threats (Maglaras *et al.*, 2018). In a separate study conducted by Blaise *et al.* (2020), they used machine learning to detect zero-day attacks using port-based approach. Their study used University of California San Diego Network Telescope dataset largely composed of botnet scans. In this study, they did observe certain indicators (infection of hosts or device fingerprinting) in their datasets that could have easily been detected to prevent the Mirai botnet attack. Using IBR data, an anomaly detection technique was developed that allowed the research investigators to identify main changes in the usage of specific ports as desired to identify botnets. In addition to this, IBR data has been used to assess the level of security threat that network scans can have on application software (Fachkha and Debbabi, 2016; Houmz *et al.*, 2021). These vulnerabilities in the software are identified using network scans when an external

host (SRCIP address in the case of network telescopes) probes the networks. Network telescopes are used as a threat intelligence to capture large-scale scans and how worms spread on the network. Considering that port scanning is one of the techniques used when attackers are gathering information about their target, it makes network telescopes a good source of such data (Fachkha and Debbabi, 2016). These scans come with vulnerabilities that the scanned network has. Such vulnerability disclosure impacts different aspects of the information security domain as systems' vulnerabilities become public knowledge, which can easily be exploited if not patched (Richter and Berger, 2019). Among some of the information revealed through such disclosures include: when last a software was patched and how quickly it can be patched, what exploit should be used, and the volume of attacks and scans. Thus by using network telescopes, a network telescope user is allowed to trace tracking scanning activity of their software from threat actors. The network telescope user can then patch their software accordingly using IBR data. This becomes more critical when specific software is a victim of a localised scan which poses a potentially greater threat as compared to an Internet-wide scan.

What is common in each of the malware analysis use cases conducted by the network telescopes datasets is that IBR data can be used to identify worms and track how these worms spread across the Internet. This could prove difficult to analyse during an actual attack in a live network. With IBR data, the malware behaviour and patterns can easily be characterised and identified using IBR data. Using location of SRCIPs in IBR data during data analysis can help security experts know the geolocation of most of the malware. All these offer a good base for threat intelligence which, due to the nature of network telescopes, could not be clearly understood at a cheaper cost but also be prevented in time or aid to avoid future attacks. Based on these few cases, one can see how IBR data is such a valuable asset of threat intelligence data and its applications and usage will still be viable, particularly in an age where everything is going digital.

2.5 IPv4 Address Exhaustion

IPv4 address exhaustion is the depletion of the pool of unallocated IPv4 addresses (Zander *et al.*, 2013; Dainotti *et al.*, 2016; Beeharry and Nowbutsing, 2016) i.e. limited supply of Internet Protocol version 4 (IPv4) addresses. The original architecture IPv4 provides 2^{32} (4,294,967,296) IP addresses (Cotton, 2001; O'Neill *et al.*, 2001), but the emergence of Internet of Things (IoT), among other reasons, has led to a dramatic growth rate that was not initially anticipated when Internet was invented (Bush, 2011; Durand *et al.*, 2011;

Zander *et al.*, 2013; Dainotti *et al.*, 2016). However, in early 2000, with the introduction of new devices (like smartphones for example) that also needed IP addresses to connect to the Internet, other researchers were able to forecast the possibility of such a depletion (Arkko and Townsley, 2011; Zander *et al.*, 2013; Beeharry and Nowbutsing, 2016; Dainotti *et al.*, 2016). This, coupled with the coming in of novel technological advances such as 5G, Internet of Things (IoT) and smart cities, has led to the rapid decline of IPv4 address worldwide (Mamushiane *et al.*, 2021). As of 2015[2], it was recorded that four Regional Internet Registries (RIR) were unable to allocate new IPv4 addresses to users. Internet Assigned Numbers Authority (IANA)[3], a standards organization that oversees global IP address allocation and autonomous system number allocation, among other things (Cotton and Vegoda, 2010; Durand *et al.*, 2011), made plans that allowed the Internet to continue its amazing growth and promote global innovation. As of October 2016, IANA changed its name to Public Technical Identifiers (PTI)[4]

Although this was a good development, such a change in protocol usage has not been adopted fully by all Internet users (Bush, 2011; Durand *et al.*, 2011; Zander *et al.*, 2013; Dainotti *et al.*, 2016). As explained in **Section 1.2**, unlike other parts of the world like America, Asia, and Europe, Africa is one of those continents where the Internet Service Providers (ISP) do not offer widespread support for IPv6, particularly for commercial use (Perkins, 2010; Beeharry and Nowbutsing, 2016; Mamushiane *et al.*, 2021). Commercial users are the major contributors to innovation and with the scarcity of IPv4 addresses, they can be negatively affected (Bush, 2011). Despite the campaigns that have happened in Africa to raise the urgency and need to migrate to IPv4, not many countries have made the successful transition (Nyirenda-Jere and Biru, 2015; Livadariu *et al.*, 2017; Mamushiane *et al.*, 2021).

The depletion of the availability of large IPv4 network blocks is of great concern, particularly in the cybersecurity field that often relies on acquiring large network blocks for its threat intelligence gathering (Pang *et al.*, 2004; Bailey *et al.*, 2005). Large net-blocks are significant because they give a broad spectrum from which to observe threats and thus are better placed to make a more informed decision than what one would get if a smaller network telescope or data were used (Atifi and Bou-Harb, 2017; Piotr *et al.*, 2019).

There have been at least 200 training sessions in 45 countries in Africa to raise awareness of the urgency needed to migrate to IPv6, however, it has been difficult to track the

[2]https://www.nro.net/about/rirs/
[3]https://www.iana.org/
[4]https://pti.icann.org/

progress of what these sessions have achieved (Livadariu *et al.*, 2017). In addition to this, the rate at which the African population is growing coupled with the growth rate of Internet users in Africa makes this problem more severe than America, Asia, and Europe, where organisations have started using IPv6 and are getting support (Perkins *et al.*, 2004; Nyirenda-Jere and Biru, 2015; Livadariu *et al.*, 2017). Africa does not have enough IPv4 address blocks to support this massive growth in Internet usage (Livadariu *et al.*, 2017). There has been much resistance in the adoption of IPv6 in Africa, with only 20% of African autonomous systems publicly advertising IPv6 prefixes (Livadariu *et al.*, 2017; Mamushiane *et al.*, 2021). Apart from AFRINIC, all Regional Internet Registries (RIRs) have allocated their last /8 address blocks of IPv4 (Durand *et al.*, 2011; Livadariu *et al.*, 2017; Hamarsheh *et al.*, 2021). Though efforts have been made to migrate to IPv6, deployment to IPv6 has been very slow partly due to a lack of commercial support from ISPs and end users are not directly benefiting from the transition (Livadariu *et al.*, 2017; Hamarsheh *et al.*, 2021; Mamushiane *et al.*, 2021).

The problem gets worse with the increase in the use of mobile devices, adoption of mobile banking, coupled with the growth rate of Internet users in Africa (Nyirenda-Jere and Biru, 2015; Mamushiane *et al.*, 2021). These factors make this problem more severe on the continent compared to other parts of the world (Perkins *et al.*, 2004; Nyirenda-Jere and Biru, 2015; Livadariu *et al.*, 2017). This also means that the problems that were anticipated in early 2000 have a high probability to impact Africa and any other continent that has not yet migrated to IPv6. It is important to note that this IPv4 address exhaustion is a global problem[5] and not an African problem only as this can be seen in Asia and Europe as well (Perkins, 2010; Zander *et al.*, 2013; Beeharry and Nowbutsing, 2016; Lencse and Kadobayashi, 2019).

For instance, there are problems like a network operator being unable to receive large IPv4 blocks from RIRs that are sufficient to address any significant Internet infrastructure for its customers[6] (Cotton and Vegoda, 2010; Bush, 2011). When things like this happen, organisations have had to adopt new policies that allow IPv4 address blocks to be transferred between consenting parties, under specific conditions, just to remain in operation (Bush, 2011). Alternatively, ISPs will have to allocate temporary lease of blocks of IP addresses to their customers if they have them available or allow the co-existing of IPv4 and IPv6 in their network as they slowly transition (Lencse and Kadobayashi, 2019). When the lease expires, the ISP will need to re-assign the block of IPs to a different customer (organisation). For continual usage, organisations must motivate why they need more IP

[5]https://www.nro.net/ipv4-free-pool-depleted
[6]https://afrinic.net/exhaustion

addresses and how they plan to use them. It won't be long before Regional Internet Registry (RIR) for Africa (AFRINIC[7]) and Internet Service Providers (ISPs) start denying requests for IPv4 address blocks as it did with other RIRs (Bush, 2011; Zander *et al.*, 2013; Mamushiane *et al.*, 2021). This leaves them only on open market and only acquired at exorbitant costs.

This is not to say that AFRINIC is not doing anything regarding the promotion of migrating to IPv6 or how to manage the currently available IP addresses, as evidence[8] shows otherwise. AFRINIC has embarked on a journey to ensure a phased approach[9] in transitioning from IPv4 to IPv6, but this project needs significant investment from all parties involved (Arkko and Townsley, 2011). There is progress, but it is difficult to track the efforts that have been put in place through training and implementation of supporting systems (Livadariu *et al.*, 2017; Mamushiane *et al.*, 2021). Such exhaustion of IP addresses threatens the usability of network telescopes as no organisation will be willing to reserve IP addresses for passive monitoring (Chindipha *et al.*, 2019b). Neither ISPs nor AFRINIC can approve of such a proposal to lease the depleted IPv4 for what is seen as 'non-productive' use such as threat intelligence gathering (Zander *et al.*, 2013). It is from this context of knowing the extent to which IPv4 exhaustion has occurred that this research was conducted. Worth noting is that the practical applications of this study go beyond IPv4 address exhaustion. Its application fits with the use of IPv4, but it is the issue of exhausting that necessitated this study.

2.6 Time Series Analysis

Time series analysis has been used in different sectors of the industry, be it in economics when dealing with stock market prices, geophysics when dealing with earth tremors, and meteorology when predicting weather conditions. These time series can be categorised into three main categories:

1. Stationary and non-stationary time series (Huang *et al.*, 1998)
2. Long-term memory and short-term memory time series (Granger and Joyeux, 1980)
3. Equidistant and non-equidistant time series (Rüping, 2001)

[7]https://afrinic.net/
[8]https://afrinic.net/policy/manual
[9]https://afrinic.net/20200113-afrinic-enters-ipv4-exhaustion-phase-2

Stationary time series are those that have constant statistical values like mean, standard deviation and variance, while non-stationary time series are those that have such statistical values (moving average, mean, variance, std) fluctuating over time (Cogley and Nason, 1995). Long-term memory time series are those whose rate of dependency between newly observed values and their predecessors have their auto-correlation function declining at a very slow rate (Granger and Joyeux, 1980). On the other hand, short-term time-series are those whose auto-correlation function decreases rapidly between newly observed values and their predecessors (Granger and Joyeux, 1980).

Lastly, equidistant time series are those whose data values are recorded at a constant period (Rüping, 2001). On the other hand, non-equidistant time series have no fixed time frame as the time of data collection varies from time to time (Rüping, 2001). For instance, examples of equidistant time series include total sales every month and quarterly observations of volumes of US E-Commerce (Milhoj, 2013).

IBR data fits into the *time-series* data category because of the time associated with it. IBR data for this study was collected daily for years, which appends time to it. Due to this daily collection, it would be classified as an *equidistant time series* when aggregated by frequency (daily, weekly, month etc.) and it is *non-stationary time series* because its statistical values change with each collection when binned by time. Due to its random nature, it cannot be classified as long-term memory since a day's traffic could be independent of its preceding date.

Time-series analysis is performed to understand the components and behaviours of the time-series coupled with the trend behaviour *T(t)*, seasonality factor *S(t)*, cyclic nature *C(t)* of the time series under study, or the randomness of the series R(t) (Ostashchuk, 2017). The researcher identifies factors that are responsible for any of the behaviours listed in the preceding sentence in the process of doing the analysis (Varouchakis and Hristopulos, 2013). **Equations 2.1** and **2.2**, *Z(t)* denote the time series at a given time t (Ostashchuk, 2017). In addition to this, time series analysis allows us to have some ability to predict the future behaviour of the time series under study (Ostashchuk, 2017). This study will primarily focus on the first reason as predicting the future behaviour of the time series is not part of its scope.

In order to get accurate results, the analyst needs to identify the right model for the time series data under study. This is primarily dependent on the nature of the growth rate. Depending on how quickly the growth rate of a time series changes over time, Wei (2006) found that a time series could be classified as an additive model or multiplicative model.

Ostashchuk (2017) explained that a multiplicative model is applicable to those models that have a fast growth rate while the additive model is used for those that have a slow growth rate.

$$Z(t) = T(t) + C(t) + S(t) + R(t) \cdots Additive - Model \tag{2.1}$$

$$Z(t) = T(t)C(t)S(t)R(t) \cdots Multiplicative - Model \tag{2.2}$$

For example, consider airlines ticket sales over the year. It is a well-known fact that prices for flights peak during the festive season (December - January). If each year the prices by the airlines are increased by a specific amount, let us say R200, then this becomes an additive model because our seasonality factor is a constant value. However, if the airlines want to take advantage of Rand fluctuations over the year (be it gain in value or loss) then instead of a fixed amount, the prices may increase by 17%. This makes our model multiplicative and in each of the cases, the seasonality factor is accounted for. In both additive and multiplicative models, the seasonality factor (*S(t)*) which signifies the repetitive behaviour of time data over an identified period is identified (Kalekar, 2004). The equation for these models is shown in equations 2.1 and 2.2 . The data for this study falls in the multiplicative model because the fluctuations of the IBR data do not have a fixed growth value. As we move from the first day of the month to the last, there are random changes that could be identical for certain months, but the value of changes is not the same. Its random nature makes it ideal for multiplicative models than additive ones.

In time series, the correlation can exist between adjacent values or seasonally (Matalas, 1967). This is to say that entities from two adjacent seasons are performing similarly to the point that a pattern can be formed (Matalas, 1967). Thus, the correlation coefficient can be computed using days within a week (adjacent values). Alternatively, if the seasonality is a week, then one can compute correlation weekly. This line of thought can work with monthly computations or quarterly.

Time series is critical to this study because a majority of the analysis that is conducted is over time series data. This is both in terms of graphical analysis, regression analysis, and mathematical computation where the mathematical models have a time bound to them.

2.6.1 Correlation of Time series

Correlation is a process of establishing a relationship that exists between two or more variables with the aim of quantifying how strongly the variables are connected (Zebende, 2011). In other words, correlation measures the degree of association between variables of interest. Initially, the process was introduced when working with signals, but over time, time series analysis has incorporated correlation techniques (Kohn, 2006). This works for both stationary and non - stationary time series (Horvatic *et al.*, 2011).

There are several forms of correlation but those of particular interest to this study are autocorrelation and cross-correlation (Zebende, 2011). This is the case because, essentially, subnet equivalents are samples of the baseline dataset which would indirectly make it autocorrelation to see the similarity between the baseline dataset and its subnet equivalent. However, since certain data points are missing in the subnet equivalents, the cross-correlation helps us to understand how one variable in the dataset affects the other to bring about the differences observed (Kohn, 2006). Cross-correlation in this case is defined as the process of establishing a relationship of two different time series to detect if there is a connection between metrics (Bourke, 1996). On the other hand, autocorrelation compares a time series with itself at a different time with the purpose of detecting repeated patterns or seasonality (Kohn, 2006). Each of these correlation techniques measures the degree of association, either with itself or another time series (Bourke, 1996).

2.6.2 Applications of Time series Data

The usefulness of the data collected is largely dependent on when the data collected will be applied or the time frame with which the data was collected. Decisions made based on data have a time factor added to them. In other words, time is a very crucial factor when collecting data for it informs the data users of when particular events occurred. In this subsection, the study looked at some uses cases of time series analysis. In forecasting time series data, the objective is to predict how the data observation will continue or vary into the future based on past and current events (Hewage *et al.*, 2020).

One of the most common use cases of time series analysis is in the prediction of weather patterns (Karevan and Suykens, 2020). The study of changes in the weather is necessary to get numerous advantages such as daily decision making on what clothes to wear, saving lives of people who live close to the coast, or mitigating risk when a hurricane has been detected, for example. Farmers need such information in order to know if it is

appropriate for them to start planting or wait (Hewage *et al.*, 2020). All this is made possible when meteorologists accumulate weather-related data to compute the state of atmospheric conditions. This data is then analysed using data-driven forecasting models for an application of weather forecasting.

Time series analysis also helps organisations when it comes to decision making based on sales trends. For instance, based on certain fluctuations in trends or essential patterns over time, by using various data visualization techniques, organizations could study seasonal trends and research more to understand the causes of these trends (Bai and Ng, 2008; Kumar *et al.*, 2020). This way, the organisations would know when to conduct certain sales and when not to do so. Clothing companies benefit greatly from such trend patterns' analysis. In so do doing, they cut on unnecessary productions by producing clothes that will not be bought.

Thus, time series analysis helps businesses make informed business decisions, as organisations analyse past data patterns in order to forecast future possibilities and assess how effective proposed changes have performed (Kumar *et al.*, 2020). With real-time data at hand, the organisation's proposed changes can be assessed in real-time to see if they are bringing intended results. Such data can also be used to assess the growth of an organisation over time. In the health sector, epidemiologists have used time series data to understand the spread and behaviour patterns of diseases (Held *et al.*, 2017). Covid-19 is a very good example of how epidemiologists were able to predict peak points and when the next wave of attack would come (Friedman *et al.*, 2021). Using such data, health experts were able to know the spread pattern and the efficiency of the treatments being offered to patients.

Times series analysis has also been used to analyse real-time traffic targeting websites (Shelatkar *et al.*, 2020), but also being able to understand unusual traffic patterns, which, more often than not, are indicative of threat activity within a network. It has also been used to help understand customer and employees behaviour as the amount traffic routed towards websites show which services are accesses at specific times (Casado-Vara *et al.*, 2021). Casado-Vara *et al.* (2021) also stated that time series analysis of network traffic has helped companies offer their client good response time to their requests to online services as customers experiencing lengthy waiting times abandon delayed services. Online video gaming is a good example of online services that companies are offering. Companies need to know when most of their client play online so that they can manage their resources and offer ideal gaming experience to their customers. Long Short Term Memory (LSTM) Recurrent Neural Network and Autoregressive integrated moving average have been used

to achieve this purpose (Shelatkar *et al.*, 2020; Casado-Vara *et al.*, 2021).

Casado-Vara *et al.* (2021) developed an architecture for web traffic forecasting based on artificial intelligence with LSTM for time series forecasting. This architecture offered real time data on how organisation can help improve customer's user experience online. By detecting unusual patterns in network traffic check point were able to identify the re-emergence of SQL Slammer worm. This was reported in work done by (Chindipha and Irwin, 2017) where they used IBR data to confirm the observations made by Check point software. Essentially, in both studies, a time series analysis showed unusual high volume of traffic that was not recorded before November 2016. Thus using such an unusual trend, these researchers were able to analyse the network traffic and identify the cause of this anomaly.

Cortez *et al.* (2006) developed three time series neural network methods that had the capability of forecasting the amount of traffic in TCP/IP based networks by using. The objective was to create tools that would improve anomaly detection in traffic. Cortez *et al.* (2012) repeated the experiment using an adapted novel neural network ensemble approach and time series methods (ARIMA and Holt-Winters). They also used real-world data collected from two large Internet source providers (ISPs) which were used a training datasets. They made predictions every 5 min, hourly and daily with minor errors. This improved and validated their results when compared to their earlier experiment.

These are just some of the practical applications of time series data once it has been analysed and meaning is drawn from it. Times series analyses are significant in the economic sectors, astrology, health sectors and our daily livelihood, thus their significance cannot be overemphasised.

2.7 Chapter Summary

This chapter introduced the reader to network telescope which monitors a portion of routed IP address blocks on which little or no legitimate traffic exists. Thus monitoring unexpected traffic arriving at a network telescope using network sensors brings about IBR data. This was explained in **Section 2.1**. **Section 2.2** expanded on this by explaining what Internet background radiation (IBR) data is. This was the main source of data used in this research. **Section 2.3** discussed more on the feasibility of having a small-sized network telescope and what one can achieve with it. This was immediately followed by **Section 2.4** which discussed some of the scenarios that show how IBR and network

telescopes were used to gather threat intelligence data. This led to **Section 2.5** which explained why IPv4 addresses are running out so rapidly and shed more light on the implications of this exhaustion on Internet users. Considering that the research is dealing with data that has timestamps in it, the study introduced concepts related to time series analysis in **Section 2.6**. This is the knowledge that is needed in order to understand why the statistical techniques discussed in **Chapter 3** were chosen.

3

Review of Statistical Techniques

This chapter introduces the reader to the statistical techniques that were used in this research study. The nature of the data demanded certain statistical techniques in order to achieve the objectives introduced in **Chapter 1**. The tools introduced in this chapter are used in **Chapters 5** and **6.**

The chapter begins by giving a brief introduction of data sampling in Section **Section 3.1**. This is immediately followed by **Section 3.2** which introduces Bootstrapping. It is in this section that background information regarding how bootstrapping has been used by other researchers is introduced. This chapter also looked at strengths and limitations of Bootstrapping techniques in **Section 3.3**. This leads to **Section 3.4** which explains and introduces a reader to Confidence Interval (CI) and links it to bootstrapping. This section also addresses the limitations that come with computing CI using bootstrapping in. In **Section 3.5**, the study explains how other researchers used bootstrapping, thus application of bootstrapping and confidence interval are in this section. This section then leads into **Section 3.6** which introduces regression analysis to the reader before expanding into **Section 3.7** to talk about types of regression analysis that were considered

for this study. From here, the study focused on what features constitute a mathematical model. The concepts and steps needed to formulate mathematical models are explained in **Section 3.8**

Time series similarity scoring techniques are introduced in **Section 3.9**. It is in this section that the study explained how Mean Absolute Percentage Error (MAPE), Symmetric Mean Absolute Percentage Error (SMAPE), Mean Absolute Error (MAE) and Mean Absolute Scaled Error (MASE) are used in this study. **Section 3.10** concludes the chapter by explaining the information retrieval techniques that were considered for this research and how they were used in this study. The techniques explained in this chapter are used in **Chapters 5** and **6.**

3.1 Data Sampling

Candès and Wakin (2008) defined data sampling as a process where a portion of the larger dataset (baseline dataset) is analysed and exposed to manipulation to identify how representable the sample is to the baseline dataset. Data sampling is a long-standing technique when it comes to statistical analysis, and without such techniques to gauge the samples, it is difficult to evaluate how representative the samples are to the original dataset (Aizawa, 2003).

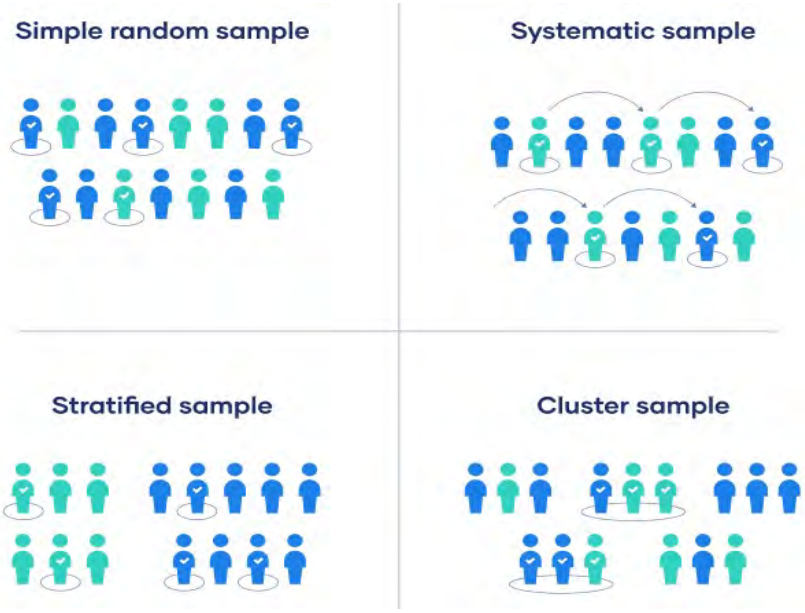


Figure 3.1: Examples of sampling methods (McCombes, 2019)

Data sampling can be done using multiple methods, but this study focused on sequential and random sampling as explained by Parthasarathy (2005). **Figure 3.1** shows some of the methods that are used when sampling data. For each of the methods of data sampling, a portion of the large dataset is selected using a specific criteria. In **Figure 3.1**, each of the circled individual or group have been selected based on various criterion as described in the figure. The findings from these selected individuals are then applied to the overall population.

Data sampling helps us in addressing the central question this study aims to address: how accurately does the identified sample (referred to as *subnet equivalent*) reflect the baseline dataset? However, to answer this question, various techniques have been put in place to assess the viability of such samples combined with their sampling techniques. The two main criteria used were sequential and random sampling. Sampling is done in this research study in order to quantify the differences that exist between the baseline dataset and the samples drawn from it. More statistical methods as discussed in **Sections 3.6**, **3.8**, **3.9** and **3.10** were applied to analyse the differences in datasets created from sequential and random sampling techniques.

3.2 Bootstrapping

Bootstrapping works on the principle of starting with a dataset with an unknown underlying distribution from which a sample, which is a partially randomised version of the available data, is selected (Kirby and Gerlanc, 2013; Efron and Hastie, 2016; Marcaccioli and Livan, 2020). Using any specific population parameter of interest (it could be mean, standard deviation, variance etc), a normal distribution is formulated by applying the statistical function to the parameter of interest (Martin, 1990; Davison and Kuonen, 2002; Kirby and Gerlanc, 2013; Chamandy *et al.*, 2015).

The re-sampled data, which approximates the normal distribution, is formulated through simulation of the original data using the statistical function of interest as the guiding population parameter of interest (Zoubir and Iskandler, 2007; Kirby and Gerlanc, 2013; Chamandy *et al.*, 2015). Essentially, bootstrap eliminates the unknown variables (e.g. unknown data distribution, stationarity) that are presented in the original data by simulating the data to come up with known variables from which values like confidence interval (CI) can be computed using the plug-in-principle (Zoubir and Iskandler, 2007; Kreiss and Lahiri, 2012; Chamandy *et al.*, 2015; Hesterberg, 2015). This study used the *mean* value as the statistical parameter of interest to bootstrap IBR data.

Kreiss and Lahiri (2012); Hesterberg (2015) reported that the plug-in principle works on the notion of computing an estimate of the unknown variable and replacing that which is unknown in order to understand a population under study. Thus, the unknown parameter(s) is substituted for the ones computed as an estimate of what was not known initially. In this study, the mean (average number of unique SRCIP observed per DSTIP) was used as the parameter of interest. The *mean* is first calculated from the baseline dataset and the value computed becomes the value of interest.

Through the simulation process, using our population parameter estimator (mean), this study reused the original data through re-sampling to create a new data sample which is referred to as a bootstrap sample by Efron (1992); Zoubir and Iskandler (2007), among other authors. With known variables being present in the bootstrap sample, one can proceed and compute the confidence interval (Kirby and Gerlanc, 2013), an estimate of the shape of the sampling distribution, an estimate of the standard error of the quantity (standard error of the mean (SEM) was computed for this study) (Hesterberg, 2015) and an estimate of bias and P-value (Rousselet *et al.*, 2019). This research is more interested in the CI side of bootstrapping and it is this aspect that this report will focus on. This is in line with the study's goal of offering a certain degree of assurance and reliability of the data to whoever is going to use the data.

One can choose to ensure that the bootstrap samples have the same size as the original data or make the samples larger than the original dataset by sampling from the original data with replacement on intention of estimating the impact of large samples on the standard errors of the data (Kirby and Gerlanc, 2013; Chamandy *et al.*, 2015; Efron and Hastie, 2016). When creating a bootstrap sample, each bootstrap sample consists of a simple random sample selected with replacement from the total number of observations from the original data.

When creating a bootstrap sample with the same size as the original dataset, the aim is to ensure that the standard errors observed in the original dataset are reflected in our bootstrap sample as compared to having hypothetically larger or smaller samples (Dixon, 2006; Hesterberg, 2015). This essentially means that the number of observations can be the same but the composition will not be the same. This is particularly important to our research study because the different samples being compared do not have the same number unique SRCIPs, however, for comparability purposes, they are designed to contain the same number of data points when bootstrapping. Dixon (2006); Hesterberg (2015) observed that better accuracy in the outcome of the estimated output of the bootstrap samples is ensured when samples that are larger than 10,000 are used.

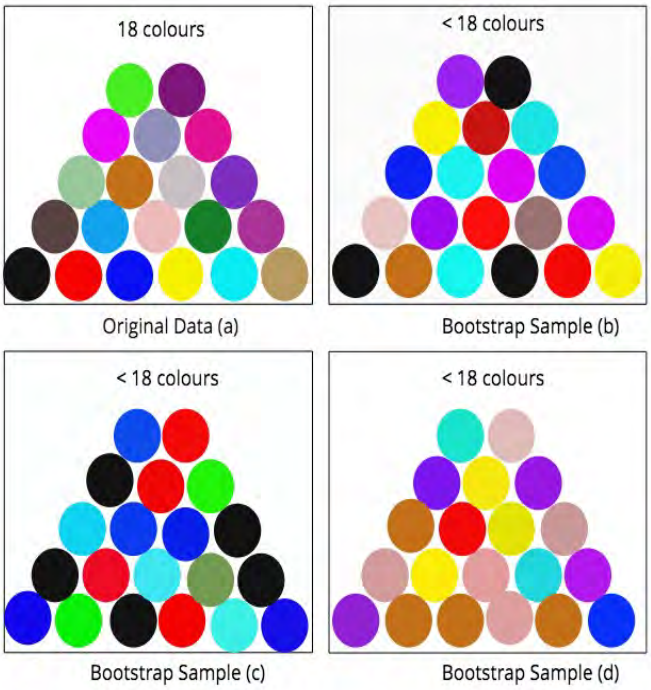


Figure 3.2: The Independent data bootstrapping re-sampling principle

Figure 3.2 shows the graphical representation of the principle behind bootstrap re-sampling with replacement with the aim of ensuring that the bootstrap sample has the same number of observations as the original data, irrespective of having a different composition of the observations. As it can be seen in **Figure 3.2**, the *original data (a)* represents the baseline datasets, while *bootstrap samples (b), (c)* and *(d)* represent different variations of bootstrapping.

The reader will note that samples shown in **Figure 3.2 b - d** have some original observations sampled more than once while others are not sampled at all. For each bootstrap sample, an estimate is computed (sample mean was used in this study for non-parametric bootstrapping as our population parameter), thus if one draws out and simulates 10,000 bootstrap samples and computes the mean for each of them as the population parameter of interest, one would end up with an approximately normal distribution of 10,000 bootstrap estimates of the sample mean, which now becomes the new dataset from which computations are based. More details on non-parametric bootstrapping are explained in **Section 3.2.2**.

Depending on the random observations selected from the baseline data, the value of the mean may vary as it is computed from the selected observations. Large bootstrap samples

tend to have very small variations in their mean (or any statistical parameters of interest) as the sample pool is large enough to offer a variety of observations (Hesterberg, 2015; Efron and Hastie, 2016; Rousselet *et al.*, 2019). It is good practice to ensure that the bootstrap samples (also known as re-samples) contain the same number of data points or observations as the original dataset in order to minimise biases and variability of the population parameters (Dixon, 2006; Kirby and Gerlanc, 2013; Efron and Hastie, 2016). If the bootstrap sample, as well as the baseline dataset, are large enough, the distribution of bootstrap sample estimates provides a good approximation of the sampling distribution of the baseline dataset, thereby making inference practical (Kreiss and Lahiri, 2012; Hesterberg, 2015; Rousselet *et al.*, 2019).

It is also important to note that all values that constitute the bootstrap sample are independent and identically distributed (IID) since each observation within the pool of baseline dataset has an equal probability of being a member of the bootstrap sample (Efron, 1992; Kreiss and Lahiri, 2012; Efron and Hastie, 2016). Efron and Hastie (2016) showed that an empirical probability distribution of the bootstrap sample maximises the probability of obtaining the observed samples under all possible choices hence i.i.d. (Efron, 1992; Dixon, 2006; Kreiss and Lahiri, 2012; Chamandy *et al.*, 2015).

Having explained the core of bootstrapping, **subsections 3.2.1** and **3.2.2** will explain the two main categories of bootstrapping. In this study, bootstrapping was categorised into two parts: parametric and non-parametric bootstrapping.

3.2.1 Parametric Bootstrapping

Parametric bootstrapping involves the need to make parametric assumptions based on an underlying equation or any specific model (Martin, 1990; Zoubir and Iskandler, 2007; Rousselet *et al.*, 2019; Marcaccioli and Livan, 2020). In parametric bootstrapping, the assumptions are made on the underlying model or equation and the parameters of that model (Martin, 1990; Kreiss and Lahiri, 2012; Hesterberg, 2015). Our bootstrap samples are then drawn from the assumed model together with the estimated parameters. It is worth noting that the estimated parameters are in no way a replacement of the baseline estimates. This is the case because Kreiss and Lahiri (2012); Hesterberg (2015) found that the estimated mean cannot replace the baseline mean, but rather make inferences from such a computation to understand the unknown values of the baseline dataset.

Dixon (2006) observed that with parametric bootstrap, one either knows or assumes the function (or model) responsible for the distribution of their bootstrap sample coupled

with unknown parameters that are revealed once the sample is computed. In some cases, one does not need to use the model but explicitly state the distribution observed in the original data and use it as a baseline. Either way, the bootstrap sample is generated on explicitly stated assumptions and with replacement (Dixon, 2006). More often than not, the unknown parameters estimated offer maximum likelihood estimates to cover as much ground as possible (Dixon, 2006). Results for parametric bootstrapping are reported in **Sections 5.5.1** and **5.5.3**. The research approach used to process the data for parametric bootstrapping is explained in **Section 5.2**

3.2.2 Non-Parametric Bootstrapping

Independent research studies conducted by Kreiss and Lahiri (2012); Hesterberg (2015); Marcaccioli and Livan (2020) reported that non-parametric bootstrap draws its samples from an empirical distribution sample generated from an estimate of the standard error of the quantity. Non-parametric bootstrapping involves none of the assumptions made in parametric bootstrapping, but rather is based on an estimate of the standard error of the quantity. Dixon (2006) explained that non-parametric bootstrapping usually constitutes large sample sizes and numerous simulations to get the appropriate bootstrap sample correctly.

Martin (1990); Simar and Wilson (1998) found that it is also possible to make adjustments to a non-parametric bootstrap by ensuring that samples are done with replacement (often referred to as bootstrap with replacement). This was the approach that was taken in this study while it maintained the use of non-parametric bootstrapping because no assumptions were made regarding its distribution or parameters. As such, it still qualifies to be called non-parametric bootstrap, something that Simar and Wilson (1998) partly alluded to in his work. This approach improves the accuracy of the results and some of the criticism offered in its lack of having statistical basis when formulating its bootstrap sample (Simar and Wilson, 1998).

For this study, the mean of the individual datasets was chosen as the estimate of the standard error of the quantity. It is the result from this statistical function of the mean (upon simulation) from which parameters of interest, like the maximum likelihood CI and standard errors, are computed from (Dixon, 2006; Zoubir and Iskandler, 2007; Wilcox, 2011; Hesterberg, 2015). A detailed research approach for computing CI for non-parametric bootstrapping is presented in **Section 5.2**

It is worth noting that median, standard deviation, variance or any quantile can be used as an estimate of the standard error of the quantity when creating a non-parametric bootstrap sample (Dixon, 2006; Kreiss and Lahiri, 2012; Rousselet *et al.*, 2019). The main assumption when it comes to non-parametric bootstrapping is that the distribution of the bootstrap sample more often than not take the shape of the data from which it was taken (Kreiss and Lahiri, 2012; Rousselet *et al.*, 2019). The re-sampling with replacement involved in computing their confidence interval makes both parametric and non-parametric bootstrap fall under the main category of percentile bootstrap (Dixon, 2006; Wilcox, 2011; Rousselet *et al.*, 2019). In this study, a linear regression model was used as the underlying model from which computations of the percentile confidence interval were calculated. The same applies to its graphical representation (bootstrap distribution) which are based on the assumption that the data follow the linear regression model.

3.3 Strengths and Limitations of Bootstrapping Techniques

Chan (2003); Higgins (2004) observed that with parametric bootstrap, it is the same as calculating things in a normal distribution assumption. Chan (2003); Bagdonavicius *et al.* (2013) reported that using the assumption of a certain distribution brings an extra accuracy in the parametric bootstrap over the non-parametric bootstrapping if the assumptions made are correct. What this means is that if the assumptions made about the data distribution and model used are wrong, it will in turn affect the accuracy computed from such a dataset. This inverse observation was also reported by Efron and Tibshirani (1991); Chan (2003); Higgins (2004)in their separate studies. The application of parametric tests requires various assumptions to be satisfied first. For example, the first assumption is that data follows a normal distribution and the population variance is homogeneous. However, some data samples may show skewed distributions, and in such cases, parametric bootstrapping will give poor results as it is affected by the extreme values (Chan, 2003; Higgins, 2004; Bagdonavicius *et al.*, 2013).

Another limitation of parametric bootstrapping is the size of the population. If a sample size is reasonably large, the applicable parametric sampling can be used (Chan, 2003; Higgins, 2004). However, if the sample size is too small, one may not be able to validate the distribution of the data (Chan, 2003; Higgins, 2004). Thus, the application of non-parametric bootstrapping becomes the only suitable option.

The benefits of the non-parametric bootstrap lie in the fact that one does not need to assume any distribution or assume any specific models (Bronars, 1987; Chan, 2003; Higgins, 2004). As such, there are fewer assumptions made when using non-parametric bootstrapping and the assumption of data being normally distributed do not apply. Thus, in cases where the underlying data does not meet the assumptions about the population sample, non-parametric bootstrapping becomes very useful in giving accurate results (Chan, 2003; Higgins, 2004).

Unlike parametric tests that can only work with continuous data, non-parametric bootstrapping can be used for all data types (Chan, 2003; Higgins, 2004). These data types include: data with nominal variables, interval variables, or data that has outliers or that have been measured imprecisely (Chan, 2003; Higgins, 2004; Bagdonavicius *et al.*, 2013). For such types of variables, non-parametric bootstrapping are the only appropriate solution. Furthermore, small sample sizes are acceptable with non-parametric bootstrapping (Chan, 2003; Bagdonavicius *et al.*, 2013). However, in cases where assumptions haven't been violated, non-parametric bootstrapping is less powerful than parametric bootstrapping.

All the parameters used are computed first from the baseline dataset (Chan, 2003; Higgins, 2004). In both parametric and non-parametric bootstrapping cases, nothing new was learnt that was not in the original data as all computations are based on the baseline dataset (Bronars, 1987; Bagdonavicius *et al.*, 2013). In addition to this, both parametric and non-parametric techniques inherit errors that came with the original data i.e. they do not eliminate errors found in the baseline data (Efron, 1983; Higgins, 2004; Bagdonavicius *et al.*, 2013)

3.4 Confidence Interval

Dixon (2006) defined confidence interval coverage as the probability that the confidence interval includes the true parameter under repeated sampling from the same underlying population. The true parameter is the actual computed value that one would expect to find in the dataset being analysed. For instance, the study presented in this report is using the mean as is population parameter. Thus the true parameter is the computed mean value of the dataset. Mean was chosen because of its overall representativeness on the data. This has been presented in great detail in **Section 5.2**.

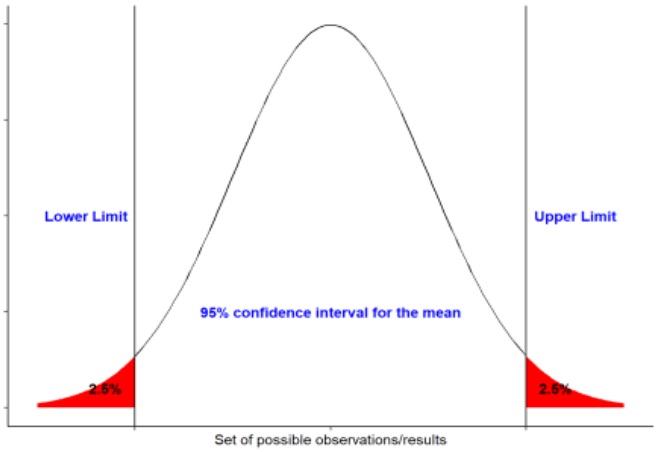


Figure 3.3: 95% Confidence Interval of a Normal Distribution (Fneish, 2021)

Having sampled from the baseline dataset multiple times, it is expected that the mean value will fall in the computed range. Assume the interval given is between 450 and 469 unique source IP (SRCIP) addresses per destination IP (DSTIP) computed at 95% CI. If the researchers take 100 random samples from the baseline population, then they should expect the mean to fall between 450 and 469 unique SRCIP addresses per DSTIP in 95 of those samples. If the researchers want even greater confidence, they can expand the interval to 99% confidence. Doing so invariably creates a broader range, as it makes room for a greater number of sample means. From here onward, Confidence Interval will be referred to as CI. If they establish the 99% confidence interval as being between 450 and 469 unique SRCIP addresses per DSTIP, in the case of the data under study (see **Chapter 4**), they can expect 99 of 100 samples evaluated to contain a mean value between these numbers. The true average number of unique SRCIP addresses per DSTIP for each sampled population is likely covered by a range of values called a confidence interval (Dixon, 2006). Each sample will definitely have its own mean, which acts as the true parameter.

Working with this definition, it essentially means that our value of interest after multiple re-sampling from the baseline data, our parameter of interest (average number of unique SRCIP addresses per DSTIP) ought to fall within the range generated by the bootstrap sample created. The range computed from the bootstrap sample is what gives the user the level of confidence needed in the data in order to make informed decisions. When computing the confidence intervals, this study used the quantile ranges in order to compute a specific CI (Rousselet *et al.*, 2021; Marcaccioli and Livan, 2020).

The rule of thumb is that for each CI (the overall confidence level selected) the value is split into two to accommodate both the lower limit and the upper limit to imitate two

standard deviations from the mean principle (Dixon, 2006). For instance, in a 95 % CI, the lower bound is defined as the 2.5th quantile of the bootstrap distribution, and the upper bound at the 97.5th quantile. **Figure 3.3** shows a graphical representation of how a 95 % CI would look like. The lower limit is placed at 2.5% which is the 2.5th quantile while the upper limit is placed at 97.5th quantile. Adding all proportions gives the reader a 100% dataset.

On the other hand, if the computation is being made at 95% CI, it means the study is only interested in the values under the curve and between the lower limit and the upper limit. That interval will give a set of all possible values needed for that CI. If 99 % CI is computed, the lower bound will be defined as the 0.5th quantile of the bootstrap distribution, and the upper bound at the 99.5th quantile (Dixon, 2006; Rousselet *et al.*, 2017).

Hesterberg (2015) presented work that showed that bootstrap samples that are generated from smaller baseline datasets tend to have a wider range for their CIs because the spreads and shapes of the samples vary substantially. Empirical and theoretical studies of coverage conducted by Tibshirani and Efron (1993); Davison and Hinkley (1997) showed that the percentile CI may not always give accurate results i.e. it works very well in some cases while in other cases, the likelihood of one getting accurate results is very slim. Dixon (2006); Rousselet *et al.* (2019) observed that the accuracy of CI is dependent on how the CI endpoints are calculated, the size of the bootstrap sample, the type of bootstrapping used (whether it is parametric or non-parametric), the statistical functions used as a parameter estimator, and how the bootstrap samples are selected. Davison and Hinkley (1997); Dixon (2006) conducted separate studies that showed that skewed sampling distribution tends to be less accurate as compared to symmetrical sampling distribution when calculating endpoints of CI.

The size of the sample as well as the method used to compute the endpoints of CI affect the accuracy of the coverage. Using two different forms of bootstrapping helps to evaluate the extent to which these errors affect the data under study. For instance, non-parametric bootstrapping is not affected by the size of the data while parametric bootstrapping can be affected with very small sizes. Martin (1990); Dixon (2006) observed that increasing the number of simulations when generating bootstrap samples boosts the confidence in the confidence interval computed from the sample. Application and findings of bootstrapping that use confidence interval are presented in **Section 3.5**.

3.4.1 Limitations of Bootstrapping that use CI

Despite its many advantages, including computational transparency, bootstrapping that uses CIs has setbacks emanating from sources of inaccuracy (DiCiccio and Efron, 1996; Kirby and Gerlanc, 2013). First, the work done by Kirby and Gerlanc (2013) showed that many sample statistics used are biased estimators of their corresponding population parameters, such that the expected value of the population parameter of interest does not equal the actual value of the baseline dataset. In addition to this, the standard error of an estimate of the population parameter of interest may not be independent of the value of the original parameter of the baseline dataset (Kirby and Gerlanc, 2013). Consequently, even for unbiased estimates, the lower and upper percentile cutoffs may not be the same number of standard error units from the estimates' population parameter (Kirby and Gerlanc, 2013).

When empirical probability distribution of the bootstrap samples approaches the distribution of the baseline dataset as the number of observations in the bootstrap sample grows large, only then does the standard error of the bootstrap approach the true standard error of the baseline's parameter of interest (Kreiss and Lahiri, 2012; Efron and Hastie, 2016). Another setback of bootstrapping is that it is less accurate when working with smaller samples, especially when computing percentile bootstrap confidence interval (Bronars, 1987; Kreiss and Lahiri, 2012; Hesterberg, 2015). On the other hand, it works very well with large samples. When working with smaller samples, one is better placed to work with t-tests as compared to any of the bootstrap techniques as this is an inherent setback in all bootstrap techniques (Chan, 2003; Hesterberg, 2015).

3.5 Applications of Bootstrapping and Confidence Intervals

In this section, the study will show some of the real-world use cases of bootstrapping and confidence interval. As explained in **Section 3.2**, bootstrapping is a simulation tool that uses random sampling with replacement to estimate a sampling distribution for a given statistic, and that the goal of this sampling is to accurately represent a population of inference. In other words, bootstrapping is a computationally intensive statistical technique that allows the researcher to make inferences from data without making strong distributional assumptions about the data or the statistic being calculated (Haukoos and

Lewis, 2005; Wood, 2005). This allows the researcher to estimate CI for statistics that do not have simple sampling distributions. CI can be paraphrased as a range of plausible values from which the true population parameter under study can be observed. The effectiveness of a CI is judged by whether or not it contains the true value of the population parameter (Haukoos and Lewis, 2005; Wood, 2005).

The use of CI has formed an integral part of the research, be it in evaluating the performance of models and giving confidence to users in the data, or in understanding the behaviour of variables unknown to the researchers Haukoos and Lewis (2005). When models are developed, irrespective of the fact that they give accurate results, research demands that there has to be another metric needed to validate the outcome of the results. CI has emerged as one of the statistical techniques that researchers use to give confidence in the performance of the models they are either using or that they developed (Wood, 2005). CI has also been used to test the efficacy of drugs when an organisation is conducting clinical trials (Bender, 2001). Before the drug is released, the organisations manufacturing the drug want to give reassurance to their clients in how effective the new drug is in combating the problem at hand. Thus a series of tests are conducted at different levels of CI to support the argument of how effective the new treatment is.

Bootstrapping has been used to study the effect of sampling variation by quantifying the variation of the sample estimates using their standard deviation (Ismay and Kim, 2019). In addition to this, Ismay and Kim (2019) did two case studies using bootstrapping and CI to uncover unknown variables in a population using sample data that was bootstrapped. In one of the real-life case studies, Ismay and Kim (2019) showed how they used bootstrapping to understand polls that were conducted by the Kennedy School's Institute of Politics at Harvard University[1]. In the study, 41% of millennials (adults ages 18-29) approved of Obama's job performance. However, the online survey only had 2,089 participants and had no idea of the population size of the millennials who shared the same views.

Using bootstrapping, Ismay and Kim (2019) showed how the organisers could have made sense of the data that could have been representable of the total population of millennials. In a different study, Ismay and Kim (2019) tested the myth of how contagious yawning can be by computing confidence interval using bootstrapping. The ability that bootstrapping has in making sense of sample data through its rigorous simulation process has proved beneficial in cutting down costs when conducting censuses as the random sample datasets

[1]https://www.npr.org/sections/itsallpolitics/2013/12/04/248793753/poll-support-for-obama-among-young-americans-eroding

used can be considered as unbiased and representative of the population. Thus any results based on the sample could be generalized to the population (Efron and Hastie, 2016; Ismay and Kim, 2019).

Psychologists have benefited from bootstrapping by applying bootstrap estimation to data from clinical samples and measures relevant to experimental psychopathology (Wright *et al.*, 2011; Field and Wilcox, 2017). In so doing, psychologists have been able to understand patients' responses to treatments and make inferences from such data. Psychologists often tried to fit their hypotheses to a small but well-known set of statistics with mathematical formulae for calculating the standard error, however, bootstrapping has allowed psychologists to avoid this restrictive approach (Wright *et al.*, 2011; Field and Wilcox, 2017). Bootstrapping had also been used to understand the behaviour of employees in an organisation by bypassing the need to interview every employee to address a problem at hand (Howell, 2012). With, bootstrapping, a sample is interviewed and inferences made from it. Each time, this approach has proved to deliver the intended results.

3.6 Regression Analysis

Rawlings *et al.* (2001); Chatterjee and Hadi (2015) presented regression analysis as a set of statistical methods used for the estimation of relationships between the main variable that an individual is trying to understand or predict (referred to as a dependent variable) and one or more variables that are suspected to have an impact on the dependent variable of interest (referred to as independent variables). Regression analysis can be used to assess the strength of the relationship between variables but also for modelling the future relationship between the variables of interest (Rawlings *et al.*, 2001; Chatterjee and Hadi, 2015). An individual goes through the path of regression analysis with the aim of quantifying how each of the variables affect each other. If there is variability, then the individual mathematically sorts out which of those variables has more impact than the other (Rawlings *et al.*, 2001). **Section 5.4** shows how regression analysis has been applied in this study to address the research questions.

Among the questions addressed by regression analysis include, but are not limited to, the following: Which variables in the data under study matter most? Which ones can be ignored? How are the variables of interest interacting with each other? How certain or confident are we about all of the identified variables of interest? These questions were reported by Chatterjee and Hadi (2015) in their work.

In order to answer these questions, one is required to draw a scatter plot that shows the relationship that exists on the variables of interest in the dataset of interest. A line of best fit is then drawn on the plot to offer the degree of certainty on the relationship among the variables of interest. The line of best fit best describes the relationship that exists between the dependent variable and the independent variable(s) found in one's dataset (Rawlings *et al.*, 2001; Chatterjee and Hadi, 2015). It is worth noting that the line of best fit is the best estimate of the available dataset and may not always be the same if some margin of error occurs within the dataset, bringing about variability (Rawlings *et al.*, 2001; Chatterjee and Hadi, 2015). Rawlings *et al.* (2001) proved that a small margin of error offers more confidence in the line of best fit drawn from a dataset, and that the relationship that exists among the variables and a large margin of error gives less confidence in the data and the line that describes the relationship of the variables of interest.

The applications of regression analysis are limitless, especially in the research field. For instance, regression analysis can be used as a tool for predictive analytics and forecasting in market research (Laitinen, 2018; Fisher and Kordupleski, 2019). Considering that it is used to understand the relationship between two or more variables, regression analysis can be used to understand how a company's revenue can be impacted by the ups and downs of oil prices, the consumer price index (CPI), and gross domestic product (GDP) (Al-Tamimi *et al.*, 2011; Barakat *et al.*, 2016). A clear understanding of how sales of an organisation are affected by consumer preferences can help businesses and organizations prioritise efforts to improve measures like overall satisfaction, likelihood to recommend, or net promoter score (NPS) is presented by Laitinen (2018); Fisher and Kordupleski (2019). Using regression analysis in quantitative research provides the opportunity to take corrective actions on the items that will most positively improve overall satisfaction.

3.7 Types of Regression Analysis

Regression analysis includes several approaches. These include techniques such as linear, multiple linear, and nonlinear as explained by Rawlings *et al.* (2001); Chatterjee and Hadi (2015). The most common models are simple linear and multiple linear. Nonlinear regression analysis is commonly used for more complicated data sets in which the dependent and independent variables show a nonlinear relationship (Rawlings *et al.*, 2001). This research focused on linear and non-linear regression only, and that is what this subsection will focus on. During the experimental stage, a test was done to see how the data would

perform using multiple regression analysis, but the results were very poor, resulting in them being left out of this study. In order to use linear regression analysis, for your data, the data in question must meet the baseline assumptions needed to qualify for use. This is critical as it validates the quality of your results.The baseline assumptions form the benchmark from which results can be compared against.

3.7.1 Linear Regression Analysis

Linear regression analysis is used when one wants to predict the value of a dependent variable based on the value of the independent variable (Twomey and Kroll, 2008; Montgomery *et al.*, 2012; Hoffmann and Shafer, 2015). Unlike multiple regression, which has two or more independent variables, linear regression has one independent variable to work with (Montgomery *et al.*, 2012; Hoffmann and Shafer, 2015). When using linear regression, there are a number of factors that need to be put into consideration. Firstly, there is a need to ensure that there is a linear relationship between the two variables under study, which is usually done using a scatter plot (Twomey and Kroll, 2008; Montgomery *et al.*, 2012; Hoffmann and Shafer, 2015).

If the relationship displayed in one's scatter plot shows linearity, one will have to resort to another form of regression analysis; either non-linear or polynomial regression (Sinan and Alkan, 2015). This was the approach used in this study. Twomey and Kroll (2008); Montgomery *et al.* (2012) placed more emphasis on making sure that the independent and dependent variables are measured at the continuous level i.e. they are either interval (their central characteristic is that they can be measured along a continuum and they have a numerical value) or ratio variables (interval variables, but with the added condition that 0 (zero) of the measurement indicates that there is none of that variable) (Montgomery *et al.*, 2012).

Thirdly, there should be no significant outliers on the scatter plot (Twomey and Kroll, 2008; Hoffmann and Shafer, 2015). An outlier with regard to scatter plot will be any point on a scatter plot that is (vertically) far away from the regression line, indicating that it has a large residual (error) (Sinan and Alkan, 2015). The plots are plotted at 95% CI of the data to identify the outliers (within 2 standard deviations of the mean). Any data point out of this range is identified as an outlier. One major drawback that outliers pose regarding regression analysis is their overall negative effect on the regression analysis, thereby their fitness on the regression equation is questionable (Twomey and Kroll, 2008; Hoffmann and Shafer, 2015; Sinan and Alkan, 2015). If the equation is negatively affected,

then the values computed from it will be inaccurate (Sinan and Alkan, 2015). The outliers presented may often be as a result statistical anomalies or produced by errors in the measurements that would not affect the curves, hence are usually ignored.

Another point of consideration when using linear regression is to check for the independence of observations within the variables of interest (Montgomery *et al.*, 2012; Hoffmann and Shafer, 2015). Usually, Durbin-Watson (DW) statistic as presented by (Akter, 2014), is used to test for auto-correlation in a dataset i.e. independence of observations. The DW statistic always has a value of between zero and 4. A value of 2 means there is no auto-correlation detected in the sample (Akter, 2014). Values from zero to 2 indicate positive auto-correlation, and values from 2 to 4 indicate negative auto-correlation (Akter, 2014). Lastly, the reader needs to check that the residuals (errors) of the regression line are approximately normally distributed (Hoffmann and Shafer, 2015). Two common methods to check this assumption include using either a histogram (with a superimposed normal curve) or a Normal P-P Plot (Hoffmann and Shafer, 2015; Das and Imon, 2016). This study has parametric and non-parametric bootstrap sample plots to check this at different confidence intervals.

3.7.2 Non-Linear Regression Analysis

Smyth (2006); Archontoulis and Miguez (2015) explained that a non-linear regression analysis is used when a series of events do not clearly or directly follow from another i.e there is no direct relationship between the dependent and independent variable. Nonlinear regression is a method of finding a nonlinear model of the relationship between the dependent variable and a set of independent variables (Smyth, 2006; Archontoulis and Miguez, 2015) i.e. relates the depended and independent variables in a nonlinear (curved) relationship. Unlike traditional linear regression, which is restricted to estimating linear models, nonlinear regression can estimate models with unpredictable relationships between independent and dependent variables (Archontoulis and Miguez, 2015). This is accomplished using iterative estimation algorithms (Kass, 1990; Baty *et al.*, 2015).

Chen *et al.* (2020) presented work that showed that non-linear regression is often more accurate as it learns the variations and dependencies of the data. Worthy of note is the realisation that any relationship that is not linear, can be termed as non-linear and is usually represented by the polynomial of $\boldsymbol{k}$ degrees (maximum power of X) (Ostertagová, 2012; Archontoulis and Miguez, 2015). Many different nonlinear regression models exist that may be used to fit whatever the data set looks like and these can go on to infinite

degrees (Ostertagová, 2012). For this study, sixth degree was used in order to find the results that best fit and described the data. sixth degree was used because at this level, that is when the highest levels of accuracy were observed (the best fit was found at this sixth degree, giving a better representation of the data.). This is shown in **Section 5.4**. If one uses a wrong $\boldsymbol{k}$ degree for the polynomial at hand, the accuracy of the model is negatively affected.

The overall objective of a nonlinear model is to make the sum of the squares as small as possible (Kass, 1990; Motulsky and Brown, 2006; Sanft and Walter, 2020) i.e. the smaller the sum of these squared figures, the better the function fits the data points in the set. The sum of squares is a measure that tracks how far the dependent observations vary from the nonlinear (curved) function that is used to predict dependent variables (Motulsky and Brown, 2006; Sanft and Walter, 2020). These distances are called residuals (Motulsky and Brown, 2006). The sum of squares is significant in regression analysis as it is used to determine the fitness of a regression model, which is computed by calculating the difference between the mean and every point of data (Sanft and Walter, 2020). Nonlinear regression analysis mainly concerns the prediction of responses, statistical inferences of parameters, estimates, and the goodness of fit of the nonlinear model (Kass, 1990).

One advantage that nonlinear regression models have over linear regression is their ability to accommodate different mean functions (Ritz and Streibig, 2008). In order to obtain accurate results from the nonlinear regression model, one ought to make sure the function specified describes the relationship between the independent and dependent variables accurately (Ritz and Streibig, 2008; Sanft and Walter, 2020). If this step is missed, all the results and interpretation of the analysis will be wrong (Baty *et al.*, 2015). The value of the coefficients can be correctly interpreted only if the correct model has been fitted. Therefore, it is important to identify useful models (Motulsky and Brown, 2006). The selection of the model to use is based on the theory from which the data being analysed is based and past experience in the field (Motulsky and Brown, 2006). For example, in demographics, for the study of population growth, the logistic nonlinear regression growth model is useful. Nonlinear regression also assumes that the data being used is quantitative, and as such, categorical data must be coded as binary (Ritz and Streibig, 2008). Real-world datasets tend to find meaningful applications when analysed using nonlinear regression models than linear models (Chen *et al.*, 2020). **Section 5.4** provides the research finding on regression analysis. i.e. the study evaluates the relationship between time and the number of unique SRCIP observed per DSTIP.

3.8 Mathematical Modeling

Mathematical modelling is a cyclical process in which real-life problems are translated into mathematical language, solved within an algebraic system, and the solutions tested back within the real-life system (Bush and Mosteller, 1951; Barbosa, 2003; Barnes and Fulford, 2011). Some of the good examples of applications that use mathematical modelling on daily basis are google maps when a driver wants to find the fastest route from point A to point B. The weather application and meteorology departments use mathematical models to understand changes in weather patterns and calculating the likelihood of the weather changing, be it from a rainy day to a sunny one.

Another example would be when a driver wants to calculate the time taken to drive through a specific distance between two towns. Knowing the distance and preferred speed, the driver can estimate the time it takes to travel between the towns using the mathematical model that relates time to distance over speed. Engineers and architects use mathematical models when designing houses or constructing roads in order to find out the amount of resources needed for their projects. Living in the Covid-19 Pandemic era, epidemiologists have used a series of mathematical models to calculate the rate at which the virus is spreading. These are just some of the real-world applications of mathematical models.

To use mathematical modelling, one has to be presented with a real-life situation to examine its structural features or characteristics, and through the application of relevant mathematics, find a solution that solves the problem at hand (Barbosa, 2003; Blomhøj, 2009). The process of developing a mathematical model hinges on three critical points. Firstly, one developing the model needs to identify the most important parts of the problem at hand that one is trying to develop, something without which the model will not work (Barbosa, 2003; Barnes and Fulford, 2011). Secondly, knowledge about a system one is solving and the objective with which the model is being developed is also required. Models are developed for an array of reasons among which are developing scientific understanding, testing the effect of changes in a system, and aiding decision making which could be tactical in nature or strategic (Abramowitz and Stegun, 1972; Barbosa, 2003; Barnes and Fulford, 2011). In this research study, the main objective in mind when developing mathematical models was to aid model users in decision making and give them some level of confidence in the system they are using.

Whatever the objective is, each process has the same cycle which it has to go through in order to develop it. The cycle involves building the model, studying it, and testing it

(Barbosa, 2003; Blomhøj, 2009). Once that is done, the objectives that were identified at the beginning need to be considered in the usability of the model (Barnes and Fulford, 2011). If a model cannot be used, then there is no use developing it, making this last aspect crucial in this development cycle. In the development cycle, all underlying assumptions governing the model need to be laid out to the end-user (Abramowitz and Stegun, 1972; Blomhøj, 2009).

It is these assumptions that allow the model to achieve its intended objectives. Future analysis and evaluation of the system treats these assumptions as being true and the results of such an analysis are only as valid as the assumptions (Abramowitz and Stegun, 1972; Barbosa, 2003). Every user of the model needs to accommodate all these underlying assumptions, otherwise the intended objective will not be met. Where the system being modelled is more complex, one cannot simply jump from an assumption to an equation. There has to be a methodical approach, both when describing the system and when stating assumptions. In such cases, flow diagrams describing the problem offer a visual aid to this end.

Having defined the problem and made the assumptions, one needs to clearly define the problem parameters and identify all the variables that support and explain the problem defined (Barbosa, 2003; Barnes and Fulford, 2011). Defining the variables is critical because at this stage one gets an understanding of how the different parts of the problem interact with each other. Relationships are defined at this stage i.e. which of the defined variables is dependent (the information one is seeking to find from the model) and which one is independent (the information that one already had and acts as input to the model)? (Blum, 2015; Bora and Ahmed, 2019). At times, one is faced with a scenario where constants (variables that do not change) need to be taken into consideration.

Thus from the problem statement, one gets the dependent variables (output) (Bora and Ahmed, 2019). When assumptions are taken into account, coupled with further analysing of the data and brainstorming all factors to consider, one gets an understanding of the dependent, independent and fixed model parameters of the model to be developed (Bora and Ahmed, 2019; Kaiser, 2020). After these variables are defined, it is when one begins to understand how the mathematical model would work to accommodate all the factors considered. Without such definitions, one cannot develop the equation(s) which later form the model to address the problem at hand.

The tools used to test, analyse, and arrive at the intended solution, vary depending on the problem i.e. from pen and paper, to computer software tools like matrix laboratory(MATLAB), R, Statistica and Python to test and analyse the findings (Blum, 2015;

Bora and Ahmed, 2019; Kaiser, 2020). Bora and Ahmed (2019); Kaiser (2020) added the need to use every mathematical theory or topic that would help in finding the solution to the problem at hand. This would range from basic algebra, regression analysis, to calculus. Blomhøj (2009); Barnes and Fulford (2011) presented another approach to model development that involves working with the models that are already available but redefine the underlying assumptions.

Using models that are already available prevents the reinvention of some processes or steps that are already known and thus one focuses on the new dimensions or derivatives of the old mathematical model(s). When developing from models that are already in existence, the new underlying assumptions need to be stated explicitly, and as such, a derivation leads to a model that serves a different purpose than the original objective with which it was created for (Abramowitz and Stegun, 1972; Blomhøj, 2009; Barnes and Fulford, 2011). Underlying assumptions are so critical to model development to an extent that they are the reason why there is a fundamental difference between classical mechanics and relativity theory developed by Einstein (Abramowitz and Stegun, 1972). In classical mechanics, Newton assumed that mass is a universal constant, whereas Einstein considered mass as being variable (Abramowitz and Stegun, 1972). That is how critical and sensitive assumptions are to model development. The models developed in this research study took this approach of working with models that already exist while changing their underlying objectives and assumptions.

Another factor to consider is the nature of the data that a user needs to test the model and validate it (Barnes and Fulford, 2011). The behaviour of the model hinges on this point. A model can be described as qualitative, which aims to answer the question *of how* or it can be described as quantitative, when it aims to answer the question *of how much* (Blomhøj, 2009). These two questions help a model developer in knowing what approach to use in model development and the kind of data needed to answer this question. In this study, the models developed were aimed at answering the question *of how much*. The sensitivity of the model to parameter changes needs to be considered as well (Barnes and Fulford, 2011). This is important because it helps to vary model parameters and assess the associated changes in model outcomes that allow the model developer to identify its weak points (Blomhøj, 2009). Once identified, these weak points can be strengthened by experimentation, or may be simply noted for the developer to take caution in how the model is applied in a real-world scenario. Model sensitivity analysis also helps the model developer to accommodate a range of possible (acceptable) values if there is a change in value input (Blomhøj, 2009; Barnes and Fulford, 2011).

It is also important to take into consideration the possibility of errors that come along with the computation of the results of the model (errors in functional form, or to parameter estimates) and inherent errors in the data that is to be used (Barnes and Fulford, 2011). Errors can also come in due to an oversight of certain factors during model development, or the environment with which the model is being used (Barnes and Fulford, 2011). Once the model developer understands why these errors occur, then a basis for deciding how to react to them can be formulated (Blomhøj, 2009).

All of these factors are identified during the testing phase of the model development cycle. Once rigorous tests have been done, and the model developer is happy with the performance then the model can be deployed for use by others. The model needs to have proper documentation to support all the underlying conditions, methods, and parameters with which it was developed (Blomhøj, 2009).

3.9 Time Series Similarity Scoring Techniques

One of the most common analyses that are done in time series analyses is time-series forecasting (Matalas, 1967). The basic principle of time series forecasting is the computation and prediction of future values and behaviour of a time series based on past events (Contreras *et al.*, 2003; Ostashchuk, 2017). Several tests are conducted in **Chapter 6** which aim at identifying the accuracy of the forecasting. The overall objective of these tests is to quantify the differences that exist between the baseline dataset (which represent actual values) and data samples (which represent predicted values) are from the actual values. While this study is not interested in predictive analysis, the technique used in quantifying the differences in predicted and actual values fits very well with our line of study when it comes to quantifying the differences between two-time series datasets. Among the most important tests include Mean Absolute Percentage Error (MAPE), Symmetric Mean Absolute Percentage Error (SMAPE), Mean Absolute Error (MAE), and Mean Absolute Scaled Error (MASE).

3.9.1 Mean Absolute Percentage Error (MAPE)

Mean Absolute Percentage Error (MAPE) is an average of absolute percentage error (Kim and Kim, 2016). MAPE conveys the level of accuracy (exactness) of a forecasted timeline to the actual timeline as a percentage of the error (de Myttenaere *et al.*, 2016). However,

this study is not aimed at forecasting the baseline or the subnet equivalent. Instead, using the same principles of computing the accuracy of forecasted values, the study will compute the gap that exists between the baseline study and the subnet equivalents. In this case, the study compares the /24 IPv4 sub-sampled values against each of the subsequent subnet values. On the other hand, the original MAPE has forecast values. To use MAPE within the study's context, the actual values were equivalent to the baseline dataset (/24 IPv4 dataset) while the observed values were samples drawn from the baseline datasets which mimicked the size of /25 to /30 formulating subnet equivalents. That is how actual and observed values have been defined in the rest of the thesis. Let *Z(t)* and *Z^(t)* denote the actual and subnet equivalent sample values at data point t respectively (Kim and Kim, 2016). Smaller values of MAPE indicate a better fit, i.e. the smaller the mean absolute percentage error, the closer you are to finding the line of best fit (de Myttenaere *et al.*, 2016). Thus smaller values in our study are proof of how the subnet equivalent under study is closer to the baseline dataset (/24 IPv4 subnet). MAPE is defined in **Equation 3.1** as:

$$MAPE = \frac{1}{N}\sum_{t=1}^{N}\frac{|Z(t) - \hat{Z}(t)|}{Z(t)} \times 100 \tag{3.1}$$

3.9.2 Symmetric Mean Absolute Percentage Error (SMAPE)

Symmetric Mean Absolute Percentage Error (SMAPE) is an alternative to MAPE when there are zero or near-zero demand for items (Hyndman and Koehler, 2006). Let *Z(t)* and *Z^(t)* denote the actual and the subnet sample (be it equivalently sized to a /27, /28, or /29 subnet) values at data point t respectively (Kim and Kim, 2016). In contrast to the MAPE, SMAPE has both a lower bound and an upper bound. This symmetrical nature of SMAPE gives it a higher level of accuracy in its computational value than MAPE. SMAPE delimits to an error rate of 200% in order to reduce the influence of low volume items (Hyndman and Koehler, 2006). Low volume items are problematic because they could otherwise have infinitely high error rates that skew the overall error rate (Hyndman and Koehler, 2006). The interpretation is similar to that of MAPE since they are all percentage-based. In line with the study, SMAPE is the subnet samples of /24 IPv4 minus actual values divided by the sum of baseline value and subnet equivalent values as

expressed in **Equation 3.2**:

$$SMAPE = \frac{2}{N}\sum_{t=1}^{N}\frac{|Z(t) - \hat{Z}(t)|}{Z(t) + \hat{Z}(t)} \times 100 \tag{3.2}$$

3.9.3 Mean Absolute Error (MAE)

Mean Absolute Error (MAE) measures the average magnitude of the errors between actual observations and a set of predictions in which all values have equal weight (Varouchakis and Hristopulos, 2013). When measuring the magnitude of errors, MAE does not consider the direction of the set pairs under observation (Willmott and Matsuura, 2005). Like in Equation 3.2, *Z(t)* and *Z^(t)* denote the actual and subnet sample values at data point ***t*** respectively. MAE has negatively-oriented scores, meaning that lower values are better than higher values (Willmott and Matsuura, 2005). The smaller the mean absolute error, the closer one is to finding the line of best-fit (Wang and Bovik, 2009). Thus smaller values in our study are proof of how the subnet sample is closer to /24 IPv4 subnet. MAE is defined in **Equation 3.3** as:

$$MAE = \frac{1}{N}\sum_{t=1}^{N}|Z(t) - \hat{Z}(t)| \tag{3.3}$$

3.9.4 Mean Absolute Scaled Error (MASE)

Mean Absolute Scaled Error (MASE) is unlike the other two quantifying techniques, MASE uses a scaling error based technique instead of a relative measure (Hyndman and Koehler, 2006). MASE can only be computed when there are multiple time series to compute against each other (Hyndman and Koehler, 2006). MASE uses a scale based on the in-sample MAE as shown in Equation 3.3, which is independent of the scale of the data (Franses, 2016). The scale makes MASE less sensitive to outliers and easy to interpret and use in the same lines as MAPE or SMAPE (McKenzie, 2011). According to Hyndman and Koehler, the authors of the technique, a scaled error is less than one if it arises from a better forecast than the average one-step naive forecast computed *in-sample*. On the other hand, if the forecast is worse than the average one-step naive forecast computed in-sample, then it is greater than one. Bringing it into our context, if the sample drawn from /24 IPv4 subnet is better aligned to the original /24 IPv4, then the value will be less

than one. Also, if there are significant differences, then the value will be higher than one. Thus values of MASE that are less than one are ideal (Franses, 2016). In **Equation 3.5**, MAE in-sample, naive is the mean absolute error produced by a naive forecast (subnet sample). In this equation, $\boldsymbol{A_t}$ represents the true values of the baseline dataset at time $\boldsymbol{t}$ while $\boldsymbol{A_{t-1}}$ represents the predicted values of the baseline dataset at time $\boldsymbol{t-1}$. However, the explanation of how the equation has been used in this study is found in **Section 6.1.4**.

$$MAE_{in-sample,naive} = \frac{1}{N-1}\sum_{t=2}^{N}\left|\,(A_t) - (A_{t-1})\,\right| \tag{3.4}$$

$$MASE = \frac{MAE}{MAE_{in-sample,naive}} \tag{3.5}$$

Thus, using MASE, MAPE and SMAPE the study computed the error margin that exists between /24 IPv4 subnet and the subnet equivalents. Using the error margins, the study computed the level of accuracy that each subnet equivalent had in relation to the baseline dataset.

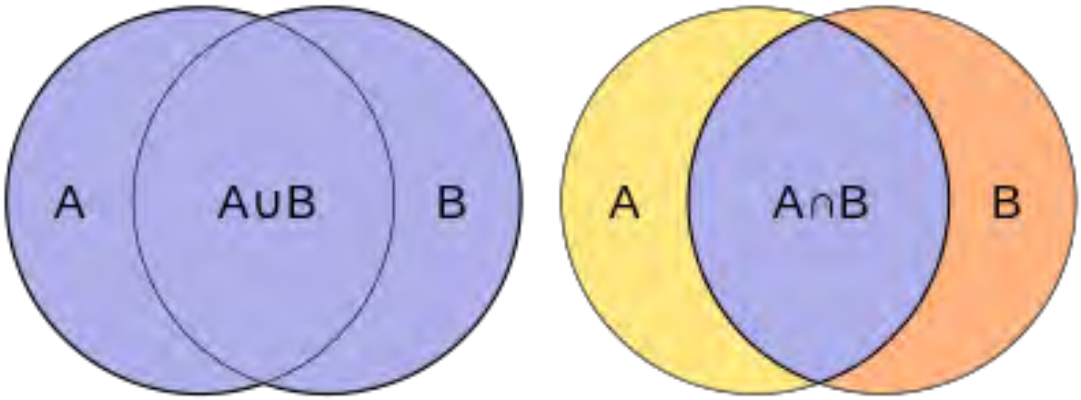
3.10 Information Retrieval and Text Mining Techniques

In line with the Information retrieval and text mining technique, this study used random sampling of /24 IPv4 net-block, with interest in the TCP destination ports (DPORTs) of the baseline dataset. The reason for the primary focus on TCP DPORTs has been explained in **Section 1.5** The aim was to identify which sample can best represent the baseline dataset. Initially, the techniques were developed to provide a statistical measure that evaluates how significant certain words are to a given document in collection or corpus (Schütze *et al.*, 2008). However, this study found a purpose for these techniques in quantifying the differences that are contained in the subnet samples when comparing against /24 IPv4 net-block.

When these techniques are presented in future sections, they will be presented in the context of this study and not text processing i.e. the terms used are in line with the content of the datasets at hand. This study will show how comparable the techniques are in terms of their scores. It will also show which subnet equivalent is comparable

to the baseline dataset. More importantly, it will prove that information retrieval and text mining techniques are a viable option to quantify subnet equivalent ports of an IBR dataset. The study proposed the use of information retrieval (IRT/IR) and text mining techniques to gauge the samples against the original dataset (Aizawa, 2003). Among such techniques, include Term Frequency (TF), Inverse Document Frequency (IDF), Term Frequency - Inverse Document Frequency (TF-IDF) and Jaccard Distance (JD) (Shameem and Ferdous, 2009). The results and application of all these techniques found in this section are presented in **Section 6.9**.

3.10.1 Jaccard Distance (JD)



(a) Union (b) Intersection

Figure 3.4: Union (∪) and intersection (∩) of set A and set B

Jaccard Distance (JD) quantifies how dissimilar two sets are, i.e. how is *Set A* different from *Set B* (Schütze *et al.*, 2008). The use of JD to measure the distance between two or more sets is at the core of many analyses such as clustering (Aggarwal, 2003), and time series (Parthasarathy, 2005). It is also often referred to as Intersection over Union because of the formula used (Schütze *et al.*, 2008). Basically, it is an intersection over a union of two sample sets understudy as shown in **Figure 3.4**.

$$JD(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|} \tag{3.6}$$

In this study, JD quantifies how dissimilar /24 IPv4 net-block dataset (represented by $\boldsymbol{A}$) is from subnet equivalents ($/_e25$ to $/_e30$) (represented by $\boldsymbol{B}$). Using JD this research will

compare the differences that exist between baseline datasets from the telescopes under study and their sub-samples ($/_e25$ to $/_e30$) (represented by $\boldsymbol{B}$). To be more specific, JD will quantify the differences in terms of DPORT count in the baseline and its sub-samples. A good interpretation of JD is presented by Parthasarathy (2005) where he stated that a JD of *zero* means that the baseline dataset is identical to the subnet equivalent while a JD of *one* means that there is no correlation between the baseline and the subnet equivalent. Using this interpretation, it can be inferred that sub-samples (which represent sets) which show a JD score closer to *zero* have DPORTs that are identical to those found in the baseline datasets. Those sub-samples that show a JD score closer to *one* implies that there is no correlation between the baseline and the sub-sample under study.

Equation 3.6 shows the mathematical representation of JD. It is important to note that the use of JD was inspired by how it had been applied in other fields. For instance, Parthasarathy (2005) showed in his work that JD was used to assess the clinical drugs for efficacy and hepatotoxicity of drugs where each patient provides a new dataset, and thus two patients assessed in that regard. Yuan *et al.* (2017) showed how JD has been used to automate skin lesion segmentation using Convolutional Neural Networks (CNN). This was done through the use of a novel loss function which was based on JD to eliminate the need for sample re-weighting.

JD has also found its applications in the field of machine learning as stated by Shameem and Ferdous (2009), where numerous cases have incorporated it for data partition, categorisation of items of a similar pattern with the K-Means algorithm. Mobile Ad Hoc Networks (MANETs) have also found a purpose for JD, where it is used to select dissimilar nodes during the discovery phase (Reina *et al.*, 2014). The purpose is to reduce the redundancy of routing packets during the discovery phase of the reactive routing protocols for MANETs (Reina *et al.*, 2014).

From the few examples cited this far, it is apparent that measuring the relative size of the overlap of two finite sets A and B has much use in practical applications. In this study, JD was used primarily when working with DPORTs to evaluate how the destination ports were affected by the sampling techniques used, be it sequential or random sampling. The research findings for JD have been presented in **Section 6.9**

3.10.2 Term Frequency (TF)

Term Frequency (TF) was designed to quantify how frequent a term appears within the document under study (Schütze *et al.*, 2008). It was designed in such a way that it

takes into account the length of the document since documents cannot all have the same length (Callan, 2002). This eliminates the bias of the ubiquity of certain terms in longer documents than shorter ones. However, in this study, documents have been equated to subnet equivalents. The terms of interest in our case were SRCIP and DPORTs for TCP and UDP traffic.

Term Frequency (TF) has been used to aid organisations with strategic decision making because of its ability to follow external rules (Santhanakumar *et al.*, 2018), i.e. it can be customised to meet user's needs. Like, Jaccard Distance, TF has found its use in machine learning algorithms (Schütze *et al.*, 2008). Among these include its use in the development of Support Vector Machines (SVM), identification of web crawlers, clustering and ranking of data and vector space classification (Schütze *et al.*, 2008). Term frequency has been compared with and used along the lines of some useful statistical models like Naive Bayes (Lewis, 1998). Search engines like Google have developed their algorithm around term frequency where items of interest are ranked based on their scores (Callan, 2002). Its ability to aid in decision making is viable proof that TF is reliable for the identification of variations that exist between subnet equivalents and /24 IPv4 net-block.

In this study, TF quantifies how frequent a DST port appears in a subnet equivalent. The weight is used to compare with the scores found in a baseline dataset.

In this study, **Equation 3.7** and the variables it has have been used as follows: $\boldsymbol{tf_{i,j}}$ represents the number of occurrences of port $\boldsymbol{i}$ in subnet/subnet equivalent $\boldsymbol{j}$, $\boldsymbol{n_{i,j}}$ represents the number of subnet equivalents containing port $\boldsymbol{i}$. $\boldsymbol{k}$ is the total number of documents being reviewed, and since this study is working with one subnet at a time, the value of $\boldsymbol{k}$ in this study will always be 1.

$$tf_{(i,j)} = \frac{n_{i,j}}{\sum_{k} n_{i,j}} \tag{3.7}$$

For example, if when analysing how many times Port 23/TCP appears in /25 subnet, then $\boldsymbol{k}$ is 1 meaning /25 subnet, $\boldsymbol{tf_{i,j}}$ represents the number of occurrences of Port 23/TCP in subnet 25. The same analysis will continue for other subnets as well to see how many times Port 23/TCP appeared in the subnet under study. If the frequency of the Port 23/TCP is identical in all subnets, then the scores will be the same, and if the frequency within the subnet is different, the scores will be different. In our case, the study was interested in all the ports registered, not just Port 23/TCP, thus the same principle will apply as

well when computing the scores. The research findings for TF have been presented in **Section 6.9**

3.10.3 Inverse Document Frequency (IDF)

Inverse Document Frequency (IDF) is a slight variation of TF in that while TF measures the frequency in which a term appears in the document, IDF takes into account the fact that not all terms are significant (Aizawa, 2003). This is to say that frequency of a variable is weighed more that its uniqueness. The rationale is that one can learn more from terms that are not frequent than those that are frequent. As such, there is a need to weigh down the frequent terms while at the same time scaling up rare terms. In short, it is interested in unique occurrences within a document. To a degree, IDF follows the law of diminishing returns.

Inverse Document Frequency (IDF) was introduced to strike a balance between the terms that were uncommon in documents and has since escalated to other fields such as image and language processing (Aizawa, 2003; Parthasarathy, 2005). This escalation has been attributed to its heuristic nature (Aizawa, 2003; Parthasarathy, 2005) i.e. its ability to adapt to practical application. Its application has extended to automatic term extraction in computational terminology and machine learning(Aizawa, 2003). In statistics, its application is seen in chi-squared tests (Smadja, 1993), log-likelihood ratio, and pairwise mutual information (Wiener *et al.*, 1995), among others.

In this study, Inverse Document Frequency (IDF) primarily focuses on the DST ports that are not common between the subnet equivalent. The rationale is that one can learn more from the DST ports that are not common in both the /24 IPv4 and the subnet equivalents under study than with the common ports. As such, the common DST ports are given a score of zero while those that are significant are weighted and a score is computed. A high score shows how dissimilar the subnet equivalent is from the baseline dataset.

In this study, **Equation 3.8** and the variables it has have been used as follows: $\boldsymbol{N}$ represents the total number of subnet equivalents under study. $\boldsymbol{df_i}$ represents the number of subnets containing $\boldsymbol{i}$.

$$idf = log\frac{N}{df_i} \tag{3.8}$$

For example, in this study, by focusing on those ports that are not common, more insights can be found in seeing how one subnet or subnet equivalent is different from the other. An uncommon port can be any TCP/UDP port that is not included in the common services ports category, i.e. other than the commonly used ports such as 80 (HTTP), 443 (HTTPS), 20/21 (FTP), 22 (SSH), 23 (Telnet), 3389 (RDP), 1521 (Oracle), 3306 (MySQL), 5432 (PostgreSQL), 53 (DNS), 1433 (MSSQL) and 137/138/139/445 (SMB/CIFS). If the frequency of the uncommon ports is identical in all subnets, then the scores will be the same, and if the frequency of the uncommon ports within the subnet is different, the scores will be different. This way it is possible to quantify how one subnet is different from the next. The research findings for IDF have been presented in **Section 6.9**

3.10.4 Term Frequency - Inverse Document Frequency (TF-IDF)

Lastly, the study reviewed Term Frequency-Inverse Document Frequency (TF-IDF) which is a merge of TF and IDF (Aizawa, 2003). TF-IDF takes into account the weight of frequent terms computed in TF and unique terms that are easily overlooked in TF (Aizawa, 2003). This combination ensures that term frequency and the uniqueness of terms between documents are taken into account when computing their difference (Chum *et al.*, 2008). In addition, TF-IDF improves probabilistic interpretation of weighting items in a document, which gives a better understanding of the statistical ranking mechanism (Hiemstra, 2000). Because of its hybrid nature, TF-IDF is a perfect representation of the best of TF and IDF as it takes into account the weight of frequent terms computed in TF and unique terms that are easily overlooked in TF (Aizawa, 2003). This ensures that the frequency of terms and the uniqueness of terms between documents are taken into account in computing their difference.

The formula for computing TF-IDF is shown in **Equation 3.9**. In this study, **Equation 3.9** and the variables have been used as follows: $\boldsymbol{N}$ represents the total number of subnet equivalent being evaluated, $\boldsymbol{df_i}$ represents the number of subnet equivalent containing port $\boldsymbol{i}$ and $\boldsymbol{tf_{i,j}}$ represents the number of occurrences of port $\boldsymbol{i}$ in subnet equivalent $\boldsymbol{j}$

$$tf.idf_{i,j} = tf_{i,j} * log(\frac{N}{df_i}) \tag{3.9}$$

Having verified that the information retrieval and text mining techniques herein can be applied to more than text and documents, this study will move to show how they were

used in weighting the differences that exist between /24 IPv4 net-block and its subnet equivalents using unique ports found in IBR datasets. The research findings for TF-IDF have been presented in **Section 6.9**

3.11 Chapter Summary

This chapter introduced and explained all the statistical techniques that will be used in **Chapters 5 and 6**. The main objective of this chapter was to show how most of these well-known statistical techniques were used by other researchers and explain how this research study used them to achieve its objectives.

The chapter started by discussing data sampling and why it is needed in this study in **Section 3.1**. This is followed by a discussion on Bootstrapping as a simulation tool in **Section 3.2**. This section covered two types of bootstrapping: parametric and non-parametric. Essentially, sampled data will have missing values and thus will not be exactly as the baseline data. However, with bootstrapping, we can simulate the samples to mimic the data points observed in the baseline data by using a statistical parameter of interest to reproduce the data. In our case, the *mean* was used as the statistical variable for bootstrapping the data samples. The study went further to look at the strengths and limitations of each bootstrap technique used in this study in **Section 3.3**. From here, the study reviewed the literature on Confidence Interval (CI) and the limitations that come with bootstrapping when using CI in **Section 3.4**. Applications of bootstrapping and confidence interval were discussed in **Section 3.5**. The study was also interested in knowing how the various variables in IBR data relate to each other over time. In order to achieve that, this study had to assess and identify if there is any relationship between the variables of interest by using regression analysis in **Section 3.6**. This section was immediately followed by **Section 3.7** which looked at the types of regression analysis that were considered for this study. The definition of concepts, and all important features needed to formulate mathematical models are explained in **Section 3.8**

Since the study aimed to observe how representative different samples of data collected were to the baseline dataset, there had to be a way of quantifying how such samples differ from the main dataset. This led the study to tools like MAPE, SMAPE, MAE and MASE to quantify such differences. This study also had to assess differences that exist between ports, particularly those in the DSTIP address blocks as these are the ones a network telescope user has control over. To do that, information retrieval techniques were

used to quantify these differences. These details are discussed in **Sections 3.9** and **3.10** respectively.

Chapters 5 and **6** will use these techniques to compute the results and derive answers needed from the datasets used. More referencing to this chapter from **Chapters 5** and **6** will be made in order to have a proper grasp of how these statistical techniques were used in this research study. Future sections that use this chapter have also been highlighted throughout its course.

4

Data Definition and Exploration

This chapter focuses on defining the data that was used for this research study. It also looks at an exploratory data analysis approach to the IBR data. The chapter explains the transformative process that raw *pcap* files have had to go through in order to be ready for use in the chapters that follow. The datasets explored in this chapter are the ones that the study will look at in **Chapters 5** and **6**

The chapter starts with **Section 4.1**, which discusses where the data was taken from and the datasets used in the study. This section goes further to explain the characteristics of each dataset using data dictionaries. The kind of sampling used and how the data was processed are also considered in this section. Following this is **Section 4.2** which adds more details on the characteristics of the data for the reader to better understand it. **Section 4.3** follows immediately and looks at the graphical representation of the datasets that were used. The Chapter concludes with a summary in **Section 4.4**. Note that exploratory work was conducted over a larger range of data (datasets from 2017 - 2021), however, for reporting purposes, most recent data dated 2021 was selected.

4.1 Data Sources

The Rhodes University network telescopes have been active for over fifteen years (Irwin, 2011), but this research will focus on most recent datasets to provide up to date activities happening in the network telescopes and based on current IBR trends. The data used in this research was obtained from Rhodes University's network telescopes (Irwin, 2011), collected from five network telescopes. This provided an opportunity to observe traffic from different network sensors. All the network telescope sensors are physically located in South Africa. Each of the five telescope sensors consisted of a /24 net-blocks, routed to a collection server. Note that there are a lot of similarities in the datasets that were collected in the Rhodes University telescopes as is shown in previous work (Irwin, 2013; Nkhumeleni, 2014). Thus in presenting the data, not all telescope datasets are going to be presented given that the results would be near identical given previous work done on them.

For this research, nine datasets have been selected from three different /24 IPv4 network telescopes. Each network telescope has contributed three months worth of data collected between 1 January 2021 to 31 March 2021 and discussed in **Section 4.1.1**. For security reasons, only the first octet of the destination IP addresses is displayed while the other three octets have been masked. Each dataset is named after the network telescope from which its data was collected. This is followed by the month in which the data was collected and, thereafter, the year in which this data was collected. To illustrate this naming convention, let us look at the first dataset: **146/8-012021**, this shows the reader the name of the network telescope (**146/8**) while hiding the last three octets of that network telescope followed by the month of January (*represented as 01*) and the year in which the data was collected (*in this case 2021*). For Network telescope **196/8**, there are three network telescopes running on **196/8** but the remaining three octets are different. Thus for labelling purposes, this study only picked one of the three and labelled it **196-A/8**. The datasets from it are named after this naming convention plus the month and year in which the data was collected. For example, dataset **196-A/8-012021** was collected in January. More on the naming convention used throughout this report can be found in **Section 4.1.1**. The naming convention used in **Section 4.1.1** is consistent with what was presented in **Section Section 1.6**.

4.1.1 Data Dictionaries for IBR Datasets

This section gives a breakdown of the datasets by showing the dataset composition using the data dictionaries for each dataset. A quick overview of each of the data dictionaries shows the time frame from which the data was collected, the unique name of the dataset, and the number of unique source (SRC) IP addresses that contributed to the traffic observed in each given month. The data dictionaries also show the number of unique destination (DST) IP addresses.

Each data dictionary contains dataset names and their attributes contributed by individual network telescopes. **Table 4.1** shows a breakdown of network traffic based protocol and the number of unique sources contained in each dataset. Note that these are the attributes contained in each of the data dictionaries presented in **Tables 4.1 - 4.2**. The total number of unique SRCIP addresses observed are later broken down to see under which protocol contributed more unique SRCIP addresses. It also shows the total traffic [1] observed in each month, which is later broken down by names of the protocols that each packet used during transmission from the source IP address to the targeted network telescope.

For this, the study had Transmission Control Protocol (TCP), User Datagram Protocol (UDP) and Internet Control Message Protocol (ICMP). Both the total traffic and the number of unique sources observed were represented using their proportions to see their representativeness in the overall dataset. From here onwards, the study will focus on the specific attributes and not give this description again.

The primary focus of the data dictionaries presented in this section is to identify the name of the network telescopes where the data was collected from, the number of packets from each of the major protocols identified in the dataset, the duration of data collection for the specific dataset, and the number of unique sources contained in each dataset and how proportional these unique sources are to the overall traffic observed. These details are seen in **Tables 4.1 - 4.3**. In all the datasets used, TCP packets made a major contribution to the traffic composition, followed by UDP and then ICMP. The other protocols identified were GRE and SCTP, but their proportion was negligible and as such they were added into the category of *other* to fit in the dictionaries.

[1] packets and traffic will be used interchangeably to mean the same thing

Table 4.1: Data Dictionary for Telescope 146/8

Dataset Name:	**146/8-012021**				
Start	Fri Jan 1 00:00:00 2021		**End**	Sun Jan 31 23:59:59 2021	
Duration	31 Days				
Total Traffic:	34,238,782	**Unique Sources:**	617,420	**Unique Destinations:**	256
Protocols	pkts	%	Sources	Uniq src %	
TCP	30,350,954	88.64	484,009	78.39	
UDP	3,594,676	10.50	124,900	20.23	
ICMP	293,062	0.85	8,421	1.36	
Other	90	0.01	90	0.01	
Dataset Name:	**146/8-022021**				
Start	Mon Feb 1 00:00:00 2021		**End**	Sun Feb 28 23:59:59 2021	
Duration	28 Days				
Total Traffic:	36,298,338	**Unique Sources:**	533,461	**Unique Destinations:**	256
Protocols	pkts	%	Sources	Uniq src %	
TCP	32,460,888	89.42	408,591	76.59	
UDP	3,578,212	9.85	116,718	21.88	
ICMP	259,146	0.71	8,060	1.51	
Other	92	0.02	92	0.02	
Dataset Name:	**146/8-032021**				
Start	Mon March 1 00:00:00 2021		**End**	Wed March 31 23:59:59 2021	
Duration	31 Days				
Total Traffic:	37,772,339	**Unique Sources:**	599,170	**Unique Destinations:**	256
Protocols	pkts	%	Sources	Uniq src %	
TCP	33,722,026	89.27	462,369	77.17	
UDP	3,743,119	9.90	128,071	21.37	
ICMP	307,005	0.81	8,541	1.43	
Other	189	0.02	189	0.03	

The data dictionary shown in **Table 4.1** presents data collected from **146/8** network telescope. This telescope recorded a total of **108,309,459** million events from *January 2021* to *March 2021*. These events are presented in form of the number of packets. These packets were sent from **1,750,052** unique SRCIP addresses that were sent to all 256 DSTIP addresses. In this data dictionary, there is no direct relationship between the total number of events observed within a month and the total number of unique SRCIP observed in the same month. This is observed in dataset **146/8-012021** which shows **617,420** unique SRCIP, the highest for network telescope **148/8**, yet shows the lowest number of events observed within this month (**34,238,782** packets).

Table 4.2: Data Dictionary for Telescope 155/8

Dataset Name:	**155/8-012021**				
Start	Fri Jan 1 00:00:00 2021		**End**	Sun Jan 31 23:59:59 2021	
Duration	31 Days				
Total Traffic:	39,489,736	**Unique Sources:**	637,014	**Unique Destinations**	256
Protocols	pkts	%	Sources	Uniq src %	
TCP	35,497,782	89.89	501,739	78.76	
UDP	3,769,639	9.54	126,730	19.89	
ICMP	222,209	0.56	8,439	1.32	
Other	106	0.01	106	0.02	
Dataset Name:	**155/8-022021**				
Start	Mon Feb 1 00:00:00 2018		**End**	Sun Feb 28 23:59:59 2021	
Duration	28 Days				
Total Traffic:	39,948,711	**Unique Sources:**	550,915	**Unique Destinations**	256
Protocols	pkts	%	Sources	Uniq src %	
TCP	36,031,250	90.19	424,861	77.12	
UDP	3,721,754	9.21	117,807	21.38	
ICMP	195,599	0.48	8,139	1.48	
Other	108	0.12	108	0.02	
Dataset Name:	**155/8-032021**				
Start	Mon March 1 00:00:00 2018		**End**	Wed March 31 23:59:59 2018	
Duration	31 Days				
Total Traffic:	43,043,116	**Unique Sources:**	644,182	**Unique Destinations**	256
Protocols	pkts	pkt %	Sources	Uniq src %	
TCP	37,753,171	87.71	484,030	75.13	
UDP	5,088,935	11.82	151,742	23.56	
ICMP	200,819	0.47	8,219	1.28	
Other	191	0.00[1]	191	0.03	

[1] rounded due to two decimal places [actual value = 0.0004].

The data dictionary shown in **Table 4.2** presents data collected from **155/8** network telescope. This telescope recorded a total of **122,482,563** million events in the first quarter of 2021. Apart from the high rise in packets, there is also an increase in the number of unique SRCIP addresses **1,832,111** that were sent to all 256 DSTIP addresses. A comparison between Network telescope **148/8** and **155/8** can be made here. In this data dictionary, there is no direct relationship between the total number of events observed within a month and the total number of unique SRCIP observed in the same month. The highest number of packets and unique SRCIP addresses are observed in dataset **155/8-032021** which shows **644,182** unique SRCIP. Dataset **155/8-032021** also shows the highest number of packets when compared to other months (**43,043,116** packets).

Table 4.3: Data Dictionary for Telescope 196-A/8

Dataset Name:	**196-A/8-012021**				
Start	Fri Jan 1 00:00:00 2021		**End**	Sun Jan 31 23:59:59 2021	
Duration		31 Days			
Total Traffic:	48,079,105	**Unique Sources:**	665,897	**Unique Destinations**	256
Protocols	pkts	%	Sources	Uniq src %	
TCP	42,043,059	87.44	538,891	80.93	
UDP	5,764,771	11.99	118,618	17.81	
ICMP	271,166	0.56	8,279	1.24	
Other	109	0.00*	109	0.016	
Dataset Name:	**196-A/8-022021**				
Start	Mon Feb 1 00:00:00 2021		**End**	Sun Feb 28 23:59:59 2021	
Duration		28 Days			
Total Traffic:	42,086,481	**Unique Sources:**	565,005	**Unique Destinations**	256
Protocols	pkts	%	Sources	Uniq src %	
TCP	36,046,727	85.65	444,282	78.63	
UDP	5,767,477	13.70	112,744	19.95	
ICMP	272,188	0.64	7,890	1.39	
Other	89	0.00[1]	89	0.03	
Dataset Name:	**196-A/8-032021**		**Description**		
Start	Thu Feb 1 00:00:00 2018		**End**	Wed March 31 23:59:59 2021	
Duration		31 Days			
Total Traffic:	49,866,136	**Unique Sources:**	618,660	**Unique Destinations**	256
Protocols	pkts	pkt %	Sources	Uniq src %	
TCP	33,679,719	67.54	488,379	78.94	
UDP	15,824,966	31.73	122,157	19.75	
ICMP	361,224	0.72	7,897	1.27	
Other	227	0.00[2]	227	0.04	

[1] rounded due to two decimal places [actual value = 0.0002].

[2] rounded due to two decimal places [actual value = 0.0004].

Table 4.3 shows data dictionary with datasets collected from **196-A/8** network telescope. This network telescope recorded a total of **140,031,692** million events from *January 2021* to *March 2021*. This is the highest number of packets recorded by any of the network telescopes under study. Of all network telescopes, **196-A/8** network telescope recorded the highest number of unique SRCIP addresses **1,849,562**. There is also no direct relationship between the total number of events observed within a month and the total number of unique SRCIP observed in the same month. Overall, February recorded the lowest number of events in all three network telescopes. This is true for both the number of unique SRCIP addresses and the total number of packets. On the other hand,

March showed the highest number of packets recorded by all three network telescopes.

Table 4.4: Total packets received per telescope

Network Telescope	Total No. of Packets
146/8	108,309,459
155/8	122,482,563
196-A/8	140,031,692

Table 4.4 is a summary table for total traffic received by each of the three network telescopes under study. From the table, it is apparent that during the same time frame, Network Telescope 196-A/8 recorded more packets than the other two. Details about the traffic breakdown by month have already been presented in **Tables 4.1 - 4.3**. A breakdown of this traffic by top 20 SRCIP and DPORTs is presented in **Section 4.2**.

Due to the high volume of TCP packets and the high number of unique sources observed per network telescope, much of this research study focused only on TCP packets than anything else. UDP datasets are mentioned in **Section 4.2** when evaluating the top 20 SRCIP, DPORTs and SRCIP addresses otherwise each time a dataset is mentioned, it refers to TCP dataset unless otherwise specified. UDP traffic was used for validation purposes only. However, the procedures that were carried out on TCP packets can be done on any of the protocols. The initial analysis worked with packets but eventually the focus shifted to working with unique SRCIP and DSTIP addresses.

4.1.2 Data Sampling

The two main sampling strategies used in this research study were random sampling and sequential sampling. For sequential sampling, an assessment was made to gauge how each subnet sample represented a baseline dataset. **Table I.1** found in **Appendix I** shows the proportions of each of sequential samples.

For random sampling, the samples were randomly drawn into smaller pools which were named *subnet equivalents* denoted as $/_{\text{e}}$ x, where x is the name of the subnet . These subnet equivalents equate the sizes of */25, /26, /27 /28, /29* and */30 subnets* which contain 128, 64 32, 16, 8 and 4 DSTIP addresses, respectively.

Table 4.5: Random Sampling Subnet Equivalents

Subnet Equivalent Name	Sample Draw Size
$/_e$ 25	128
$/_e$ 26	64
$/_e$ 27	32
$/_e$ 28	16
$/_e$ 29	8
$/_e$ 30	4

Table 4.5 shows the proportions of each of the subnet equivalent samples. These, throughout this report, will be referred to as *subnet equivalents* because the size of the samples is equivalent to that of the aforementioned subnets. This is to say that since a /27 subnet is expected to have 32 DSTIP addresses, then to come up with a $/_e$27 subnet equivalent dataset, 10 random draws were made from the /24 IPv4 addresses, where each draw contained 32 unique DSTIP addresses that were randomly sampled. 10 random draws per subnet size were made to create one subnet equivalent sample to ensure proper representation of the random sample. Thus an average of the 10 samples at each given data point within the sample were averaged to form one synthetic sample from which computations were made. Packets and SRCIP addresses are never in fractions or decimals. As such, the decimals from the averages were rounded off to whole numbers. This process was repeated for each subnet equivalent. Each subnet equivalent formulated contained, the date, source port, destination port, SRCIP, and DSTIP for those randomly sampled DSTIP addresses.

Note that the random IP addresses drawn were taken from the DSTIP addresses. For each DSTIP address sampled, the study looked at how many SRCIP addresses had been received within a given observation window. There was never a point where a drawn DSTIP address did not contain SRCIP addresses as the script validated this. This is to say that if a sample contained 16 DSTIP addresses, all 16 DSTIP addresses received traffic that contained SRCIP addresses.

For comparability purposes, each subnet equivalent dataset was normalised using an actual subnet size that matches with the subnet equivalent. For instance, to compare /24 IPv4 dataset (which contains 256 unique DSTIP addresses) with a *$/_e$27 subnet equivalent* dataset (which contains 32 randomly samples destination hosts), one would need to divide the traffic contained in the destination hosts by their respective subnet sizes.

This also ensured that calculations are done based on average observations per unique DSTIP address. This applied to all subnet equivalents used in this study. This random sampling approach to analysing network traffic is significant because future use of IBR collection methods is likely to be limited to smaller IP address pools, which may not be contiguous. Note that the use of the word ***subnet equivalent*** is because the number of DSTIP addresses contained in each of the sample draws its equivalent to the number of DSTIP addresses contained in an actual subnet of in IPv4 network. Note that random sample draws are not contiguous.

4.1.3 Data Processing

Pre-processing of the raw pcap data was performed using a combination of standard UNIX text processing tools and a series of Python scripts that were used to pre-process the pcap data and convert it to a *.csv* file format. The *.csv* format made the data readable and easier to work with when analysing, be it for statistical or visual purposes. All scripts related to data processing have been added to **Appendix D**. As explained in **Section 4.1.1**, TCP and UDP are the primary focus because they contributed in excess of 95% of the total traffic and are commonly used. The variables of interest for this research study were SRCIP addresses, DSTIP addresses, DPORTs and date of packet transmission. The date used the 24 hour binning. The date represents the day in which the data was collected and the 24 hour binning indicates the time stamp of each packet received.

SRCIP addresses were of interest because they helped to identify the distribution of traffic based on the area of origin. This would help in modelling the data as later on the study focused on the unique sources which give a more accurate distribution of traffic by source IP address. DSTIP addresses were selected because they are the ones the study had control over. More importantly, the key research question is to identify how many of the destination IP addresses one needs in order to collect data that offers a reasonable representation of the data collected in the baseline data. Destination ports were selected to identify which ports were frequently associated with high traffic. They were also used in modelling the problem at hand. The date of collection was selected because this enabled the researchers to identify the variation of traffic over time. In addition to this, it helps to run a comparative analysis among the different telescopes.

Having transformed the data format, the next step was to split the data into different protocols. For this study, more attention was given to TCP and UDP. This split made it easier to isolate traffic and thus add more accuracy to the results and computations made

from it. Since the overall objective was to identify the variations that happen between the baseline IBR dataset and its subnet equivalent, the data was then sampled using sequential and random sampling. This has been explained in detail in **Section 4.1.2**.

To check the number of SRCIP addresses received per day, 31 lists were created, each representing a day of the month under study. The use of a list ensured that if a unique SRCIP address appeared multiple times, its frequency would be considered unlike the use of sets that only focus on unique occurrences. The count (number of occurrences in a sample) for each SRCIP address was recorded and its output was saved to a file that contained days and frequency of SRCIP addresses being received by the DSTIP addresses. The frequency is what constituted the count of packets. More details on this are found in **Chapter 6** where both samples are applied to mathematical models to see their performance.

4.2 Descriptive Statistics for IBR Data

This section of the chapter provides more insight into the nature of the data that was used. The section is split into three subsections where **Section 4.2.1** explains the descriptive character of unique SRCIP addresses. **Section 4.2.2** focuses on the descriptive statistics on network ports using packets that each port registered as a key element. Lastly, **Section 4.2.3** focuses on the attributes of unique DSTIP addresses by focusing on the packet distribution from the unique SRCIP addresses.

4.2.1 Source IP addresses

Data is presented to show the breakdown of unique SRCIP addresses that were prevalent in all the datasets used. The tables in this subsection show the top 20 SRCIP addresses that sent more packets than other SRCIP addresses within the time frame of data collection. The SRCIP addresses in the tables are split based on the protocol that they used to transmit the packets to the DSTIP addresses. TCP and UDP are the two protocols that this study focused on, thus the tables will present SRCIP addresses for each month based on the protocol used. **Tables 4.7 - 4.9** shows top 20 SRCIP addresses ranked based on the amount of TCP packets (traffic) they transmitted while **Tables 4.10 - 4.12** shows top 20 SRCIP addresses ranked based on the amount of UDP packets (traffic) they transmitted. These SRCIP addresses were present during the period of data collection

for the 18 datasets. Of the 18 datasets, nine were TCP datasets while the other half were UDP datasets.

Each of the **Tables 4.7 - 4.12** shows seven columns in the following order: *Rank* which shows the position of the SRCIP in the order of top senders of traffic. **146/8** is the next column which shows SRCIP addresses that were collected by **146/8** network telescope. **155/8** column shows SRCIP addresses that were collected by **155/8** network telescope. **196-A/8** column shows SRCIP addresses that were collected by **196-A/8** network telescope. Each **%** column shows the proportion of traffic sent by each of the top SRCIP addresses shown in the tables. Each table has footnotes that show the total amount of traffic collected by each network telescope in a given month. This helps to put the proportions within context. Each table in this section also shows bold SRCIP addresses representing SRCIP addresses that are common in all three network telescopes for the given month. Detailed tables that show the actual number of packets by each of the SRCIP addresses can be found in **Appendix A**.

Table 4.6 shows a summary table of the % composition of the total sum of top 20 SRCIP. This is for both TCP and UDP datasets. The UDP datasets show that their top 20 SRCIP addresses received more traffic than the TCP datasets. Among the TCP datasets, February shows that the top 20 SRCIP received more traffic than any other month. The total sum of the top 20 SRCIP % composition were computed from **Tables 4.7 - 4.12**.

Table 4.6: % Sum of Top 20 SRCIP per Protocol

Dataset Name	Total TCP %	Total UDP %
146/8-012021	23.57	36.55
155/8-012021	21.10	32.76
196-A/8-012021	15.22	54.15
146/8-022021	37.43	39.66
155/8-022021	34.14	37.30
196-A/8-022021	31.30	57.74
146/8-032021	24.26	36.12
155/8-032021	22.10	36.65
196-A/8-032021	16.53	76.07

Table 4.7: Top 20 SRCIP Based on Volume of TCP Traffic [Jan 2021]

Rank	146/8[1]	%	155/8[2]	%	196-A/8[3]	%
1	**92.63.197.97**	4.26	**92.63.197.97**	3.64	**92.63.197.97**	3.07
2	**185.175.93.24**	3.65	**185.175.93.24**	3.26	**185.175.93.24**	2.75
3	**79.124.62.74**	2.38	**79.124.62.74**	2.03	**79.124.62.74**	1.72
4	**194.26.25.125**	2.27	**194.26.25.125**	1.94	**194.26.25.125**	1.64
5	194.147.140.41	1.29	194.147.140.41	1.12	**64.95.96.217**	0.62
6	194.147.140.42	0.88	178.33.221.97	0.76	194.147.140.8	0.60
7	194.147.140.6	0.84	194.147.140.42	0.75	**193.27.229.47**	0.54
8	45.129.33.128	0.78	194.147.140.6	0.71	**103.145.13.58**	0.53
9	**193.27.229.47**	0.74	45.129.33.128	0.67	**74.106.249.155**	0.52
10	**74.106.249.155**	0.73	205.220.231.26	0.67	**45.146.164.211**	0.46
11	**103.145.13.58**	0.71	**193.27.229.47**	0.63	**103.195.100.208**	0.46
12	45.129.33.47	0.71	**74.106.249.155**	0.63	**141.98.10.138**	0.44
13	**103.195.100.208**	0.64	45.129.33.47	0.59	194.26.25.13	0.32
14	**45.146.164.211**	0.63	**45.146.164.211**	0.55	89.248.160.178	0.32
15	**141.98.10.138**	0.61	**103.195.100.208**	0.55	45.146.165.171	0.31
16	122.228.19.79	0.57	**141.98.10.138**	0.52	93.174.93.123	0.29
17	**64.95.96.217**	0.55	122.228.19.79	0.49	103.145.13.43	0.28
18	45.146.165.171	0.44	**64.95.96.217**	0.47	205.220.231.26	0.27
19	89.248.160.178	0.44	**103.145.13.58**	0.44	161.189.114.127	0.27
20	194.26.25.13	0.43	205.220.231.25	0.43	38.130.221.107	0.26

[1] Total TCP traffic for 146/8-012021 was **30,350,954** packets

[2] Total TCP traffic for 155/8-012021 was **35,497,782** packets

[3] Total TCP traffic for 196-A/8-012021 was **42,043,059** packets

[*] SRCIPs in bold were present across all Datasets for January

Table 4.7 shows top SRCIP addresses that were registered by all three network telescopes in January 2021. SRCIP addresses **194.147.140.41** and **194.147.140.42** were persistent for all the three months but only present in **146/8** and **155/8** network telescopes for February and March. SRCIP **45.146.164.211** was present in all three network telescopes throughout the observation period. From this table, only **122.228.19.79** was found in the top 20 for both TCP and UDP traffic, however, it was not identified in network telescope **196-A/8**. Going Geolocation[2] on the dominant SRCIP shows that most of the top five are being used in Russia and in Bulgaria. For instance, **92.63.197.97** is being used by *LLC Digital Network* in Russia, **185.175.93.24** is being used in Russia by *Chistyakov Mihail Viktorovich.* On the other hand, **79.124.62.74** is being used by *Dm Auto Eood* in Bulgaria. All these were present in all network telescopes.

[2]https://www.findip-address.com

Table 4.8: Top 20 SRCIP Address Based on Volume of TCP Traffic [Feb 2021]

Rank	146/8[1]	%	155/8[2]	%	196-A/8[3]	%
1	**89.248.165.101**	17.50	**89.248.165.101**	15.72	**89.248.165.101**	15.71
2	**79.124.62.74**	5.73	**79.124.62.74**	5.16	**79.124.62.74**	5.16
3	**79.124.62.234**	4.13	**79.124.62.234**	3.73	**79.124.62.234**	3.73
4	194.147.140.41	1.24	194.147.140.41	1.13	**89.190.156.53**	1.42
5	**89.190.156.53**	0.94	**89.190.156.53**	0.84	45.79.121.175	0.93
6	194.147.140.42	0.75	194.147.140.42	0.69	**74.106.249.155**	0.50
7	194.147.140.68	0.66	205.220.231.26	0.56	**45.146.164.211**	0.49
8	194.147.140.66	0.61	178.33.221.97	0.56	**89.190.156.52**	0.44
9	194.147.140.70	0.56	194.147.140.68	0.55	89.248.160.178	0.35
10	**74.106.249.155**	0.55	194.147.140.66	0.54	94.232.46.244	0.35
11	194.147.140.40	0.55	**74.106.249.155**	0.50	89.248.165.104	0.31
12	**45.146.164.211**	0.54	194.147.140.70	0.49	93.174.93.123	0.31
13	194.147.140.69	0.51	194.147.140.40	0.49	103.145.13.58	0.29
14	194.147.140.26	0.50	**45.146.164.211**	0.49	89.248.165.53	0.29
15	122.228.19.79	0.49	194.147.140.69	0.49	205.220.231.26	0.28
16	**89.190.156.52**	0.49	194.147.140.26	0.45	89.248.165.51	0.28
17	194.147.140.67	0.47	194.147.140.96	0.45	194.61.25.194	0.27
18	194.147.140.96	0.44	122.228.19.79	0.45	103.145.13.43	0.27
19	89.248.160.178	0.39	**89.190.156.52**	0.45	89.248.165.93	0.26
20	94.232.46.244	0.38	194.147.140.67	0.40	45.125.65.105	0.25

[1] Total TCP traffic for 146/8-022021 was **32,460,888** packets

[2] Total TCP traffic for 155/8-022021 was **36,031,250** packets

[3] Total TCP traffic for 196-A/8-022021 was **36,046,727** packets

[*] SRCIPs in bold were present across all Datasets for February

Table 4.8 shows top SRCIP addresses that were registered by all three network telescopes in February 2021. The top 3 ports in **Table 4.8** transmitted at least 20% of the overall TCP traffic in the respective network telescopes. SRCIP address **89.248.165.101** being used by *IP Volume inc* in the United Kingdom transmitted the highest traffic in February in all three network telescopes. This is followed by **79.124.62.74** and **79.124.62.234** being used by *Dm Auto Eood* in Bulgaria. **79.124.62.74** was also observed in January. **89.190.156.53** which is associated with *Alsycon B.V* in the Netherlands was observed for both TCP and UDP traffic in the months of February and March. The number of identical SRCIP addresses among the network telescopes declined to seven from 11 observed in January. Just as in January, **122.228.19.79** (used by *China Telecom Wenzhou*) was observed in both TCP and UDP traffic in February. It is one of the persistent SRCIP observed in all datasets belonging to **146/8** and **155/8** network telescope.

Table 4.9: Top 20 SRCIP Based on Volume of TCP Traffic [Mar 2021]

Rank	146/8[1]	%	155/8[2]	%	196-A/8[3]	%
1	194.147.140.122	3.87	194.147.140.122	3.46	**45.93.201.188**	3.39
2	194.147.140.126	3.86	194.147.140.126	3.45	**82.102.137.130**	2.23
3	**45.93.201.188**	3.40	**45.93.201.188**	3.03	**193.27.229.207**	1.10
4	**82.102.137.130**	2.27	**82.102.137.130**	2.03	**193.27.229.47**	1.03
5	194.147.140.41	1.40	194.147.140.41	1.25	**89.190.156.52**	0.83
6	**193.27.229.207**	0.94	**193.27.229.207**	0.97	**69.25.114.212**	0.81
7	**193.27.229.47**	0.90	**193.27.229.47**	0.92	**45.155.205.155**	0.69
8	**89.190.156.52**	0.83	**89.190.156.52**	0.74	**89.190.156.53**	0.61
9	**69.25.114.212**	0.81	**69.25.114.212**	0.73	72.251.228.103	0.58
10	194.147.140.42	0.66	**45.155.205.155**	0.62	**89.248.165.101**	0.58
11	194.147.140.26	0.63	194.147.140.42	0.59	**45.146.164.211**	0.57
12	**45.155.205.155**	0.61	194.147.140.26	0.57	89.248.165.203	0.57
13	**89.248.165.101**	0.60	**89.248.165.101**	0.52	**45.146.165.24**	0.53
14	**89.190.156.53**	0.52	**45.146.164.211**	0.51	**103.99.2.190**	0.52
15	**103.99.2.190**	0.52	**103.99.2.190**	0.47	185.188.182.105	0.46
16	**45.146.165.24**	0.51	**45.146.165.24**	0.47	94.232.46.244	0.44
17	**45.146.164.211**	0.51	**89.190.156.53**	0.47	41.57.124.37	0.43
18	185.156.73.67	0.48	194.147.140.29	0.45	45.146.164.170	0.39
19	194.147.140.29	0.47	185.156.73.67	0.43	45.125.65.105	0.39
20	194.147.140.40	0.47	194.147.140.40	0.42	89.248.165.104	0.38

[1] Total TCP traffic for 146/8-032021 was **33,722,026** packets

[2] Total TCP traffic for 155/8-032021 was **37,753,171** packets

[3] Total TCP traffic for 196-A/8-032021 was **33,679,719** packets

[*] SRCIPs in bold were present across all Datasets for March

Table 4.9 shows top SRCIP addresses that were registered by all three network telescopes in March 2021. In March, the number of identical SRCIP addresses was the highest. However, the top three is not uniform when the observation is made in the months of January and February. The top SRCIP for **196-A/8** network telescope is different from the other two. Both **194.147.140.122** and **194.147.140.122** are associated with *Leading Mechanical Industry PJS* in Mongolia while **45.93.201.188** and **193.27.229.207** are associated with *OOO Network of data-centers Selectel* in Moscow, Russia. A persistent SRCIP **89.248.165.101** (belonging to IP Volume inc, United Kingdom) is observed in March as well. There is an introduction of new SRCIP in the top five and new countries. For instance, there is **82.102.137.130** which is associated with *Partner Communications* in Israel, and **194.147.140.41** (*IP Volume inc*), another persistent network, this time, in Iran.

Table 4.10: Top 20 SRCIP Based on Volume of UDP Traffic [Jan 2021]

Rank	146/8[1]	%	155/8[2]	%	196-A/8[3]	%
1	**146.88.240.4**	8.01	**146.88.240.4**	7.63	196.216.37.82	17.74
2	**95.214.52.175**	4.34	**95.214.52.175**	4.17	77.247.108.45	7.74
3	**95.214.53.145**	3.32	**95.214.53.145**	3.17	77.247.108.35	7.47
4	**69.162.117.142**	2.41	**69.162.117.142**	2.24	**146.88.240.4**	4.99
5	**95.214.54.95**	2.18	**95.214.54.95**	2.09	**95.214.52.175**	2.92
6	**193.29.14.109**	1.89	**104.243.40.37**	1.54	**95.214.53.145**	2.07
7	**104.243.40.37**	1.62	**185.94.111.1**	1.41	**95.214.54.95**	1.44
8	80.94.93.24	1.53	109.248.203.69	1.36	**69.162.117.142**	1.42
9	**185.94.111.1**	1.48	**95.214.54.161**	1.27	80.94.93.24	1.05
10	**95.214.54.161**	1.35	**45.125.65.52**	1.22	**104.243.40.37**	1.01
11	**45.125.65.52**	1.28	**193.29.14.109**	1.10	**185.94.111.1**	0.92
12	**80.94.93.16**	0.86	**80.94.93.16**	0.81	**45.125.65.52**	0.90
13	**80.82.65.90**	0.85	**80.82.65.90**	0.81	109.248.203.69	0.89
14	213.59.4.26	0.84	**80.94.93.10**	0.80	**95.214.54.161**	0.86
15	**80.94.93.10**	0.83	72.251.228.101	0.69	23.148.145.30	0.70
16	83.97.20.25	0.81	104.152.52.31	0.68	**193.29.14.109**	0.69
17	193.29.14.125	0.78	104.152.52.23	0.68	**80.94.93.10**	0.57
18	72.251.228.101	0.73	122.228.19.79	0.67	196.192.178.26	0.55
19	122.228.19.79	0.73	147.203.255.20	0.58	**80.82.65.90**	0.53
20	104.152.52.26	0.71	83.97.20.25	0.51	**80.94.93.16**	0.52

[1] Total UDP traffic for 146/8-01202 was **3,594,676** packets

[2] Total UDP traffic for 155/8-012021 was **3,769,639** packets

[3] Total UDP traffic for 196-A/8-012021 was **5,764,771** packets

[*] SRCIPs in bold were present across all Datasets for January

Table 4.10 shows top SRCIP addresses that were registered by all three network telescopes for UDP traffic observed in January. Although the total volume declined when dealing with UDP packets, the proportion of top SRCIP addresses contributing to the overall traffic increased in the three months collection period. A change in protocol came along with a lot of new SRCIP addresses. The top SRCIPs in January is dominated by persistent networks than SRCIP IP addresses. For instance, **95.214.52.175** and **95.214.53.145** associated with *Meverywhere sp. z o.o* in Poland. **77.247.108.45** and **77.247.108.35** is ABC Consultancy in Belize. **146.88.240.4** (associated with *Arbor Networks*) is the highest SRCIP address with most packets in UDP traffic, being present in all UDP datasets. There was also **193.29.14.109** (associated with *Bunea TELECOM SRL* in Romania) present in January and February for all network telescopes.

Table 4.11: Top 20 SRCIP Address Based on Volume of UDP Traffic [Feb 2021]

Rank	146/8[1]	%	155/8[2]	%	196-A/8[3]	%
1	**146.88.240.4**	7.38	**146.88.240.4**	7.09	196.216.37.82	33.32
2	**77.247.108.175**	7.16	**77.247.108.175**	6.84	**146.88.240.4**	4.57
3	**77.247.108.74**	6.72	**77.247.108.74**	6.49	**77.247.108.74**	4.32
4	**77.247.108.58**	3.39	**77.247.108.58**	3.29	**77.247.108.175**	4.23
5	103.145.13.60	1.69	213.59.4.26	1.58	**77.247.108.58**	2.05
6	**185.94.111.1**	1.36	**185.94.111.1**	1.31	103.145.13.60	1.05
7	**103.145.13.55**	1.27	**103.145.13.55**	1.22	**185.94.111.1**	0.85
8	**193.29.14.109**	1.21	**156.96.156.138**	1.14	**103.145.13.55**	0.79
9	**156.96.156.138**	1.19	**45.125.65.52**	0.91	**156.96.156.138**	0.74
10	103.145.13.59	0.95	**80.82.65.90**	0.76	**45.125.65.52**	0.63
11	**45.125.65.52**	0.95	**103.145.13.18**	0.74	38.91.100.237	0.60
12	**80.82.65.90**	0.80	**193.29.14.109**	0.72	103.145.13.59	0.59
13	**103.145.13.18**	0.77	**72.251.228.101**	0.70	217.182.199.129	0.56
14	**72.251.228.101**	0.73	104.152.52.32	0.69	193.46.255.20	0.52
15	104.152.52.28	0.72	104.152.52.24	0.69	95.214.53.145	0.52
16	104.152.52.18	0.72	122.228.19.79	0.66	**80.82.65.90**	0.50
17	122.228.19.79	0.70	193.29.14.112	0.63	**103.145.13.18**	0.48
18	193.29.14.127	0.66	193.107.216.17	0.62	**72.251.228.101**	0.45
19	193.29.14.112	0.66	89.40.70.237	0.61	**193.29.14.109**	0.45
20	89.40.70.237	0.63	217.182.199.129	0.61	104.152.52.34	0.45

[1] Total UDP traffic for 146/8-022021 was **3,578,212** packets

[2] Total UDP traffic for 155/8-022021 was **3,721,754** packets

[3] Total UDP traffic for 196-A/8-022021 was **5,767,477** packets

[*] SRCIPs in bold were present across all Datasets for February

In **Table 4.11**, there are five unique SRCIP addresses that are not appearing for the first time in the datasets i.e. persistent networks and SRCIP addresses. For instance **185.94.111.1** (associated with HLL LLC) in Russia was present in all telescopes in January and February but not present **in 196-A/8** for March, **45.125.65.52** (associated with Tele Asia Limited in Hong Kong) was present for January and February. **80.82.65.90** (associated with *Novogara LTD*), with geolocation in Seychelles, was present in January and February as well. **72.251.228.101**, associated with **Voxel Hosting**, located in the USA was present in all months, but not in all network telescopes i.e. it was not present for 196-A/8 in January and March. However, **146.88.240.4** (associated with *Arbor Networks*) is still the highest SRCIP address with most packets in UDP traffic in **146/8** and **155/8** network telescopes. For 196-A/8, **196.216.37.82** which belongs to *Paratus-Telecom* in Namibia recorded the highest proportion.

Table 4.12: Top 20 SRCIP Based on Volume of UDP Traffic [Mar 2021]

Rank	146/8[1]	%	155/8[2]	%	196-A/8[3]	%
1	**146.88.240.4**	7.90	**103.145.13.75**	6.40	107.148.161.86	27.15
2	103.145.13.131	3.39	**103.145.13.74**	5.90	196.216.37.82	14.85
3	**103.145.13.74**	2.54	**146.88.240.4**	5.81	103.248.20.30	7.04
4	**193.46.255.40**	2.47	103.145.13.131	2.49	23.27.103.158	5.58
5	**103.145.13.75**	2.14	**193.46.255.40**	1.92	103.248.20.21	2.71
6	**103.145.13.167**	2.03	**193.107.216.17**	1.46	23.27.103.157	2.65
7	**89.40.70.51**	1.84	**103.145.13.167**	1.44	77.247.108.45	2.39
8	**193.107.216.17**	1.83	**89.40.70.51**	1.35	**103.145.13.75**	2.14
9	**103.145.13.78**	1.77	103.145.13.147	1.33	77.247.108.35	2.05
10	185.94.111.1	1.35	45.143.221.110	1.30	**103.145.13.74**	1.99
11	193.29.14.125	1.15	185.94.111.1	0.99	**146.88.240.4**	1.86
12	**89.190.156.53**	1.12	193.29.14.125	0.85	45.121.107.128	1.41
13	103.145.13.69	1.08	**89.190.156.53**	0.81	103.145.13.130	0.80
14	92.204.135.183	0.93	92.204.135.183	0.69	**193.46.255.40**	0.62
15	209.222.98.168	0.83	**103.145.13.78**	0.62	45.143.221.110	0.55
16	81.177.143.31	0.80	209.222.98.168	0.61	**193.107.216.17**	0.51
17	89.248.165.164	0.77	89.248.165.164	0.56	**103.145.13.167**	0.48
18	103.145.13.77	0.77	72.251.228.101	0.51	**89.40.70.51**	0.44
19	193.46.254.182	0.71	104.152.52.30	0.50	103.145.13.147	0.43
20	72.251.228.101	0.70	104.152.52.26	0.50	**103.145.13.78**	0.42

[1] Total UDP traffic for 146/8-032021 was **3,743,119** packets

[2] Total UDP traffic for 155/8-032021 was **5,088,935** packets

[3] Total UDP traffic for 196-A/8-032021 was **15,824,966** packets

[*] SRCIPs in bold were present across all Datasets for March

Table 4.12 shows top SRCIP addresses that were registered by all three network telescopes for UDP traffic observed in March. **107.148.161.86** associated with *Cnservers LLC* in the USA, transmitted more UDP packets than any other SRCIP address (27.15%). Other than this, note also that in all UDP datasets, **196-A/8** recorded more packets than any other dataset, with the highest being recorded in March. This is observed by looking at the total traffic received by **196-A/8**, shown in **Table 4.12**. Note also that the top seven unique SRCIP addresses registered in **196-A/8** network telescope are unique to it and not present in **146/8** or **155/8**. This, in part, explains why there is a huge gap between the traffic recorded in **196-A/8** and that observed in **146/8** and **155/8** network telescopes. The top seven SRCIP addresses registered in **196-A/8** are unique to **196-A/8**, each of which contributed more traffic than the top SRCIP addresses **146.88.240.4** and **103.145.13.75** (another set of persistent networks already looked at from *Arbor*

Networks and *ABC Consultancy*). Actual number of packets for this table are shown in **Appendix A**, in **Table A.6**. 196.216.37.8, another persistent SRCIP belonging to *Paratus-Telecom* in Namibia. It was first observed in February for UDP traffic.

While there were persistent unique SRCIP addresses, the study observed that there were also persistent networks i.e. those networks that probed the network telescopes more than once. For instance, network **194.26.25.X** appeared twice in January alone. Like other unique SRCIPs, the individual SRCIP addresses in this network sent traffic of similar magnitude. The most dominant of all persistent networks is **194.147.140.X** which appeared at least twenty times within the three months observation period. The count is 'at least twenty times' because the study could not print out all 500,00 plus unique SRCIP addresses that were observed in each dataset. What is known is that for those unique SRCIP addresses that contributed the most, this network sent most of the TCP traffic received by the three network telescopes under study. It appeared more in February than in any other month.

Other persistent networks observed include **45.129.33.X**, which was observed in January and March, **74.106.249.X** which appeared in January and March as well. There was also **45.146.164.X** which appeared in all three months but using a single IP address each time it was observed in all network telescopes. Just after this persistent network, there was **45.146.165.X**, appearing twice again in January and March but using different IP addresses. The closeness and the period with which **45.146.164.X** and **45.146.165.X** appeared makes one believe that they could belong to the same organisation.

From the top 20 SRCIP addresses presented in this section, there are a few major points to take away. Firstly, the observation of unique SRCIPs that are persistent and appear in more than one network telescope. The presence of the unique SRCIP addresses in all of the network telescopes shows how similar the data collected from the three network telescopes is. Secondly, these top SRCIP addresses sent high volume of internet traffic throughout the observation period and it is one of the main reasons why traffic patterns between the network telescopes are similar. These top 20 SRCIP addresses broadcasted their traffic in each of the network telescopes. There is also the observation that shows unique SRCIP addresses that appear in every month and send traffic both for TCP and UDP traffic.

Looking at the total sum of traffic contributed by the top 20 SRCIP as shown in **Table 4.6**, one would note that there is still a lot of traffic unaccounted. This is said because the overall % sum is below 40%. This is with the exception of 196-A/8 UDP traffic.

However, if the traffic sent by the top 20 SRCIP addresses is anything to go by, then it can be inferred that this behaviour of broadcasting traffic to all unique DSTIP addresses continues with all the other SRCIP addresses. There is a graphical representation of traffic distribution shown in **Section 4.2.3** that supports this. The knowledge collected from the geolocation of the unique SRCIP addresses also helps to understand where most of the probes on the network telescopes are coming from. This is working with the assumption that the traffic is originating from the identified SRCIPs. More specific details have been explained against each table to show the origin of these top SRCIPs.

4.2.2 Port Breakdown for TCP and UDP Traffic Dataset

Having looked at traffic distribution based on unique SRCIP addresses, the study focused on another important element of packet transmission, ports. Each SRCIP had to use at least one of the opened ports on the network telescopes in order to transmit their packets, be it TCP or UDP. **Tables 4.14 - 4.19** gives a summary breakdown of the destination ports (DPORT) that received a lot of traffic from the SRCIP addresses based on the two protocols under study (TCP and UDP). The labelling of the columns in the tables to represent specific network telescopes is the same as that shown in **Section 4.2.1** except this time, the column contains DPORT numbers that registered more traffic and the proportion of the traffic observed. **Table 4.13** shows the total percentage sum of the top 20 DPORTs. Tables containing the actual number of packets received by each DPORT with their proportions can be found in **Appendix B**.

Table 4.13: % Sum of Top 20 DPORT per Protocol

Dataset Name	Total TCP %	Total UDP %
146/8-012021	23.40	47.75
155/8-012021	32.17	44.86
196-A/8-012021	47.52	50.70
146/8-022021	18.73	48.14
155/8-022021	25.81	45.04
196-A/8-022021	32.57	65.15
146/8-032021	23.37	52.43
155/8-032021	31.52	39.33
196-A/8-032021	39.80	76.20

Table 4.14: Top 20 DPORT Based on Volume of TCP Traffic [Jan 2021]

Rank	146/8[1]	%	155/8[2]	%	196-A/8[3]	%
1	23	6.95	445	10.01	37215	19.09
2	22	2.63	23	6.04	445	9.62
3	80	2.09	1433	2.56	23	5.17
4	445	1.78	22	2.06	22	2.46
5	443	1.32	80	1.72	1433	2.38
6	8080	1.24	8080	1.07	80	1.45
7	3389	1.14	443	1.04	8080	1.12
8	81	1.05	3389	1.00	443	0.92
9	6379	0.79	81	0.91	3389	0.81
10	5555	0.69	6379	0.74	81	0.76
11	5038	0.53	10530	0.60	6379	0.58
12	8545	0.50	33529	0.60	5555	0.49
13	1433	0.47	12111	0.60	34694	0.47
14	50802	0.42	61380	0.59	5038	0.40
15	8081	0.40	5555	0.58	8545	0.36
16	8443	0.36	16979	0.56	50802	0.31
17	11211	0.36	8545	0.43	8728	0.30
18	2323	0.35	5038	0.40	8081	0.30
19	3306	0.35	8081	0.35	8443	0.27
20	139	0.34	11211	0.31	11211	0.26

[1] Total TCP traffic for 146/8-012021 was **30,350,954** packets
[2] Total TCP traffic for 155/8-012021 was **35,497,782** packets
[3] Total TCP traffic for 196-A/8-012021 was **42,043,059** packets

Table 4.14 shows the top 20 DPORTs that received the most TCP traffic in the month of January. There is no single DPORT that was dominant in all three network telescopes. Of more interest is port **37215/TCP** which received more traffic than any other port. This is confirmed by looking at the total traffic in the 196-A/8 network telescope. Port **37215/TCP** is used by Huawei Technologies to run Huawei HG532 routers. According to **CVE-2017-17215**[3], Huawei HG532 with some customized versions has a remote code execution vulnerability. An authenticated attacker could send malicious packets to port 37215/TCP to launch attacks. A successful exploit could lead to the remote execution of arbitrary code. According to Port Attack Activity[4], Port **37215/TCP** was scanned the most in January, thus this table confirms such a scan activity. More details on services running on the top DPORTs found in this study is in **Appendix C**

[3]https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-17215
[4]https://isc.sans.edu/port.html?port=37215

Table 4.15: Top 20 DPORT Based on Volume of TCP Traffic [Feb 2021]

Rank	**146/8[1]**	**%**	**155/8[2]**	**%**	**196-A/8[3]**	**%**
1	23	5.62	445	8.50	445	9.72
2	22	1.82	23	5.15	23	5.05
3	80	1.57	1433	2.17	37215	4.69
4	445	1.34	22	1.52	1433	2.26
5	8080	0.98	80	1.39	22	2.12
6	3389	0.96	8080	0.90	80	1.41
7	5555	0.96	3389	0.87	8080	1.12
8	443	0.90	443	0.81	3389	0.86
9	6379	0.65	6379	0.67	443	0.81
10	81	0.60	5555	0.65	5555	0.70
11	5038	0.47	81	0.54	6379	0.60
12	8081	0.39	8081	0.35	81	0.54
13	1433	0.35	8888	0.30	5038	0.43
14	3306	0.33	3306	0.30	8291	0.38
15	8888	0.33	11211	0.29	8728	0.36
16	11211	0.32	12111	0.28	8081	0.36
17	26	0.30	61380	0.28	8888	0.30
18	2323	0.29	10530	0.28	3306	0.29
19	8443	0.28	16979	0.28	11211	0.29
20	50802	0.27	33529	0.28	34694	0.28

[1] Total TCP traffic for 146/8-022021 was **32,460,888** packets
[2] Total TCP traffic for 155/8-022021 was **36,031,250** packets
[3] Total TCP traffic for 196-A/8-022021 was **36,046,727** pkts

Table 4.15 shows the top 20 DPORTs that received the most TCP traffic in the month of February. The order of the top three recipients in **196-A/8** network telescope has changed while for **146/8** and **155/8** telescopes remained the same. The change in position of Port **37215/TCP** to third conforms to the decline in probes for this port[5] which had a huge spike in January alone. The decline in the volume of traffic for Port **37215/TCP** coincides with the huge drop of total traffic for **196-A/8** telescope (see the totals). This massive drop in total traffic from **42,043,059** in January to **36,046,727** in February is not reflected in **146/8** and **155/8** telescopes. It was expected that Port **23/TCP** (Telnet, used for accessing systems remotely) and port **445/TCP** (used by Microsoft Directory Services for Active Directory (AD) and for the Server Message Block (SMB) protocol) would register more traffic because of the nature of the services they run. Overall, new ports have emerged into the top 20 while others have dropped rank, e.g Port **139/TCP**.

[5] https://isc.sans.edu/port.html?port=37215

Table 4.16: Top 20 DPORT Based on Volume of TCP Traffic [Mar 2021]

Rank	**146/8[1]**	**%**	**155/8[2]**	**%**	**196-A/8[3]**	**%**
1	23	7.10	445	10.35	445	13.27
2	22	2.22	23	6.32	23	6.89
3	80	1.77	1433	2.57	1433	2.95
4	445	1.59	22	1.87	22	2.69
5	6379	1.44	6379	1.60	37215	2.08
6	5555	1.40	80	1.56	80	1.79
7	443	1.19	443	1.05	6379	1.50
8	3389	1.09	3389	0.98	8080	1.27
9	8080	1.02	8080	0.90	443	1.21
10	81	0.76	5555	0.75	3389	1.05
11	26	0.49	81	0.67	5555	0.81
12	1433	0.42	26	0.43	81	0.75
13	8291	0.42	8081	0.35	8291	0.63
14	8081	0.39	8443	0.33	8728	0.52
15	8443	0.37	8291	0.32	26	0.48
16	5038	0.36	5900	0.32	2375	0.42
17	5900	0.35	2323	0.31	8081	0.40
18	2323	0.34	8545	0.30	9090	0.37
19	8545	0.34	8000	0.28	8443	0.36
20	8000	0.31	9999	0.26	5038	0.36

[1] Total TCP traffic for 146/8-032021 was **33,722,026** packets
[2] Total TCP traffic for 155/8-032021 was **37,753,171** packets
[3] Total TCP traffic for 196-A/8-032021 was **33,679,719** pkts

Table 4.16 shows the top 20 DPORTs that received the most TCP traffic in March. A port of interest in this table is **6379/TCP** which is used to run *Remote Dictionary Server* (redis[6]). Port **6379/TCP** moved up the ranks in March in all network telescopes. Usually, this port was found around positions 9, 10, or 11 in January and February. However, in March, it was found in positions 5 (in **146/8** and **155/8**) and 7 (in **196-A/8**). According to Speed guide.net[7], Port **6379/TCP** received more daily hits ranging between 2,500 - 3,600 per day than any other month in 2021. After March the numbers lowered to below 2,500 hits per day. It is therefore no surprise that this rise in daily hits is also detected in all the network telescopes under study, showing how the data in the network telescopes is in sync with activities occurring in the allocated address blocks. Port **37215/TCP** has further dropped on the rank and is only present in **196-A/8**.

[6] https://redis.io
[7] https://www.speedguide.net/port.php?port=6379

Table 4.17: Top 20 DPORT Based on Volume of UDP Traffic [Jan 2021]

Rank	**146/8[1]**	**%**	**155/8[2]**	**%**	**196-A/8[2]**	**%**
1	5060	14.35	5060	12.58	53	20.08
2	123	6.07	123	5.55	5060	10.37
3	53	3.79	53	3.65	123	4.75
4	1900	3.26	1900	3.15	161	2.33
5	161	2.87	161	2.74	1900	2.03
6	389	2.15	389	2.04	389	1.57
7	1434	1.62	11211	1.65	1434	1.01
8	11211	1.50	1434	1.54	11211	0.93
9	5353	1.39	5353	1.34	137	0.89
10	137	1.34	137	1.27	5353	0.86
11	5683	1.22	5683	1.17	5683	0.77
12	111	1.11	111	1.05	111	0.70
13	1194	1.07	1194	1.03	1194	0.67
14	6881	1.03	6881	0.97	6881	0.64
15	3283	0.88	33434	0.92	19	0.60
16	19	0.88	3283	0.87	3283	0.55
17	6536	0.84	33435	0.86	5070	0.54
18	5070	0.82	33441	0.83	5632	0.49
19	5632	0.79	19	0.83	5351	0.48
20	5351	0.77	33440	0.82	1027	0.44

[1] Total UDP traffic for 146/8-012021 was **3,594,676** packets

[2] Total UDP traffic for 155/8-012021 was **3,769,639** packets

[3] Total UDP traffic for 196-A/8-012021 was **5,764,771** packets

Table 4.17 shows the top 20 DPORTs that received the most UDP traffic in the month of January. With the change in protocol, the study observed a new set of ports that received more traffic in correspondence to UDP traffic. Ports **5060/UDP**, **53/UDP** and **123/UDP** sent most of the traffic in March, at least 20% of the total UDP traffic was transmitted via these ports in each network telescope. Port **5060/UDP** is used for Session Initiation Protocol (SIP) communication by signaling and controlling interactive communication sessions[8] while **53/UDP** is used for Domain Name System[9] (DNS) and **123/UDP** services Network Time Protocol[10] (NTP).

[8]https://www.speedguide.net/port.php?port=5060

[9]https://www.speedguide.net/port.php?port=53

[10]https://www.speedguide.net/port.php?port=123

Table 4.18: Top 20 DPORT Based on Volume of UDP Traffic [Feb 2021]

Rank	**146/8[1]**	**%**	**155/8[2]**	**%**	**196-A/8[3]**	**%**
1	5060	14.85	5060	13.12	53	35.64
2	123	7.46	123	6.59	5060	10.82
3	53	3.72	53	3.58	123	4.57
4	1900	3.15	1900	3.02	1900	1.97
5	389	2.72	389	2.61	161	1.83
6	161	2.46	161	2.36	389	1.56
7	1434	1.25	11211	1.26	137	0.83
8	137	1.20	1434	1.20	11211	0.82
9	11211	1.15	137	1.15	1434	0.78
10	5683	1.13	54047	1.09	5683	0.70
11	5353	1.08	5683	1.09	5353	0.67
12	5070	1.01	5353	1.05	3283	0.65
13	69	1.00	5070	0.96	5070	0.65
14	111	0.99	69	0.96	69	0.61
15	3702	0.96	111	0.95	111	0.61
16	1194	0.95	1194	0.91	1194	0.58
17	19	0.79	6576	0.81	19	0.49
18	3283	0.78	19	0.78	6881	0.47
19	6881	0.77	3283	0.78	3702	0.46
20	5632	0.72	6532	0.77	5632	0.44

[1] Total UDP traffic for 146/8-022021 was **3,578,212** packets

[2] Total UDP traffic for 155/8-022021 was **3,721,754** packets

[3] Total UDP traffic for 196-A/8-022021 was **5,767,477** packets

Table 4.18 shows the top 20 DPORTs that received the most UDP traffic in the month of February. As in January, Ports **5060/UDP**, **53/UDP** and **123/UDP** are the dominant ports as well. However, noteworthy is how the volume of traffic transmitted through these ports (Ports **5060/UDP**, **53/UDP** and **123/UDP**) has increased, especially in **196-A/8** telescope with half of the total traffic in **196-A/8** telescope. Another port of interest observed throughout the data collection period is Port **19/UDP** which services Character Generator Protocol[11] (CHARGEN). It should be disabled if there is no specific need for it, as it may be a source for potential attacks [RFC 864]. According to speedguide, Port **19/UDP** was averaging 850 hits per day between January and March in networks that are fully operational.

[11] https://www.speedguide.net/port.php?port=19

Table 4.19: Top 20 DPORT Based on Volume of UDP Traffic [Mar 2021]

Rank	**146/8[1]**	**%**	**155/8[2]**	**%**	**196-A/8[3]**	**%**
1	5060	16.77	5060	11.71	123	52.21
2	123	8.77	123	5.78	53	15.83
3	53	4.11	53	3.04	5060	5.09
4	389	3.73	389	2.80	389	0.92
5	1900	2.67	1900	2.09	161	0.91
6	161	2.45	161	1.81	1900	0.63
7	3702	1.40	49693	1.09	3283	0.60
8	1434	1.33	11211	0.98	137	0.33
9	137	1.26	1434	0.96	11211	0.32
10	5683	1.21	137	0.91	1434	0.31
11	5353	1.18	5683	0.90	3702	0.30
12	11211	1.08	25631	0.88	5683	0.29
13	1194	1.06	5353	0.87	5353	0.28
14	111	0.83	11551	0.84	1194	0.25
15	6881	0.82	44060	0.78	5070	0.20
16	6572	0.80	44830	0.78	111	0.19
17	19	0.78	1194	0.78	6881	0.19
18	5070	0.76	9757	0.78	19	0.19
19	3283	0.75	28447	0.78	5080	0.16
20	17	0.67	62495	0.77	5632	0.16

[1] Total UDP traffic for 146/8-032021 was **3,743,119** packets

[2] Total UDP traffic for 155/8-032021 was **5,088,935** packets

[3] Total UDP traffic for 196-A/8-032021 was **15,824,966** packets

Table 4.19 shows the top 20 DPORTs that received the most UDP traffic in the month of February. The ranking of the top 3 ports in **146/8** and **155/8** telescopes has not changed. However, in **196-A/8** the ranking has changed with Port **123/UDP** at the top with over 50% of the total traffic for **196-A/8** being transmitted through it. This spike is shown in **Section 4.3** where a single day in March recorded over 7,200,000 packets. It is no surprise that the total traffic for **196-A/8** almost tripled the amount received in January or February. Another port that has been persistent is Port **389/UDP** which services Lightweight Directory Access Protocol[12] (LDAP)

Due to the high number of unique DPORTs that registered traffic, the proportionality of the unique DPORTs in terms of how much traffic they registered as compared to the overall 65,535 ports was very small. This is with the exception of traffic registered

[12]https://www.speedguide.net/port.php?port=389

on UDP traffic for Ports **53/UDP** and **123/UDP**. In subsequent chapters, the study will show that each of the unique SRCIP IP addresses probed at least one of the open ports. This looks insignificant, but when data sampling (sequential or random) is done, it was observed that each sample had about 96% of the total number of unique ports that registered traffic in that sample. Now, this becomes significant because it essentially means that our sampling has very little negative effects on the overall sampling.

Of note is that traffic was sent to all **65,535** unique ports for TCP datasets, however, this was not consistent in all datasets. For instance, for the month of January, none of the TCP datasets received traffic in all **65,535** unique ports. The highest volume of TCP traffic was observed in **155/8** telescope which had **65,518** of its unique ports registering traffic while the lowest count of ports to have received TCP traffic was observed in **146/8** telescope which brings its total count of unique ports to **65,502**. The number of unique DPORTs that registered started to increase in February and by March all unique destination ports had registered traffic. The numbers were significantly lower for all UDP datasets with the highest number of unique DPORTs coming to **17,794** for **155/8** and its lowest count found in the same network telescope but different dataset in February.

The major take away from the top 20 is to note which of the services are targeted the most. These ports ought to be heavily guarded with their vulnerabilities patched. The services that match the top ports are presented in **Appendix C**. Some of the these ports are still exploited even with old vulnerabilities like Port **37215/TCP**. Another major take away is to note how unusual amount of traffic in the assigned IP addresses can be picked in the network telescope. This has been presented in **Table 4.14** which is coupled with an explanation. With network telescope scans, it is easy to identify when abnormal traffic (indicating a threat) is registered in a network. Certain ports are expected to receive high volume of traffic, like Port **445/TCP** for example. Thus, if one notes that this order has been disrupted, as it is in **Table 4.14**, it is indicative of an anomaly in traffic. If one does not know the order in which these ports receive traffic it would be difficult to identify early stages of an attack. Seeing how similar the ranking of the top 20 DPORTs and their % composition helps us to understand the scores presented in **Section 6.9**.

4.2.3 Destination IP Address

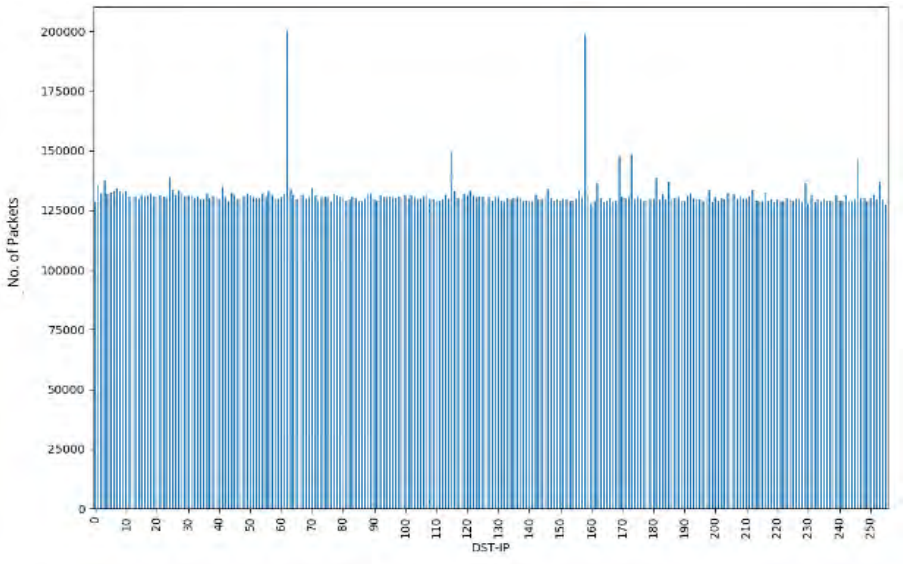


Figure 4.1: 146/8-032021: No. of Packets per DSTIP

Having looked at the unique SRCIP addresses and DPORTs, the study shifted its attention to look at one critical variable, i.e. DSTIP addresses. The approach used for describing the DSTIP addresses was different from the one used for unique SRCIP addresses and DPORTs. **Figure 4.1** is a sample graphical representation of how TCP packets were distributed per given unique DSTIP. This section shows how traffic was equally distributed (equitable distribution) across the unique DSTIP addresses for each network telescope. There were a few outliers in each dataset shown by the spikes observed in **Figure 4.1**, although they were not exactly in the same positions. What is essential is that a majority of the unique DSTIP addresses received uniform traffic throughout the data collection period. This is now where **Tables 4.21** comes into play to give a statistical value of how this packet distribution occurred in all nine datasets.

The values for consideration in descriptive statistics were *mean, median, standard deviation (std)*, the minimum number of packets registered by a unique DSTIP (represented as *min*) and the maximum number of packets received by a unique DSTIP (represented by *max*). All these are well known statistical terms that are not new to this research, however, they are used here to give more meaning to the data by describing the characteristics of the data used.

Table 4.20: Unique SRCIP monitoring based on DSTIP

Dataset Name:	**146/8-012021**			
Label	**IP Range**	**No. of SRCIP**	**SRCIP(%)**	**Packets(%)**
A	**0 - 63**	158,862	25.73	24.43
B	**64 - 127**	154,725	25.06	24.99
C	**128 - 191**	149,230	24.17	25.13
D	**192 - 255**	148,427	24.04	24.44
Total SRCIP Count:	**617,420**			
Dataset Name:	**155/8-022021**			
Label	**IP Range**	**No. of SRCIP**	**SRCIP(%)**	**Packets(%)**
A	**0 - 63**	139,656	25.35	24.93
B	**64 - 127**	141,364	25.66	25.03
C	**128 - 191**	135,580	24.61	25.56
D	**192 - 255**	134,313	24.38	24.51
Total SRCIP Count:	**550,915**			
Dataset Name:	**196-A/8-032021**			
Label	**IP Ranges**	**No. of SRCIP**	**SRCIP(%)**	**Packets(%)**
A	**0 - 63**	154,850	25.03	24.80
B	**64 - 127**	159,985	25.86	24.75
C	**128 - 191**	151,386	24.47	25.59
D	**192 - 255**	152,437	24.64	24.86
Total SRCIP Count:	**618,660**			

An exploratory analysis was conducted to observe how the unique SRCIP addresses distributed their packets to the unique DSTIP addresses. **Table 4.20** shows how many unique SRCIP addresses were registered within the given DSTIP range (which represents a subnet) and how many packets each DSTIP range received from the unique SRCIP addresses registered in the three datasets. Note that the datasets are from the three network telescopes and as can be seen from the naming, data was collected from January to March. Each network telescope represents a different month. The labels are the same because irrespective of them being from different datasets, they represent the same range.

Table 4.20 also shows proportion of both the DSTIP addresses and the packets in relation to the subnet that registered them. There are no major clear cut differences as to which subnet received more traffic or unique SRCIPs than the other. The variation of both the SRCIP addresses and packets fall under the range of 1%. From the exploratory analysis, it was noticed that the unique SRCIP addresses did not send traffic to all DSTIP addresses. If this was true then there would have been more overlaps and equal number of unique SRCIP addresses present in the subnets. This brought the conclusion that the SRCIP

addresses are not present in all DSTIP addresses. By this logic, it also means that no single subnet mask contained all unique SRCIP addresses as seen in **Table 4.20**. This means that any sequential sampling is most likely to result in similar findings as the unique SRCIP addresses are evenly distributed across the subnets. This observation may also mean that no matter how the data is sampled (systematic, sequential or random sampling) the variations may not be that significant, at least with these datasets). This is most likely attributed to the fact that the unique SRCIP addresses had different observable periods and not uniform throughout the observation period. It is this equitable distribution of unique SRCIPs that propelled the decision to conduct sequential sampling by using one sample per subnet as presented in **Section 6.2**.

Table 4.21 shows the statistical terms used to describe each dataset, name of the network telescope used to collect the data, followed by a row that shows the month in which the data was collected. For each network telescope, three months worth of statistics belonging to it is presented.

Table 4.21: Descriptive Statistics for No. of TCP Packets Observed per DSTIP

Network Telescope:	**146/8**		
Statistic	**January**	**February**	**March**
mean	118,186	126,525	131,338
median	117,574	126,283	130,237
std	1,886	1,284	6,675
min	115,782	123,564	127,469
max	133,626	131,452	200,282
Network Telescope:	**155/8**		
Statistic	**January**	**February**	**March**
mean	138,365	140,430	147,037
median	133,300	138,011	146,190
std	29,089	142,85	3,119
min	129,024	136,291	142,323
max	350,474	246,733	175,007
Network Telescope:	**196-A/8**		
Statistic	**January**	**February**	**March**
mean	163,916	140,497	131,135
median	162,435	139,799	129,605
std	12,561	64,97	8,127
min	158,107	136,261	127,408
max	360,859	241 661	202,566

For instance, the first block of statistics shown in **Table 4.21** gives descriptive statistics for TCP datasets that were collected from **146/8** network telescope from January 2021 to March 2021. The values presented represent the number of TCP packets observed per DSTIP address. Due to the high volume of TCP traffic and low volume of UDP, to demonstrate the statistics surrounding the datasets, this section primarily focused on TCP traffic.

In January, each unique DSTIP address received 118,186 packets in network telescope **146/8** on average. By looking at the *min* and *max* values, one can tell that there was a significant gap of about 17, 800 packets, which helps to explain the high value of *std* which measures the spread of data from the mean value. Looking at the *std* value for February in the same network telescope, one can note that the *std* value has gone down a bit since the gap between the minimum value and the maximum value is smaller. The highest value for network telescope **146/8** is observed in March which is easily explainable by looking and the significant gap that exists between the minimum and the maximum value.

A visual representation of the data for Network telescope **146/8** is displayed in **Figure 4.1**, which easily shows the maximum value shown in the March dataset. What these *std* values are essentially communicating is that the number of packets received by the unique DSTIP addresses is not as uniform as **Figure 4.1** is presenting them i.e. showing a few spikes. This is attributed to the presence of outliers in the datasets where in one instance, a single DSTIP address received more packets than anticipated or another received a small number of packets than anticipated, thus creating a range of values with which each of the DSTIP addresses could have received.

Network telescope **146/8** is the only telescope that had the least value spread between the maximum and the minimum values compared to Network telescope **155/8** and Network telescope **196-A/8**. On average, individual DSTIP addresses in Network telescope **196-A/8** received more traffic (163,916) than any other telescope. From Table **4.21**, Network telescope **155/8** and Network telescope **196-A/8** have high *std* values which are largely attributed to the fact that the maximum value increased by a huge margin in these two datasets on top of them having more outliers than telescope **146/8**. This is very true for January and February as shown in Network telescopes **155/8** and **196-A/8**. It is also important to note that the values computed for *std* are not just direct results of the gap between the *min* and *max* values, rather the study also took into account how many of the unique DSTIP addresses received traffic that was either above average or below average. The more DSTIP addresses received traffic away from the *mean* values, the higher the

std values. This can be seen by looking at Network telescopes **155/8** and **196-A/8** in **Table 4.21**, in particular in the month of February. From here, one can note that the *min* and *max* values are not far away from each other, but the computational value that was computed for the datasets' *std* are nowhere near each other. The *std* value for Network telescope **155/8** in February is at least two times more than the *std* value for Network telescope **196-A/8**.

While the *std* values may be high especially for Network telescopes **155/8** and **196-A/8**, **Table 4.21** also show us a significant term in the form of *median* which is quite close to the *mean* of their respective datasets. Essentially, these two values portray a picture that, although the *std* value is high, a majority of the data received by the unique DSTIP addresses gives a very good idea of how much each unique DSTIP address received in their respective months overall. Graphically, this is supported by **Figure 4.1** as most of the unique DSTIP addresses received uniform traffic.

The study took a similar approach of descriptive statistics to understand how many, on average (*mean*), unique SRCIP addresses each DSTIP received and monitor the measure of spread (*std*) between the DSTIP address that received most (*max*) unique SRCIP addresses and the one that received the least (*min*) number of unique SRCIP address. The results of such descriptive statistics are shown in **Table 4.22**. The *std* is significantly lower than the values observed when working with packets (see **Table 4.21**). This is particularly important in this study because if there is not much disparity in the unique SRCIP addresses, then it makes it easier to sample out in any format without having a lot of concern about missing out on significant loss in threat intelligence data. This is the case because it is the unique SRCIP addresses that are responsible for this traffic, thus making the disparities observed in the network traffic of little concern.

Table 4.22: Descriptive Statistics for No. of Unique SRCIP Observed per DSTIP

Network Telescope:	**146/8**		
Statistic	**January**	**February**	**March**
mean	18,784	17,094	19,938
median	18,756	17,089	19,935
std	206	147	198
min	17,566	15,954	18,141
max	19,565	17,888	20,702
Network Telescope:	**155/8**		
Statistic	**January**	**February**	**March**
mean	28,388	25,339	30,266
median	28,353	25,313	30,229
std	410	317	533
min	27,064	24,340	28,934
max	33,781	29,319	38,100
Network Telescope:	**196-A/8**		
Statistic	**January**	**February**	**March**
mean	44,275	28,929	32,241
median	44,260	28,909	32,229
std	248	190	221
min	42,692	27,598	30,698
max	45,332	29,738	33,627

The measure of spread (standard deviation - *std*) of the number of unique SRCIP addresses received by each unique DSTIP address from the *mean* was smaller than what was observed when looking at the number of packets each DSTIP address received. This has nothing to do with the number of unique SRCIP addresses being fewer than the packets, but rather somewhat uniform distribution of unique SRCIP addresses among the DSTIP addresses. The study will get back to this point in **Section 6.4** after doing some sampling.

Secondly, **Tables 4.22** also show that the *min* and the *max* values are not far off from the *mean*, supporting the same argument raised in the first point that there were small variations in the number of unique SRCIP addresses received by the unique DSTIP addresses. Larger values of *std* are observed in telescope **155/8** (shown in **Tables 4.22** and Network telescope **196-A**). However, Network telescope **155/8** contained neither the largest number of unique SRCIP nor the lowest volume of unique SRCIP addresses. Apart from data collected in March, Network telescope **196-A/8** reported the highest number of unique SRCIP addresses observed in any given month. This high volume of unique

SRCIP addresses corresponds with a high volume of traffic in January for Network telescope **196-A**. Other than that, there is no direct relationship between the high volume of unique SRCIP addresses and the number of packets contributed by the unique SRCIP addresses.

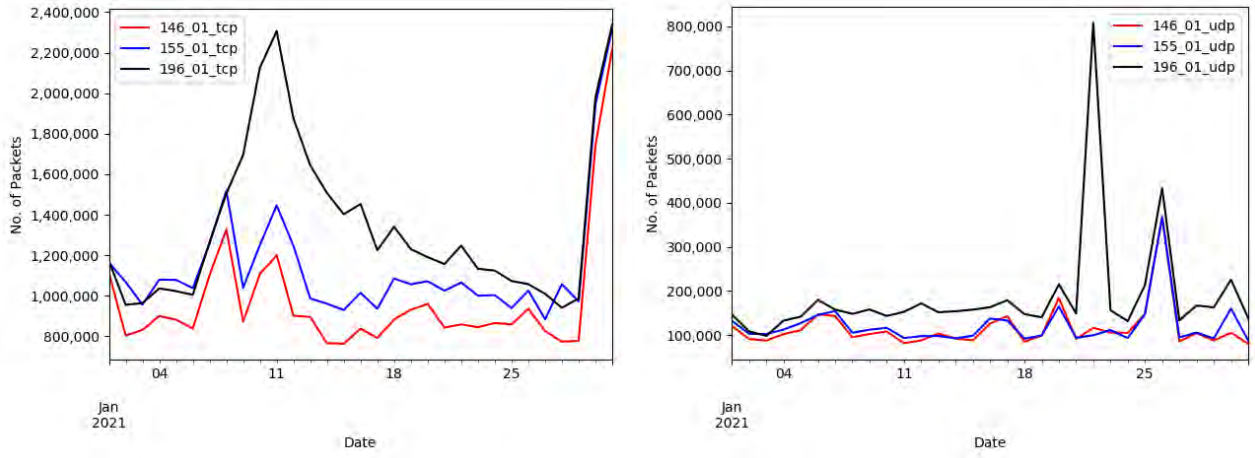
Overall, the study observed that the month of February did not contain a high volume of unique SRCIPs per DSTIP address. Though the differences for January and February are not that big for Network telescopes **146/8** and **155/8**, there is a gap of over 10,000 unique SRCIP addresses received per DSTIP in Network telescope **196-A/8**. This is observed by looking at all the statistical values in **Table 4.22**. The first assumption would have been that February has fewer days than the other two months, but the statistical values shown in Network telescope **196-A/8** cannot warrant such a huge drop in the average number of unique SRCIP. The largest number of unique SRCIP addresses observed per unique DSTIP were observed in March for Network telescopes **146/8** and **155/8** but for Network telescope **196-A/8**, its highest number of unique SCRIP addresses was recorded in January. There was a lot of disparity in Network telescope **155/8** in terms of how many unique SRCIP addresses each unique DSTIP address observed. This is shown by a big margin between the *min* and the *max* values but also its *std* is the highest among all three network telescopes every single month.

4.3 Graphical representation of TCP and UDP Datasets

The study shifted its attention from focusing on the statistical view of the data to a graphical view to aid in understanding and supporting the statistics shown in this chapter. There were two forms of graphical plots that were used to aid in giving an overview of how the data looked like: first, line plots were used followed by box plots to support the descriptive statistics already presented. Both plots (line and box plot) were used with the same overall goal in mind i.e. to offer a comparison among the six different datasets of each month and 18 in total for three months. Box plots on their own are useful for indicating whether a distribution is skewed and whether there are potential unusual observations (outliers) in the data set which cannot easily be seen in other plots.

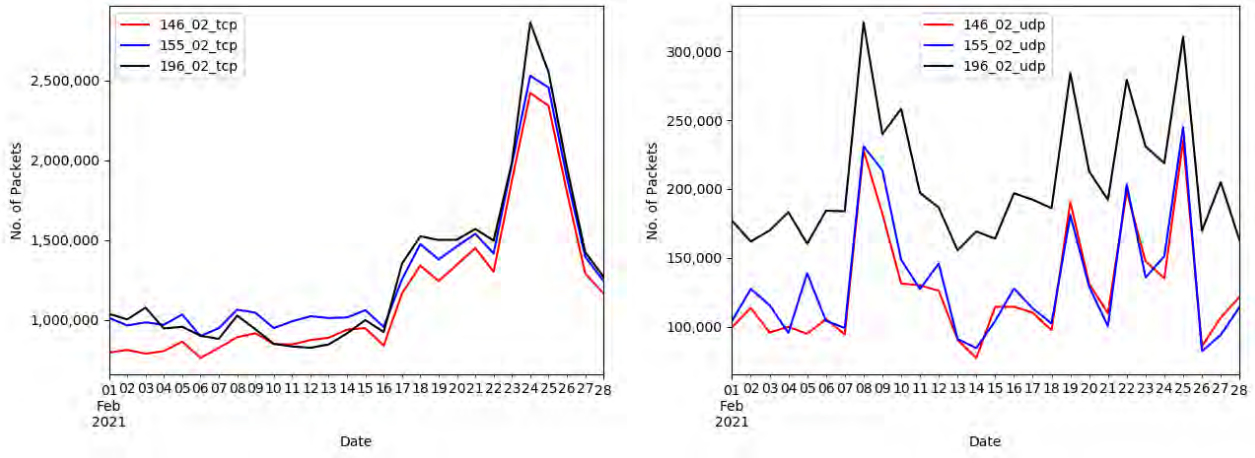
Figures 4.2 - 4.4 shows a time series of the 18 datasets and how the packets were distributed across each month under study. From these three graphs shown in **Figures 4.2 - 4.4**, one can easily see the gap that is there in terms of volume of traffic between the TCP datasets and the UDP datasets. These differences were also observed in TCP

and UDP packets from the data dictionaries and statistical tables in **Sections 4.1.1** and **4.2.3**, however, this time, a visual perspective of the same dataset is presented.



(a) TCP (b) UDP

Figure 4.2: January 2021 Time-based Traffic

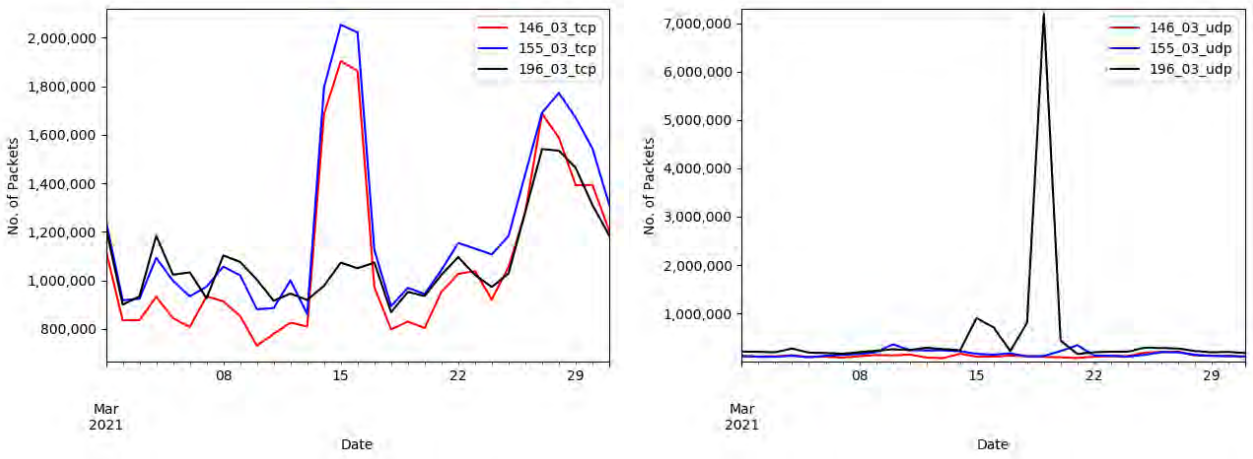


(a) TCP (b) UDP

Figure 4.3: February 2021 Time-based Traffic

Each plot has a timeline, represented in days in the *x-axis*, labelled *Date* and the *y-axis* shows the number of packets, labelled *No. of Packets*. Each of the three line plots are showing the number of packets observed in the network telescopes per day. Each individual plot represents a month worth of TCP and UDP traffic collected within a specified month. The line plots are colour coded to show which line represents which datasets and thus the legend is there to aid in the identification of that. When the study looked at traffic based

on the number of packets observed per unique DSTIP (See **Table 4.22**), it was observed that the highest volume of traffic was recorded in Network telescope **196-A/8**. This was confirmed by looking at the average number of packets each DSTIP received.



(a) TCP (b) UDP

Figure 4.4: March 2021 Time-based Traffic

The study also looked at the minimum and maximum values each DSTIP received in this network telescope. This notion was also supported by the data dictionaries that offered insight into the summary of the datasets (See **Tables 4.1 - 4.3**). Thus **Figures 4.2** and **4.3**, confirms this observation by showing a high volume of traffic for both TCP and UDP datasets observed in Network telescope **196-A/8** (represented by the black line). The major take away from the plots (**Figures 4.2** and **4.3**) is that the pattern in TCP traffic are generally similar with more variation in the UDP traffic.

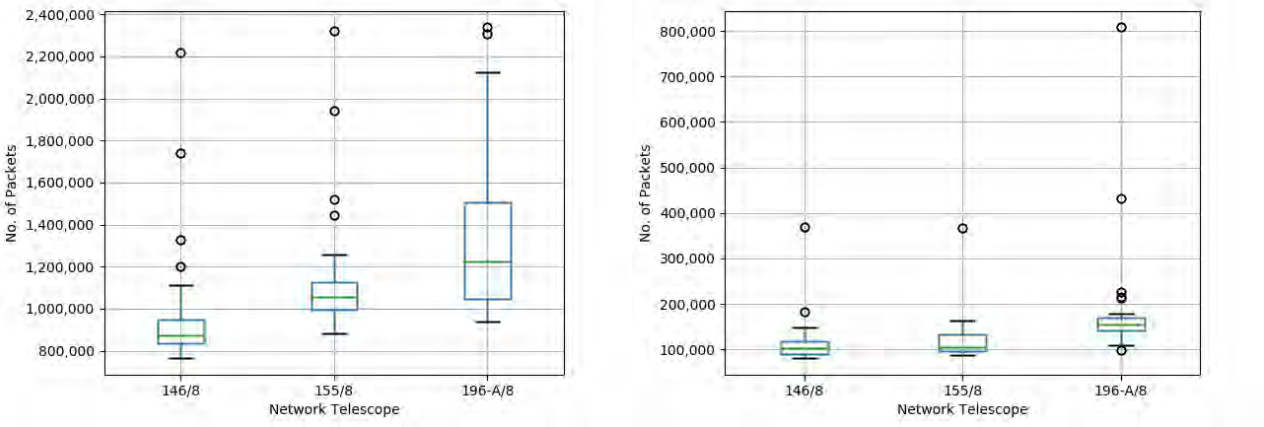
For January and February Network telescope **196-A/8** registered the highest number of traffic reported. It is only in March that Network telescope **155/8** reported more TCP traffic than **196-A/8**. This traffic is not reflected in the number of unique SRCIP observed in each network telescope (as **196-A/8** still had more unique SRCIP per DSTIP), however, it is observed when looking at the number of packets observed per unique DSTIP. Network telescope **196-A/8**, however, showed more packets for UDP traffic (see **Figure 4.4b**) which were largely received by Port **123/UDP**. In March, Port **123/UDP** received over 52% of the total UDP traffic for Network telescope **196-A/8**. The service that run on Port **123/UDP** and the potential vulnerabilities were explained in **Section 4.2.2**, **Table 4.19**. Supporting this observation is the Japan Computer Emergency Response Team Coordination Center (JPCERT/CC) Internet Threat Monitoring Report[13] which

[13]https://www.jpcert.or.jp/english/doc/TSUBAMEReport2020Q4_en.pdf

showed that in the previous quarter (October - December 2020), Port **123/UDP** was not in their top 10 ports that received more traffic, but the umber of hits for Port **123/UDP** rose at the beginning of March, making it number three on their ranking.

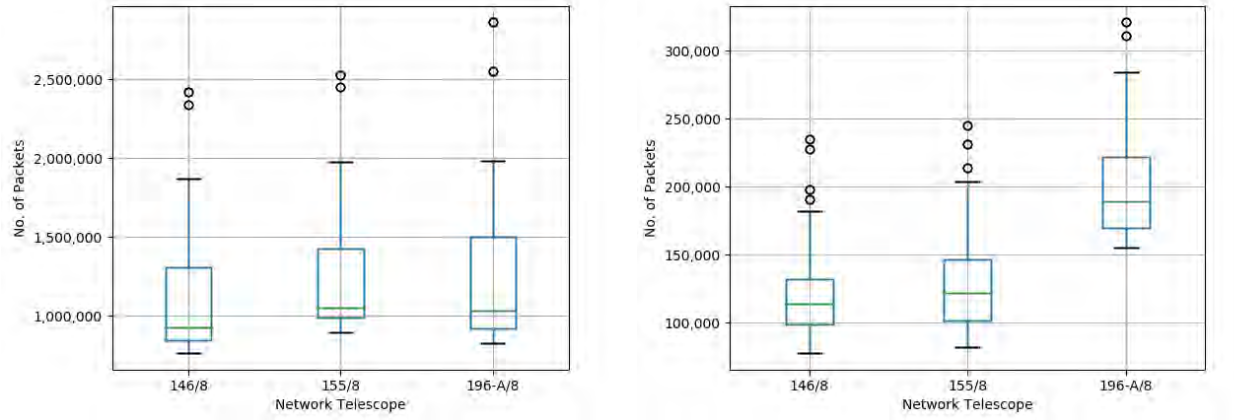
In the same months, January and February, Network Telescope **155/8** reported the highest average number of packets per DSTIP although the highest number of packets per DSTIP address was still recorded in Network telescope **196-A/8**. For the month of February, network traffic was at its lowest point in all three network telescopes, especially in the last week of the month. There are a lot of fluctuations in January for all three network telescopes' TCP traffic with each of the three network telescopes finishing with a spike. However, this spike, as traffic moved into February, is not reflected to start off at a peak (See **Figures 4.2a** and **4.3a**). On the other hand, in all the three datasets, the UDP traffic was highest in network telescope **196-A/8** with traffic record to reach 800,000 packets on the 22nd of February (see **Figure 4.2b**) and nearly 7,200,000 packets on the 19th of March (see **Figure 4.4b**). This is the highest number of packets recorded in a day by any dataset in all three months of observation.

While working with unique SRCIPs and DPORTs in **Sections 4.2.1** and **4.2.2**, it was observed that there was a lot of similarity in both ranking and the presence of specific unique SRCIPs and DPORTs between Network telescopes **146/8** and **155/8**. Looking at the graphical representation in **Figures 4.2 - 4.4**, it is apparent to note that the traffic patterns for Network telescopes **146/8** and **155/8** are more similar to each other than they are to Network telescope **196-A/8**. This is true for all three months of observation and for both TCP and UDP traffic. Traffic patterns looked more similar in network telescopes for TCP traffic in February and March (see **Figures 4.3a and 4.4a**).



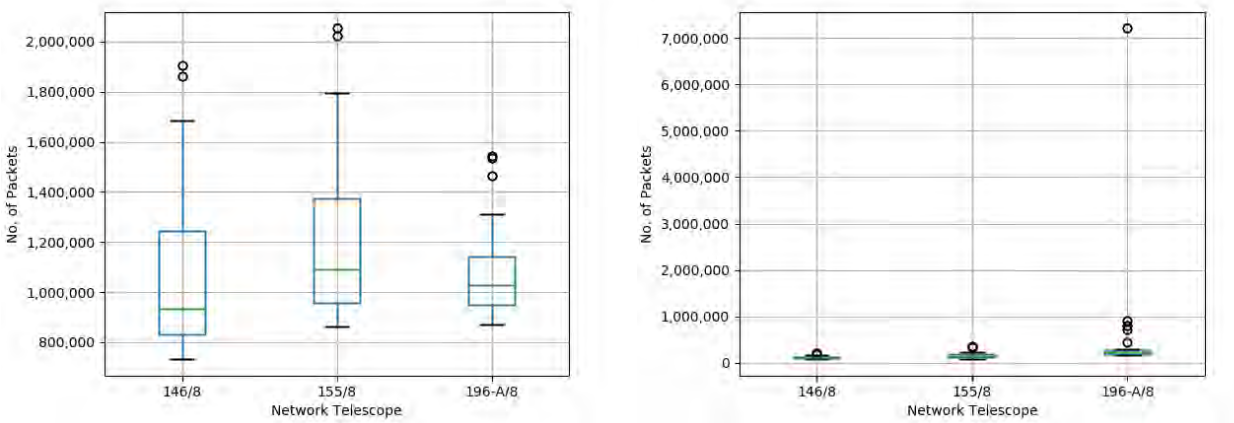
(a) TCP (b) UDP

Figure 4.5: Box plot showing Packet distribution in January 2021



(a) TCP (b) UDP

Figure 4.6: Box plot showing Packet distribution in February 2021



(a) TCP (b) UDP

Figure 4.7: Box plot showing Packet distribution in March 2021

As explained in **Section 4.2**, the high volume is not related to the high number of unique SRCIP addresses but rather unusual traffic that was received by specific unique SRCIP addresses on specific days. Plotting the datasets separately (UDP and TCP datasets) helps to easily identify the outliers present in each dataset. This is where the box plots come into play to detect the presence of outliers in the datasets. Looking at all three box plots (**Figures 4.5 - 4.7**), each month has outliers present in them. The presence of each outlier detected here has been explained in **Section 4.2**, thus this section will not focus on explaining these again but rather support the statistics already presented.

The presence of these outliers in each of the datasets helps to explain the fluctuations observed when looking at the time series plots which showed that traffic was not uniform throughout the time it was being collected. Each unusual amount of traffic recorded observed has shown (in part) that it is not necessarily as a result of the presence of new unique SRCIP addresses, but rather that the IP addresses present recorded more traffic than it had been doing previously. Most outliers here show unusual traffic recorded on a specific day. For instance, **Figure 4.7b** in Network telescope **196-A/8** shows an outlier the affects the total traffic recorded in UDP dataset for Network telescope **196-A/8**. This, as explained earlier in this section, was a result of Port 123/UDP receiving more daily hits than usual. In fact, March is the only month where UDP traffic registered more traffic than TCP traffic. There is also the presence of unique SRCIP that shows up only once in the data collection window and transmitted large quantities of packets. The presence of unusual traffic in each case led to the outliers observed throughout the datasets.

With regard to the volume of traffic received by each network telescope, it has already been established that Network telescope **196-A/8** received more packets than any other network telescope. This is with the exception of TCP traffic in March (see **Figure 4.7a**) where Network telescope **155/8** received more packets than any other dataset. The large volume of packets present in the Network telescope **196-A/8** is confirmed here by looking at the spread of each box plot representing Network telescope **196-A/8**. For instance, **Figure 4.6b** box plot for Network telescope **196-A/8** has a range that is higher than any other network telescope. The Box plot shown in Figure 4.5a reveals that the box plot for Network telescope **196-A/8** is longer (indicating the max and min values) than Network telescopes **146/8** and **155/8**. This is true as well in **Figure 4.5b**. The box plots also confirm that the range of packet distribution is fairly similar, especially for Network telescope **146/8** and **155/8**.

4.4 Chapter Summary

This chapter did an exploratory data analysis of the data that was used for this research study. The chapter began by explaining to the reader the source of the data that was used in **Section 4.1**. In this section, the study presented the data summary in form of data dictionaries but also explained how data sampling was conducted. To better understand the data characteristics, the study went further to do a descriptive analysis of all the datasets under study in **Section 4.2**. In this section SRCIP addresses that sent the most

traffic are ranked and presented. This is true for TCP and UDP. This was immediately followed by a network port breakdown in the same section, which ranked the ports based on the traffic they received and the type of protocol they used to transmit the data. A statistical approach to describe the average number of packets received by each unique DSTIP address is presented in this section as well.

A graphical representation of the data was shown in **Section 4.3**. Essentially, this section was added to support the observation made and established with the statistical approach. Both line plots and box plots are presented to make this confirmation. The chapter concludes with a summary. The datasets explored in this chapter are the ones that the study will look at in **Chapters 5**, and **6** using the statistical techniques introduced in **Chapter 3**.

5

Bootstrapping IBR Dataset

As explained in **Section 3.2**, bootstrapping works on the principle of starting with a dataset with an unknown underlying distribution from which a partially randomised sample of the available data is selected. Using any specific population parameter of interest, a normal distribution is formulated by applying a statistical function to the parameter of interest. This study used *mean* as the statistical parameter of interest to bootstrap IBR data. Building on this background, and tools and methods previously discussed in **Section 3.2**, an application of these methods is presented in this Chapter.

The chapter begins by explaining why bootstrapping is essential to this study in **Section 5.1**. This is followed by the research approach that was used to pre-process the data in **Section 5.2**. Two bootstrapping techniques were used to simulate the data and this is discussed in **Section 5.3**. The research findings were split into three categories: firstly, the study showed the relationship that exists between the average number of unique sources IP addresses in each bootstrap sample and the duration of observation in **Section 5.4**. This is shown for both the monthly and the quarterly datasets. **Section 5.5** follows, showing findings from bootstrapping; how bootstrapping operated on different levels of confidence

interval (CI) ranging from 80% CI to 99% CI. In this section, bootstrapping is split into parametric and non-parametric bootstrapping. The CI shows the range where the true average number of unique SRCIP addresses observed per hour in a given dataset lies. The research study presented a graphical representation of CI and the interpretation in **Section 5.6**. Recommendations and artefacts from this chapter are presented in **Section 5.7**. This chapter concludes with a summary in **Section 5.8**.

5.1 Bootstrapping Rationale

Bootstrapping is important to this study because the overall idea is that the study is working with the assumption that a user does not have access to a larger network telescope. Large, in this case, is defined as any network telescope sensor that contains more than 256 DSTIP addresses (i.e. /24 net-blocks). The number was chosen because this study worked with */24 net-blocks* which contain a maximum of 256 DSTIP addresses. This then means that a smaller network telescope will be any network telescope that can host less than 256 DSTIP addresses. In essence, if a network telescope user has 32 unique DSTIP addresses available, these will be the only ones to use for the telescope i.e. a user can only work with what they have. However, it is hard to tell how representable the data the user will collect with these 32 unique DSTIP addresses will be in relation to another user who has 128 unique DSTIP addresses or 256 DSTIP addresses. With bootstrapping, this study aimed to simulate the various samples of the baseline data to mimic the number of data points observed in the baseline data. The baseline dataset, in this case, represents the full data from /24 IPv4 net-blocks of network telescope while the different samples represent smaller network telescopes.

By simulating the samples to reproduce the baseline dataset, the study computed how different the bootstrap samples are from the actual baseline dataset. Using such differences between the bootstrap sample and the baseline dataset, the study shows how suitable simulating the IBR data is and how representable a smaller network telescope is, compared to a larger one. Overall, with bootstrapping, a user will know with a certain degree of confidence what range of unique SRCIP addresses will be collected given the number of unique DSTIP they have for network telescope usage. The confidence intervals (CI) will vary based on the size of the network telescope, which is defined as the number of unique DSTIP addresses to be used by the network telescope.

In this way, a user who does not have access to a larger IPv4 subnet that the network telescope can use will be able to relate or compute how many unique SRCIP addresses they

ought to get from the sub-network they have for their network telescope. The computed CI, coupled with the level of confidence chosen, will inform the user of the knowledge gap (in terms of threat intelligence) that they should be expecting. The study has an advantage in that it had access to baseline datasets with all the data points. As such, simulating samples from it offered a proper benchmark from which smaller samples could be gauged, something that a smaller network telescope user will not have.

5.2 Research Approach

There were two main bootstrap sample sizes which were based on the duration of observation. The first bootstrap sample contained 744 data points, the same number of data points that one would find in baseline data if they were to observe for one month. In this case, each hour within a month was considered a single data point. The second bootstrap sample contained 2,160 data points, the same number of data points one would find if they were to observe for three months, starting from January to March. Note that the data used is from 2021 and so February was treated to have 28 days. However, if each month is presumed to contain 30 days on average, the same number of data points would still be observed since January and March have 31 days. Thus two sets of experiments were conducted using the datasets presented in **Section 4.1**.

The study opted that all the bootstrap samples have the same number of data points as their baseline data as proposed by Kirby and Gerlanc (2013); Chamandy *et al.* (2015); Efron and Hastie (2016). The reason for creating bootstrap samples of the same size as the baseline data was to ensure that there was comparability between the baseline data and the bootstrap samples (Chamandy *et al.*, 2015; Hesterberg, 2015). Secondly, by having bootstrap samples of the same size as the baseline, the need of standardising the bootstrap samples was eliminated since they contain the same number of data points as the baseline (Efron and Hastie, 2016). In addition to this, the study used this approach to ensure that the standard errors observed in the original dataset are reflected in the bootstrap samples as compared to having hypothetically larger or smaller samples (Hesterberg, 2015). This essentially meant that the number of observations was the same but the composition was different just as explained in **Section 3.2**.

Considering that each of the unique DSTIP addresses in the network telescopes registered network traffic from outside the network, another approach which directly related to the main research question was identifying the number of unique SRCIP addresses in our

samples i.e. how many unique SRCIP addresses were registered by a small group of destination IP addresses throughout the observation period? In order to do this, the study randomly sampled the baseline data into six categories of subnet equivalent sizes i.e. 4, 8, 16, 32, 64 and 128 unique DSTIP addresses. Note that the sample sizes mimic the number of unique DSTIP addresses found in different subnets. These samples in this study are referred to as *subnet equivalents* because they contain the same number of unique DSTIP addresses that one would find in an actual network subnet.

Each initial input data was made up of samples containing the *date* in which the data was collected, a set of unique DSTIP addresses that received the traffic, and those that made up the sample of interest (samples of sizes 4, 8, 16, 32, 64 and 128). The input data also contained a unique *SRCIP address* field. Once the DSTIP addresses were used to create a sample, the input data was further transformed to contain the date field (the field was split into hours and not days) and the number of unique SRCIP addresses collected on an hourly basis. To come up with standardized samples, a fixed number of data points computed on an hourly basis was calculated using the baseline dataset data points. i.e. the total number of hours observed during one month and three months' periods as explained earlier in this section. In addition to this, the number of hours contained in the baseline dataset (744 for monthly datasets and 2,160 for quarterly datasets), was the number of times that each sample was simulated. Python scripts used to process data for this chapter are appended in **Appendix D**

Each baseline dataset (named after the network telescope) contained all the 256 DSTIP addresses, from which DSTIP addresses were randomly selected into the sizes of 4, 8, 16, 32, 64 and 128 DSTIP addresses to create new samples (subnet equivalents). It was from these subnet equivalents that the bootstrap samples were generated and the computation of *confidence interval (CI)* conducted from it. The study also bootstrapped the baseline dataset which contained all the 256 unique **destination IP** addresses. This way, it would be easier to see how far off the smaller subnet equivalents performed against the baseline bootstrap. These samples represent different sizes of the network telescope 'lens'.

From each dataset, outliers were removed to ensure that they did not affect the accuracy of the results. Using the *Interquartile Range (IQR)*, a range that contained at least 80% of the data points in each dataset was identified, thus any number falling outside this range was detected and treated as an outlier. Only random samples were used for bootstrapping because CI calculations assume you have a genuine random sample of the relevant population (Akter, 2014). If the sample is not truly random, one cannot rely on the intervals computed. In our case, to make the results reproducible the study seeded

the first sample value. This was done by fixating the starting value of each simulation. This means that if the reader wished to conduct the experiment on their own, it would make it easier to arrive at the same results. **Appendix D** shows where the script for this process is presented.

5.3 Bootstrapping Techniques

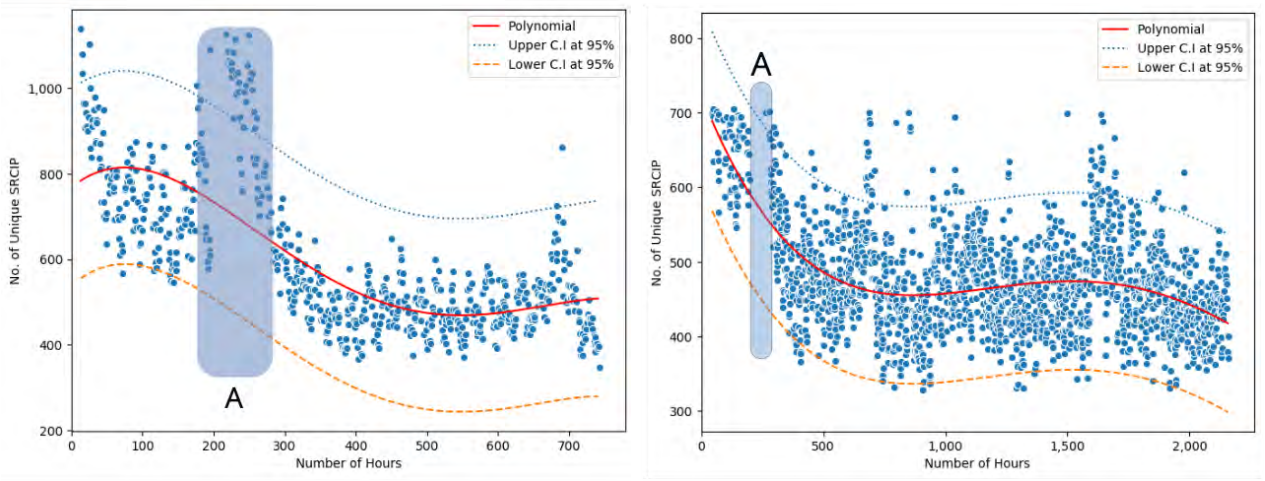
In this study, bootstrapping was categorised into two parts: parametric and non-parametric bootstrapping. Thus the results presented will be in the form of either tables or graphs which are labelled based on whether it is parametric or non-parametric. In addition to this, tables and graphs will show the name of the network telescope from which the data was collected. The results will also show the duration of observation or data collection. More details on these two bootstrapping techniques are found in **Section 3.2**.

5.4 Regression Analysis Findings

In this section, the study presents its findings regarding the relationship that exists between the number of unique SRCIP addresses observed in the network telescope and what happens to the volume of these unique SRCIP addresses over time. For demonstration purposes, the section presents six baseline datasets presented in the form of plots (shown in **Figures 5.1 - 5.3**). Each of the plots represents a /24 IPv4 baseline dataset from the network telescopes under study. During bootstrapping, the study used two main bootstrap sizes. The first bootstrapping samples contained 744 data points, representing the number of hours present in a month that has 31 days (in our case January and March). Secondly, the study conducted another set of bootstrapping with each sample containing 2160 data points (*January - March*). Please note that these graphs used in this section are directly from the baseline datasets and are not bootstrap samples. As such, the relationship between the number of unique SRCIP addresses and time presented in these polynomial regression plots is not in any way a result of bootstrapping, but rather the true reflection of the actual datasets.

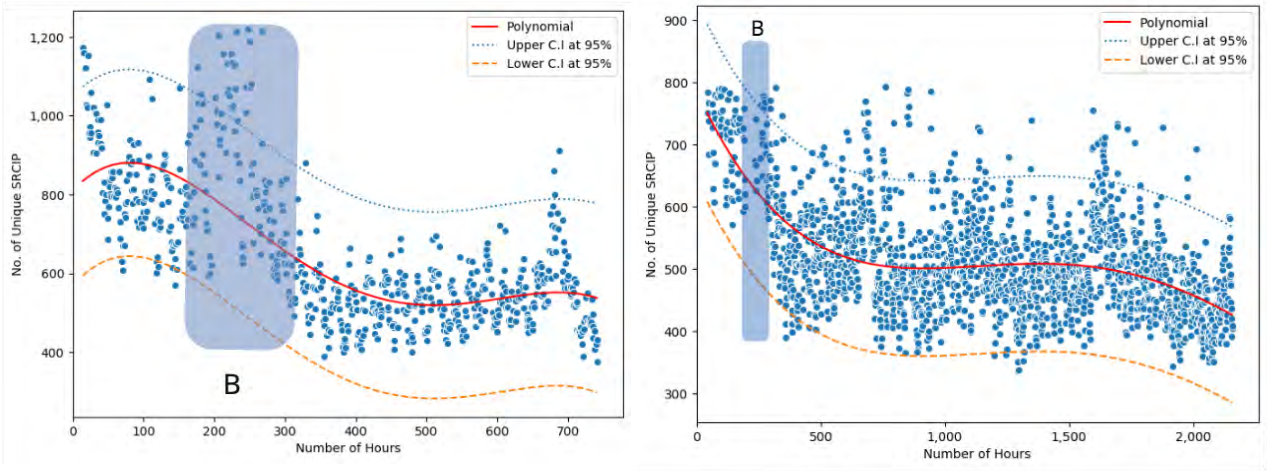
Each of the *Y-axis* presented in **Figures 5.1 - 5.3** show the number of unique SRCIP addresses while the *X-axis* show the number of hours within the given dataset. In other words, the plots display a polynomial regression that shows the relationship between the

total number of unique SRCIP addresses observed on an hourly basis. Each plot shows two datasets with every plot labelled *(a) January - March* showing three months worth of data while the plots labelled *(b) January* show the data collected in January. Thus all the datasets for the data collected in January show 744 hours while the datasets collected from January to March show a total of 2160 hours. The upper and lower confidence interval shown in the datasets were plotted at *95% Confidence Interval (CI)*. At the time of plotting, all outliers had been eliminated using IQR as explained in **Section 5.2**. However, the study did make plots with outliers included in the datasets. These plots with outliers can be found in **Appendix E.1**.



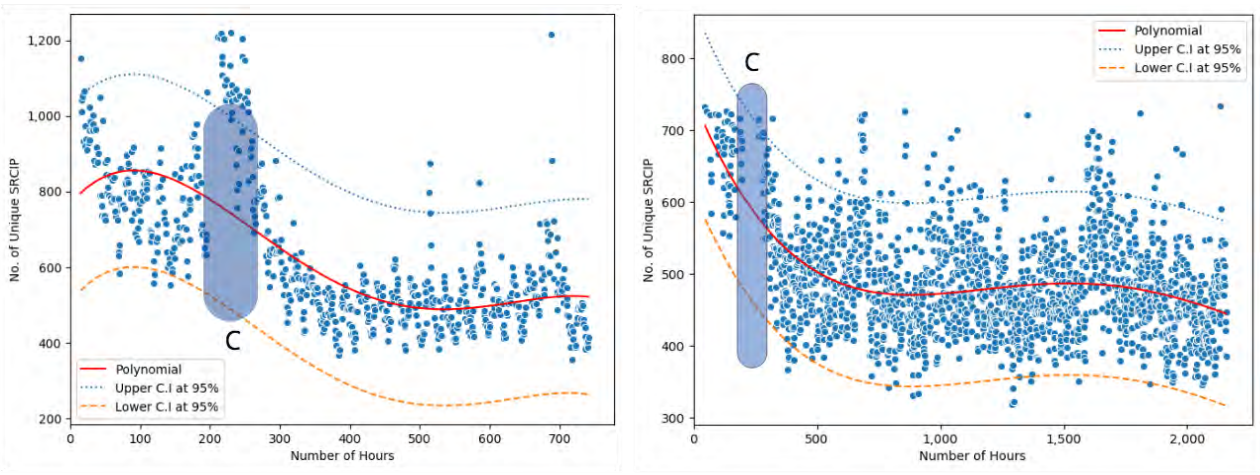
(a) January 2021 (b) January - March [2021]

Figure 5.1: 146/8: Number of Unique SRCIP observed/hour



(a) January 2021 (b) January - March [2021]

Figure 5.2: 196-A/8: Number of Unique SRCIP observed/hour



(a) January 2021 (b) January - March [2021]

Figure 5.3: 155/8: Number of Unique SRCIP observed/hour

All regression analysis plots for the remaining months (February - March) for every dataset can be found in **Appendix E.2**. The polynomial line (represented by the red line) is regression line that is explained in this paragraph. One of the main findings that can clearly be observed from all these plots is that, as the number of hours of observation increased, the volume of the unique SRCIP addresses observed per hour was declining. The plots show an inverse relationship between time and the number of new unique SRCIP addresses within any given month. Plots that had 744 data points in all the three network telescopes (see **Figures 5.1a, 5.2a** and **5.3a**) show a similar pattern where, at the beginning of the month of January, they register a high number of unique SRCIP addresses observed per hour. Then the number of new unique SRCIP takes a steady decline which is later followed by a slight rise that leads into a new month. This slight rise coming in at the end of the month is the presence of a high volume of new unique SRCIP addresses that were not present at any point in time during the month of January. However, note that in each plot there is a region labelled **A, B** and **C** (see in **Figures 5.1 - 5.3**) which shows fewer to no number of data points on the plots. In these regions, each network telescope registered a high volume of unique SRCIP addresses than any other point. This happened between 8^{th} - 12^{th} January 2021, which also resulted in high volume of traffic (See **Section 4.3**). The use of IQR and plotting of the data at 95% CI ensured that these data points are treated as outliers. This is why regions labelled **A, B** and **C** appear to have fewer to no data points than any other area on the plots. The study has shown regression plots containing this outage for each plot in **Figures E.1 - E.3**. More regression analysis plots are found in **Appendix E**.

Noteworthy is the change in the volume of unique SRCIP addresses registered per hour in the *Y-axis* of **Figures 5.1b, 5.2b** and **5.3b** when compared to **Figures 5.1a, 5.2a** and **5.3a**. *January - March [2021]* plots show a lower volume of unique SRCIP addresses observed per hour than *January 2021* plots. In addition to this, the slope for the *polynomial* regression line at the beginning of the *January - March* plots is steeper than that shown in the *January* plots. This pattern seen in *January 2021* plots is also observed in *February 2021* and *March 2021* plots (see **Appendix E.2**). This is the case because monthly dataset plots only focus on unique SRCIP addresses present within that month while quarterly plots focused on three months worth of data. For instance, using the month of March as an example, the first time some of the unique SRCIP addresses are being observed is not in March but rather in January and then February. Thus by the time observation is done in March in the three months time span, these unique SRCIP addresses registered as new in monthly datasets are not registered as new unique SRCIP addresses in the larger datasets. This holds true if the observation is extended to a longer duration i.e. a six months worth of analysis will show less number of unique SRCIP addresses per hour and the regression line will be steeper than that of three months worth of data analysis.

There is a steady presence of new unique SRCIP addresses present in February (See **Figures 5.1, 5.2** and **5.3**) which takes us into the month of March at which point the graphs for the large datasets hit their lowest points. The uniformity in the flow of unique SRCIP addresses in both large and small datasets across all three network telescopes supports the study done by (Nkhumeleni, 2014) that the distributed network telescopes at Rhodes University do collect data that is similar. With this view, the researchers of this study do expect that if more network telescopes are added to the same network, the results will take the same pattern. What this means to network telescope users is that, if they connected different network telescope sensors to their network, the same pattern in all their network telescopes is expected to take the same shape as any one of them. On a grand scale, this relationship between the number of unique SRCIP address versus time, means that any given network telescope (if monitored for a longer period of time) will show a curve similar to the ones presented in **Figures 5.1b, 5.2b** and **5.3b**. The presence of unique SRCIP addresses probing the same network repeatedly over time has been a known phenomenon in network telescope research (Pearson, 2020). Usually, such SRCIP addresses are from persistent networks (networks that probe other networks at least once each month).

When the researchers randomly sampled the baseline datasets to formulate subnet equivalents and plot the outcome of it, all the subnet equivalents exhibited the same relationship over time. In addition to this, when a 256 baseline datum was bootstrapped and its confidence interval computed (more details in **Section 5.5**), it was observed that the CI range (which is based on the count of values only) provided for each of the bootstrapped samples showed a very similar range of values that were observed in these baseline datasets. Moreover, when the study compared the volume of unique SRCIP addresses observed for March datasets and those observed from January to March, it was observed that the volume within each CI range had declined as evidenced in the plots shown in **Figures 5.1, 5.2** and **5.3**.

The study was not interested in predicting any upcoming unique SRCIP addresses within the observable period and thus it did not go further to a point of creating regression equations to show future behaviour (prediction). The main objective was to observe what happens to the unique SRCIP addresses over time when it comes to regression analysis. The study also looked at linear regression analysis, and although the trend was the same, the plots from it did not fully represent all the data points as did the polynomial regression shown in this section.

5.5 Confidence Interval Findings

This section displays how bootstrapping operated on different levels of confidence interval (CI) ranging from 80% CI to 99% CI. Each degree of confidence (80% CI to 99% CI) offers certainty that the true average number of unique SRCIP addresses observed per hour in a given dataset lies within the selected range. The data used throughout this section is from the TCP datasets. Each table herein shows the range in the average number of unique SRCIP addresses observed per hour for each respective sample. Note that normalisation of the findings for each dataset sample was done by ensuring each dataset sample contained the same number of data points. Secondly, each non-parametric bootstrap sample used its own mean to compute. This way the results become comparable (Dixon, 2006; Hesterberg, 2015).

The core idea behind bootstrapping IBR data is to show a user the range in which the average number of unique SRCIP addresses observed per hour would fall should they happen to have fewer IP addresses. For this study, **few** was defined as anything below a /24 IPv4 address block i.e. 256 unique DSTIP addresses. With this in mind, one should

not expect a /24 IPv4 baseline sample which has 256 IP addresses to display a CI identical or similar to $/_e$*32 subnet equivalent* as that is not possible. More unique DSTIP will still collect more unique SRCIP addresses. However, with CI computed from bootstrapping, a user would have the confidence to know what kind of threat intelligence data (volume-wise) to expect given the number of unique DSTIP addresses they are willing to use for their network telescope. In this way, bootstrapping addresses one of the key questions of this research by giving the network telescope user the level of confidence they should have in the data collected by their network telescope. Having three /24 IPv4 blocks as the benchmark datasets helps in comparing the findings from this benchmark to those found in smaller bootstrap samples which are a representation of smaller network telescopes. Different percentages of confidence will offer a network telescope user different CI ranges. Different subnet equivalents will also offer different ranges of CI.

Each table shows percentages of confidence at which different ranges of CI were computed. These have been labelled CI levels which range from 80% to 99%. In this section, datasets are presented by using the name of the network telescope followed by the name of the subnet equivalent. For instance, a /24 IPv4 baseline bootstrap sample for Network telescope **146/8** will be presented as **146/8 - $/_e$24**, which means that this sample had *256* unique DSTIP addresses. In other words, the study bootstrapped a baseline dataset without which comparison with sub samples would not be possible. A bootstrap sample from the same network telescope which had *128* unique DSTIP addresses will be presented as **146/8 - $/_e$25**. If the bootstrap sample belongs to Network telescope **155/8** and has 8 unique DSTIP addresses, it will be presented as **155/8 - $/_e$29**

As explained earlier in **Section 3.4**, CI coverage is the probability that the CI includes the true parameter, under repeated sampling from the same underlying population. The true parameter is the real value that shows an actual number of unique SRCIP addresses in a given dataset. For example, assuming that the mean value of a baseline dataset has been computed to be 450 from the first simulation which has involved multiple sampling of the dataset to come up with the bootstrap sample, if the data is simulated again, it is unlikely that the average will be the same as before when the sampling is random. So, what bootstrapping does is to make sure that a series of simulations done on the data samples create a range in which each time a simulation is done on the data sample, the mean value computed from it will fall within this range. Thus the expected mean is what can be described as the true parameter.

In our case, baseline and subnet equivalent datasets are sampled 744 and 2160 times and simulated for the same amount of times. Each bootstrap sample will have its own CI based on the composition of the sample. Assume the interval is between 445 unique SRCIP addresses per hour and 555 unique SRCIP addresses per hour. If the researchers take 100 random DSTIP addresses to form a sample from the baseline dataset at 95% CI, the average number of unique SRCIP addresses observed per hour in that sample should fall between 445 unique SRCIP addresses per hour and 555 unique SRCIP addresses per hour in 95 of those samples.

If the researchers want even greater confidence, they can expand the interval to 99% confidence. Doing so invariably creates a broader range, as it makes room for a greater number of sample means. If they establish the 99% CI as being between 445 unique SRCIP addresses per hour and 560 unique SRCIP addresses, they can expect 99 of the 100 samples being evaluated to contain a mean value between these numbers. If the CI is computed at an 80% confidence level, the interval is then expected to be smaller and 80 of the 100 samples being evaluated will contain a mean within the interval computed. The range computed from the bootstrap sample is what gives the user the level of confidence needed in the data in order to make informed decisions. In this section, the study focused on interpreting the results and what they mean to this research.

The study computed CI at different levels of confidence in order to offer a wider scope from which to work with. Depending on the nature of the study, different fields use different levels of CI in order to attain their objectives. For instance, medical practitioners demand the highest level of confidence because they deal with life. In our case, the study started with an 80% confidence level until 99% CI. As explained in **Section 5.3** bootstrapping was categorised into two parts in this study: parametric and non-parametric bootstrapping. Thus the results will be split into two sub-categories i.e. CI for parametric and non-parametric bootstrapping. Within these two categories, the results are also split further into two, based on the duration of data collection. The first set of tables focus on the larger datasets, which spanned over three months (January to March). These contained 2,160 data points per dataset, which invariably created 2,160 samples. The second section of results focused on the monthly analysis which covered the month of March alone. These contained 744 data points creating 744 samples from each data sample. Similar work for the month of January has been accepted for a Southern Africa Telecommunication Networks and Applications Conference (SATNAC) and can be found in (Chindipha and Irwin, 2021). The CI findings from this paper have been appended in **Appendix F**.

5.5.1 CI for Parametric Bootstrapping Simulation

To ease understanding of the results in this section and **Section 5.5.2**, consider CI as a range of values defined by an upper and lower bound, with the upper bound being above the statistical mean of the sample under study and lower bound being below the statistic's mean of the same population. The CI is likely to contain an unknown population parameter being evaluated or examined. This study identified *mean* as its population parameter because it is an unbiased estimate of the corresponding population parameters (Hesterberg, 2015; Efron and Hastie, 2016). Thus the true population parameter of this study is the actual value of the mean that has been computed from the sample being studied. The likelihood of finding this unknown mean in any sample under study is defined by the level of confidence used to compute the CI. Confidence level refers to the percentage of probability or certainty that the CI would contain the true population parameter when the reader draws a random sample many times.

With statistics, especially probability, there is always a possibility that the observed (or computed) interval may overestimate or underestimate the true mean value, hence the need to accommodate the level of certainty i.e. confidence level (which is similar to a probability) (Efron, 1992; Efron and Hastie, 2016). So if a CI for an unknown population mean is computed at 95% confidence level, what this means is that the 95% CI is the likely range of the true, unknown mean of the population under study (Hesterberg, 2015). In other words, there is a 95% probability that the CI will contain the true population mean.

A researcher cannot work with all the samples under study. As such, it becomes practical to work with bootstrapping to simulate all possible ranges of values in order to accommodate all values in the actual population should the values in a sample change. This maximises all possible values accommodated in each sample. It is vital to acknowledge that the CI does not in any way exhibit the variability in the unknown parameter. What it rather does is portray the amount of random errors in the sample and provide a range of values that are likely to include the unknown parameter (Dixon, 2006; Akter, 2014). With this knowledge, the study will now present its findings.

Tables 5.1, 5.2 and **5.3** show the CI levels at which CI was computed and the number of unique DSTIP addresses contained in each parametric bootstrap sample. These results were computed from parametric bootstrapping for network telescopes **146/8, 196-A/8** and **155/8**. Note that in each of the bootstrap samples, the number of unique DSTIP is representative of the size of the network telescope. A **146/8 - /$_e$26** bootstrap sample

means that it has 64 unique DSTIP addresses and as such, it is a network telescope that can accommodate 64 unique DSTIP addresses. In all of these tables, as the confidence levels increased, the CI range increased meaning that the user gets more range of possible values from which to identify how many unique SRCIP addresses, on average, one could get if the user bootstraps the DSTIP addresses available in their network telescope. A bootstrap sample computed at a narrow CI range of 80% CI level means that there is a high likelihood of missing out on the actual number of unique SRCIP addresses, hence given a lower CI level.

Table 5.1: 146/8: CI for No. of Unique SRCIP/hour [Jan - Mar]

	CI Level			
Bootstrap sample	80%	90%	95%	99%
146/8 - /e24	[509 - 519]	[508 - 520]	[507 - 521]	[505 - 523]
146/8 - /e25	[255 - 260]	[254 - 261]	[254 - 261]	[253 - 262]
146/8 - /e26	[130 - 132]	[129 - 133]	[129 - 133]	[128 - 134]
146/8 - /e27	[67 - 68]	[66 - 68]	[66 - 69]	[66 - 69]
146/8 - /e28	[33 - 34]	[33 - 34]	[33 - 34]	[33 - 34]

Table 5.2: 196-A/8: CI for No. of Unique SRCIP/hour [Jan - Mar]

	CI Level			
Dataset	80%	90%	95%	99%
196-A/8 - /e24	[573 - 583]	[571 - 585]	[569 - 586]	[567 - 589]
196-A/8 - /e25	[291 - 296]	[291 - 297]	[289 - 298]	[288 - 299]
196-A/8 - /e26	[147 - 150]	[147 - 151]	[147 - 151]	[146 - 152]
196-A/8 - /e27	[72 - 74]	[72 - 74]	[72 - 75]	[72 - 75]
196-A/8 - /e28	[37 - 38]	[37 - 39]	[37 - 39]	[37 - 39]

Table 5.3: 155/8: CI for No. of Unique SRCIP/hour [Jan - Mar]

	CI Level			
Dataset	80%	90%	95%	99%
155/8 - /e24	[522 - 532]	[521 - 533]	[520 - 534]	[517 - 537]
155/8 - /e25	[261 - 266]	[260 - 266]	[259 - 267]	[258 - 268]
155/8 - /e26	[134 - 136]	[133 - 137]	[133 - 137]	[132 - 138]
155/8 - /e27	[67 - 68]	[67 - 69]	[66 - 69]	[66 - 69]
155/8 - /e28	[34 - 35]	[34 - 35]	[34 - 35]	[34 -35]

As the confidence level moves from 80% CI towards 99% CI, the range gets wider, offering more possible value from which the average number of unique sources contained within the given subnet equivalent could contain. In addition to this, note that as the number of unique DSTIP contained in each bootstrap sample decreases (for instance moving from **146/8 - /$_e$24 - 146/8 - /$_e$28**), the average number of unique SRCIP observed per hour decreases as well. This is a direct result of **146/8 - /$_e$28** containing fewer DSTIP addresses, which in turn receive fewer unique SRCIP addresses per hour. The intervals herein are not computed per DSTIP but rather per hour, as a time series analysis. This, in turn, lowers the overall mean as smaller network telescopes have been known to collect less threat intelligence data than larger network telescopes. The only key question not addressed is; what is the difference, in terms of data collected, when these smaller network telescopes are compared against the larger network telescopes?

This chapter answers this question in part by offering quantitative values that one should expect given the size of the samples a user is working with. In **Tables 5.1**, looking at 80% CI for Network telescope **146/8**, one can observe that the average number of unique SRCIP addresses has declined from the range of [573 - 583] in **146/8 - /$_e$24** to [37 - 38] in **146/8 - /$_e$28**. An 80% confidence of a [37 - 38] CI simply means that there is an 80% chance that the confidence interval of [37 - 38] contains the true population mean. i.e. the real mean of an actual sample. So if a user is using a network telescope with 16 unique DSTIP addresses that are randomly sampled, the user can be 80% certain that within every hour that the network telescope is being used, the 16 DSTIP addresses being used will collect an average number of unique SRCIP that can range between 38 to 39.

The interpretation of CI presented in the preceding paragraph applies to all the bootstrap samples in **Tables 5.1, 5.2** and **5.3**. The remaining percentages, 10% for example when looking at 90% CI, acknowledge that during computation of the CI there is a 10% likelihood that the actual population parameter understudy could fall outside the interval being computed. This is the case because CI computation acknowledges the likelihood of the actual population parameter not being accurately computed but it attaches the odds of this happening. CI essentially tells the researcher how well one has determined the mean of the sample that has been bootstrapped.

In addition to this, the interval observed within each sample declines as the sample sizes get smaller. **146/8 - /$_e$24** has wider interval from which the potential average number of unique SRCIP addresses can be identified when compared to **146/8 - /$_e$28**. This is the case because there is less variation in smaller samples as compared to large samples. As the confidence level increases from 80% to 99%, the CI also increases. This observation

is true for **Tables 5.2** and **5.3** as well. What this essentially means is that accuracy of identifying the true population **mean** is better with high confidence levels and with large bootstrap samples. i.e. large samples are still better than smaller samples. This is because their CIs are broader, offering a high likelihood. The dimension that this study adds to this knowledge is that with CI computed, the proportions are known i.e. one can quantify how the baseline differs from the samples using the proportion of the intervals. What is also true in all these tables is how many unique SRCIP, on average, one would get in each sample. With this benchmark, it is now easy to compute the proportions based on the CIs computed.

5.5.2 CI for Non-Parametric Bootstrapping Simulation

Table 5.4: 146/8: CI for No. of Unique SRCIP/hour [Jan - Mar]

	CI Level			
Bootstrap Sample	80%	90%	95%	99%
146/8 - /e24	[471 - 475]	[471 - 476]	[470 - 476]	[469 - 477]
146/8 - /e25	[237 - 239]	[236 - 239]	[236 - 240]	[236 - 240]
146/8 - /e26	[119 - 120]	[119 - 120]	[119 - 121]	[118 - 121]
146/8 - /e27	[60]	[59 - 60]	[59 - 60]	[59 - 61]
146/8 - /e28	[29 - 30]	[29 - 30]	[29 - 30]	[29 - 30]

Table 5.5: 196-A/8: CI for No. of Unique SRCIP/hour [Jan - Mar]

	CI Level			
Bootstrap Sample	80%	90%	95%	99%
196-A/8 - /e24	[510 - 515]	[509 - 515]	[508 - 516]	[507 - 517]
196-A/8 - /e25	[258 - 260]	[258 - 260]	[257 - 261]	[256 - 262]
196-A/8 - /e26	[130 - 131]	[129 - 131]	[129 - 131]	[129 - 132]
196-A/8 - /e27	[64 - 65]	[64 - 65]	[64 - 65]	[63 - 65]
196-A/8 - /e28	[32 -33]	[32 -33]	[32 -33]	[32 -33]

Table 5.6: 155/8: CI for No. of Unique SRCIP/hour [Jan - Mar]

	CI Level			
Bootstrap Sample	80%	90%	95%	99%
155/8 - /e24	[486 - 490]	[485 - 491]	[485 - 492]	[484 - 493]
155/8 - /e25	[243 - 245]	[242 - 245]	[242 - 245]	[242 - 246]
155/8 - /e26	[123 - 124]	[122 - 124]	[122 - 124]	[122 - 124]
155/8 - /e27	[61]	[61 -62]	[61 -62]	[60 -62]
155/8 - /e28	[30 -31]	[30 -31]	[30 -31]	[30 -31]

Tables 5.4, 5.5 and **5.6** show results that were computed from non-parametric bootstrapping for network telescopes **146/8, 196-A/8** and **155/8**. The interpretation of the results is the same, but, this section will mostly point out how parametric bootstrapping findings relate to non-parametric bootstrapping. It will also focus on those observations only found in non-parametric bootstrapping. One major difference observed in the non-parametric bootstrap samples is that the average number of unique SRCIP addresses observed per hour has declined, especially for $/_e$**24 -** $/_e$**26**. This is something that will be explored more in **Section 5.5.4**. The lower subnet equivalents do not show a major change in the average number of unique SRCIP observed per hour. This essentially is communicating that smaller subnet equivalents tend to show results that are similar irrespective of the bootstrapping technique used. A look at **146/8 -** $/_e$**27** and **155/8 -** $/_e$**27** in **Tables 5.4** and **5.6** reveal that there is no range of values provided, it is just a value computed from it. The interpretation, however, remains the same i.e. the average number of unique SRCIP addresses observed at 80% CI is fixed at 60 and 59 respectively.

This level of certainty leaves a 20% chance of the true mean not being the identified value. It also offers less confidence in the findings of such a value since there is no variation to work with. On the other hand, at 99% CI, there is a range of values that does accommodate variation and thus offers more confidence in such a spectrum. The observation made in **Section 5.5.1** also applies in non-parametric bootstrapping i.e. as subnet equivalent size increased, the CI range increased as well, thus smaller samples do not offer more room for variability, which in turn offer less confidence when compared to larger samples. The same observation is made as the level of confidence increased from 80% to 99% on different samples.

Secondly, as in parametric bootstrap samples, large subnet equivalents show more unique SRCIP addresses observed per hour as compared to smaller subnet equivalents. This was expected. But what was not known is the proportion that exists between the smaller subnet equivalents and the large subnet equivalents. A look at **Tables 5.1** - **5.6** shows that there are very slight variations between 95% and 99% CI level. The study thus recommends the use of the data at least at 95% CI level because the variations between 95% and 99% CI level are not big. In this way, there will be a high chance of reflecting on the reflections observed in the baseline data. A 95% or 99% CI for any given sample size means that the data user is 95% or 99% confident that each of the allocated samples will contain a specific average amount of unique SRCIP addresses in the available pool of DSTIP addresses. Confident to a point of knowing how much is not accounted for if the number of unique SRCIP observed is anything lower or higher than the indicated interval present in their network telescope. The number of unique SRCIP addresses observed in

the DSTIP addresses of the host's network telescope is going to be different based on the duration of observation and the size of the network telescope 'lens'.

Another profound observation confirmed in this study was that bootstrapping balanced the scales that come with sampling in that, whether the sample is randomly sampled or sequentially sampled, the proportionality of the unique SRCIP addresses between a subnet and its subnet equivalent is similar (more on sequential sampling in **Chapter 6**). When the data was sequentially sampled to select DSTIP addresses, the representation of the unique SRCIP addresses found in a specific subnet was almost proportional, i.e. the number of unique SRCIP addresses observed in a specific subnet was proportionally approximately the same. No subnet of the same size contained more unique SRCIP addresses than its counterparts. The same observation is seen in this chapter when looking at CI i.e. as the sample size of DSTIP addresses taken from the /24 IPv4 baseline data to create a bootstrap sample increased, the CI increased by an equivalent proportion. Each time the sample size doubled, almost the same proportion was reflected in the CI. What this means is that whether the sample is randomly sampled or sequentially sampled, the proportionality of the unique SRCIP addresses between a subnet and its subnet equivalent is similar. Worth remembering is that bootstrapping does not work with sequential or sequential samples, however, the observation had to bring the scenario to complete the analysis on the data. This also shows how the analyses in this study are related to each other.

5.5.3 CI for Monthly Bootstrap Simulations

The study opted to extend the analysis by moving from quarterly observation to monthly observation. This is the case because the study had the hypothesis that the number of unique SRCIP addresses observed in the DSTIP addresses of the host's network telescope ought to be different based on the duration of observation and the size of the network telescope 'lens'. Thus during the analysis, a long observation period within the same month offered a high volume of unique SRCIPs contained in the pool of unique DSTIP addresses observed. In our study, a 30 minute observation interval showed few unique SRCIP addresses as compared to hourly observation. A day's observation offered more unique SRCIP observation as compared to an hourly observation. However, this pattern did not proceed when the study moved its observation to accommodate a month's worth of observation.

One month worth of observation showed more unique SRCIP observed per hour as com-

pared to three months worth of observation. This is shown in **Tables 5.7 - 5.12**. Another view is when the study compared individual datasets. For instance, **Tables 5.1** and **5.4** show CI for the average number of unique SRCIP addresses observed from January to March for network telescope **148/8**. This is for both parametric and non-parametric. On the other hand, **Tables 5.7** and **5.10** show the average number of SRCIP addresses observed in the same network telescope but this time only for the month of March. Note how the average number of unique SRCIP observed per hour has increased from what was seen in **Tables 5.1** and **5.4** as compared to what we have observed in **Tables 5.7** and **5.10**.

Table 5.7: 146/8-032021: CI for No. of Unique SRCIP/hour [Parametric]

	CI Level			
Bootstrap Sample	80%	90%	95%	99%
146/8 - $_e$24	[718 - 734]	[716 - 737]	[713 - 739]	[711 - 744]
146/8 - $_e$25	[358 - 367]	[357 - 368]	[356 - 370]	[354 - 371]
146/8 - $_e$26	[180 - 185]	[180 - 185]	[181 - 186]	[177 - 187]
146/8 - $_e$27	[89 - 92]	[89 - 93]	[89 - 93]	[88 - 93]
146/8 - /$_e$28	[44 -45]	[44 -46]	[43 - 46]	[43 - 46]

Table 5.8: 155/8-032021: CI for No. of Unique SRCIP/hour [Parametric]

	CI Level			
Bootstrap Sample	80%	90%	95%	99%
155/8 - /$_e$24	[748 - 767]	[746 - 770]	[743 - 773]	[737 - 775]
155/8 - /$_e$25	[369 - 378]	[368 - 379]	[366 - 380]	[364 - 383]
155/8 - /$_e$26	[186 - 190]	[185 - 191]	[183 - 190]	[183 - 192]
155/8 - /$_e$27	[93 - 96]	[93 - 96]	[92 - 96]	[92 - 98]
155/8 - /$_e$28	[47 -49]	[47 -49]	[47 -50]	[47 -50]

Table 5.9: 196-A/8-032021: CI for No. of Unique SRCIP/hour [Parametric]

	CI Level			
Bootstrap Sample	80%	90%	95%	99%
196-A/8 - /$_e$24	[789 - 807]	[786 - 811]	[782 - 813]	[778 - 819]
196-A/8 - /$_e$25	[395 - 405]	[393 - 406]	[392 - 407]	[390 - 409]
196-A/8 - /$_e$26	[195 - 200]	[194 - 201]	[193 - 202]	[192 - 202]
196-A/8 - /$_e$27	[96 - 99]	[96 - 100]	[95 - 100]	[95 - 100]
196-A/8 - /$_e$28	[49 -51]	[49 -51]	[49 -51]	[48 - 52]

This drop is in the number of unique SRCIP addresses observed in that time frame. The reason for such a drop is primarily due to the elimination of the redundant number of

unique SRCIP addresses observed, that have appeared in all three months. Thus by the time observation gets to the month of February or March in our three months time frame, some of the unique SRCIPs would have been seen already in January. If such unique SRCIP addresses are observed again in the subsequent month(s), they would not account as unique SRCIPs anymore. However, if each month is observed independently, as was the case with March, then those unique SRCIP addresses are accounted for, just for that month alone, hence the high volume in the average number of unique SRCIP addresses in **Tables 5.7 - 5.12** as compared to those observed in **Tables 5.1 - 5.6.**

Table 5.10: 146/8-032021: CI for No. of Unique SRCIP/hour [Non Parametric]

	CI Level			
Bootstrap Sample	80%	90%	95%	99%
146/8 - /e24	[571 - 582]	[569 - 583]	[568 - 585]	[565 - 588]
146/8 - /e25	[286 - 292]	[285 - 293]	[285 - 293]	[284 - 295]
146/8 - /e26	[144 - 146]	[143 - 147]	[142 - 148]	[142 - 148]
146/8 - /e27	[71 - 73]	[71 - 73]	[71 - 73]	[71 - 74]
146/8 - /e28	[35]	[35 -36]	[34 -36]	[34 -36]

Table 5.11: 155/8-032021: CI for No. of Unique SRCIP/hour [Non Parametric]

	CI Level			
Bootstrap Sample	80%	90%	95%	99%
155/8 - /e24	[599 - 610]	[597 - 612]	[595 - 614]	[594 - 617]
155/8 - /e25	[295 - 301]	[295 - 302]	[293 - 302]	[292 - 303]
155/8 - /e26	[148 - 151]	[148 - 152]	[147 - 153]	[147 - 153]
155/8 - /e27	[74 - 75]	[74 - 76]	[73 - 76]	[73 - 76]
155/8 - /e28	[37 -38]	[37 -38]	[36 -38]	[37 -39]

Table 5.12: 196-A/8-032021: CI for No. of Unique SRCIP/hour [Non Parametric]

	CI Level			
Bootstrap Sample	80%	90%	95%	99%
196-A/8 - /e24	[607 - 620]	[604 - 622]	[603 - 624]	[601 - 626]
196-A/8 - /e25	[306 - 312]	[305 - 313]	[304 - 314]	[303 - 316]
196-A/8 - /e26	[153 - 156]	[152 - 156]	[151 - 157]	[151 - 157]
196-A/8 - /e27	[76 - 77]	[75 - 78]	[75 - 78]	[75 - 78]
196-A/8 - /e28	[49 -51]	[49 -51]	[49 -51]	[48 -52]

5.5.4 Summary Statistics Non-Parametric Bootstrap Samples

Another set of experiments that the study did was to observe the variations that exist between the bootstrap samples and the baseline data when it comes to descriptive statistics. The study primarily focused on the *Standard error of mean* and the *mean* of both the baseline dataset and the bootstrap samples. Standard Error of mean (SEM) shows how accurate the estimate of the mean is likely to be, i.e. SEM measures how much discrepancy there is likely to be in a sample's mean compared to the population mean (Seabold and Perktold, 2010). SEM acknowledges that each diagnostic test has an inherent predictable amount of errors which always comes with the tests being carried out. Thus SEM provides a statement of probability about the difference between the mean of the population and the mean of the sample. All the results in this section are from the TCP dataset.

Table 5.13: Summary Statistics for 146/8 - [Jan - Mar] - No. of SRCIP/hour

Bootstrap Sample	Baseline Mean	Bootstrap Mean	Baseline SEM	Bootstrap SEM
146/8 - /$_e$24	473	474	1.62	1.63
146/8 - /$_e$25	238	238	0.85	0.86
146/8 - /$_e$26	120	120	0.48	0.49
146/8 - /$_e$27	60	60	0.29	0.29
146/8 - /$_e$28	30	30	0.16	0.17

Table 5.14: Summary Statistics for 196-A/8 - [Jan - Mar] - No. of SRCIP/hour

Bootstrap Sample	Baseline Mean	Bootstrap Mean	Baseline SEM	Bootstrap SEM
196-A/8 - /$_e$24	512	513	1.96	1.96
196-A/8 - /$_e$25	259	260	1.03	1.03
196-A/8 - /$_e$26	130	131	0.56	0.55
196-A/8 - /$_e$27	64	65	0.31	0.30
196-A/8 - /$_e$28	40	40	0.46	0.47

Table 5.15: Summary Statistics for 155/8 - [Jan - Mar] - No. of SRCIP/hour

Bootstrap Sample	Baseline Mean	Bootstrap Mean	Baseline SEM	Bootstrap SEM
155/8 - /$_e$24	488	489	1.70	1.71
155/8 - /$_e$25	244	244	0.87	0.87
155/8 - /$_e$26	123	124	0.48	0.49
155/8 - /$_e$27	61	62	0.27	0.28
155/8 - /$_e$28	31	30	0.17	0.17

Table 5.16: Summary Statistics for 146/8-032021 - No. of SRCIP/hour

Bootstrap Sample	Baseline Mean	Bootstrap Mean	Baseline SEM	Bootstrap SEM
146/8 - /e24	576	577	4.32	4.33
146/8 - /e25	289	290	2.2	2.2
146/8 - /e26	145	145	1.14	1.15
146/8 - /e27	72	72	0.62	0.62
146/8 - /e28	35	36	0.33	0.34

Table 5.17: Summary Statistics for 196-A/8-032021 - No. of SRCIP/hour

Bootstrap Sample	Baseline Mean	Bootstrap Mean	Baseline SEM	Bootstrap SEM
196-A/8 - /e24	613	614	5.11	5.11
196-A/8 - /e25	309	310	2.57	2.57
196-A/8 - /e26	149	155	1.29	1.30
196-A/8 - /e27	77	77	0.68	0.68
196-A/8 - /e28	38	38	0.39	0.40

Table 5.18: Summary Statistics for 155/8-032021 - No. of SRCIP/hour

Bootstrap Sample	Baseline Mean	Bootstrap Mean	Baseline SEM	Bootstrap SEM
155/8 - /e24	604	605	4.62	4.63
155/8 - /e25	298	298	2.26	2.27
155/8 - /e26	150	150	1.17	1.18
155/8 - /e27	75	75	0.64	0.64
155/8 - /e28	37	38	0.38	0.38

In this study, the population is the baseline datasets for the month of March and the baseline datasets collected from January to March. The samples are bootstrap samples that are generated from the subnet equivalents. What SEM essentially does is to account for these errors in the computation of CI to give a user an overview scenario of the times when the true parameter of interest falls outside the intended CI range computed at a specific level of certainty. Each table shows the average number of unique SRCIP per hour.

For instance, if the study uses 80% CI level to compute CI for telescope **148/8** for January to March, the SEM values will account for the 20% of the times that the average number of unique SRCIP addresses observed on an hourly basis failed to be found in the given CI range. This is to say one can either add or subtract from the lower value and the upper values of our range to increase it to accommodate such errors in our computation.

By adding and subtracting the SEM values to the computed CI range, it accounts for errors that come inherently with the data and the process of computation, in our case, bootstrapping. Small errors indicate small variations between the bootstrap sample and the baseline data.

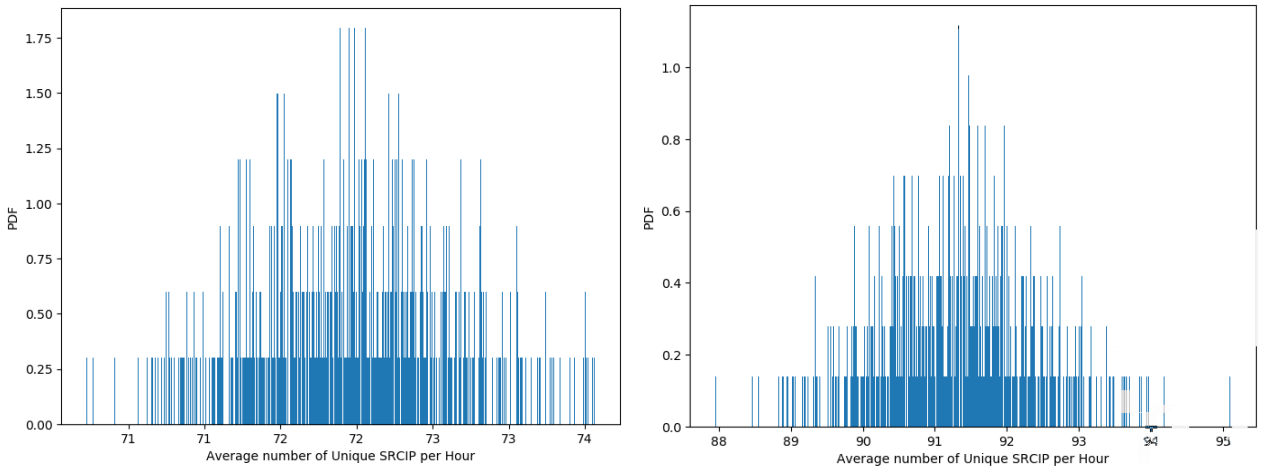
In view of this, the study acknowledged the need to have SEM computed for its bootstrapping of IBR data as it computed CI at different CI levels. **Tables 5.13 - 5.18** shows how each bootstrap sample performed when compared to the baseline bootstrap. This includes all observations made for both monthly observations as well as the three months time period. As the subnet equivalent size taken from the baseline data to create a bootstrap sample increased, SEM was increasing as well. This means that the bigger subnet equivalent samples created better bootstrap samples which were more representative of the overall baseline bootstrap, thus the SEM got closer to the baseline bootstrap.

Since the study was also looking for similarities between the baseline bootstrap samples and the subnet equivalent bootstrap samples, then from each of the tables (**Tables 5.13 - 5.18**), note how the SEM for larger subnet equivalents formed bootstrap samples that were more closer to the baseline bootstrap samples. This is important because in aiming to reproduce the datasets, it is imperative that one accommodates the inherent errors that come along with the data. With this in mind, it is safe to say that as the size of the subnet equivalent increased, the bootstrap samples generated from it resembled more of the baseline bootstrap sample. The values provided in each of these tables (**Tables 5.13 - 5.18**) show how the mean would vary with each given bootstrap sample with mean as the population parameter of interest.

Bootstrap samples generated from larger subnet equivalents offered more variation of the mean when SEM is accounted for i.e. bootstrap samples generated from large subnet equivalents had bigger errors than those generated from smaller subnet equivalents. It is for this reason that the CI range for larger samples shown in **Tables 5.10 - 5.11** are bigger than those in bootstrap samples generated from smaller subnet equivalents. Looking at the mean shown in **Tables 5.10 - 5.11** and the CI computed in **Sections 5.5.1** and **5.5.2**, IBR data has shown results that are more reflective of the baseline bootstrap when working with non-parametric bootstrapping than parametric bootstrap. As such, the study recommends the use of non-parametric bootstrapping for future studies. The results are also consistent with what is expected in any inferential computation when it comes to the value of SEM.

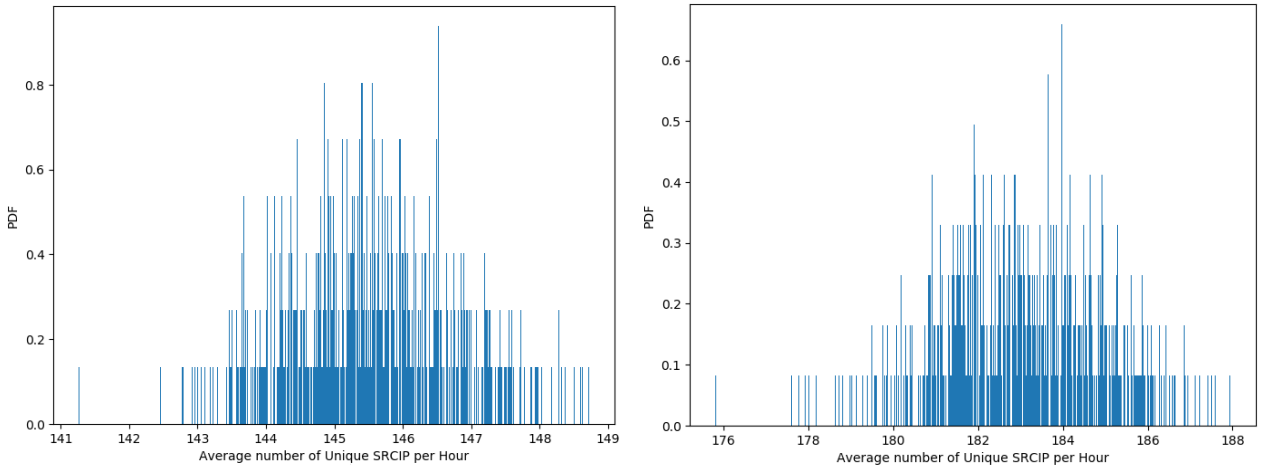
5.6 Graphical Representation of Bootstrap Samples

In this section, the study focused on showing the graphical representation of the bootstrap samples. **Figures 5.4 - 5.6** show CI plots computed from bootstrap samples that were generated from /$_e$27, /$_e$26 and /$_e$25 subnet equivalents from a **146/8** network telescope. The study opted to use one network telescope in all the subnet equivalents as a way of representing how the CI would look like if it were to be presented graphically. Considering that the results among all these network telescopes were similar, any random pick of the network telescopes would work as a way of demonstrating the results graphically. More plots regarding CI are appended in **Appendix G**.



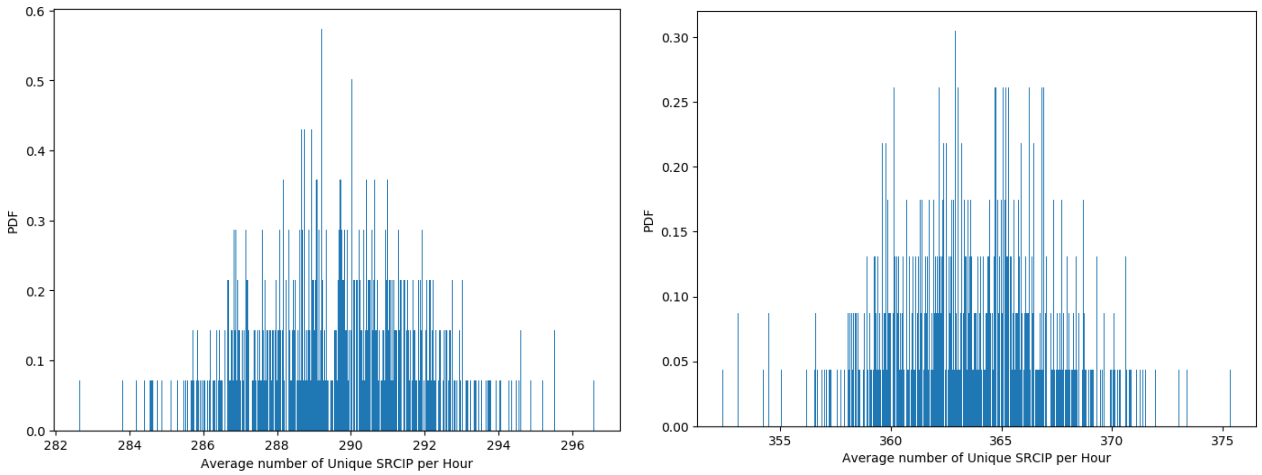
(a) Non - Parametric Bootstrap Sample (b) Parametric Bootstrap Sample

Figure 5.4: 146/8-032021: /$_e$27 Subnet equivalent Bootstrap Sample at 95% CI



(a) Non - Parametric Bootstrap Sample (b) Parametric Bootstrap Sample

Figure 5.5: 146/8-032021: /$_e$26 Subnet equivalent Bootstrap Sample at 99% CI



(a) Non - Parametric Bootstrap Sample

(b) Parametric Bootstrap Sample

Figure 5.6: 146/8-032021: $/_e25$ Subnet equivalent Bootstrap Sample at 95% CI

Each figure shows both parametric and non-parametric bootstrap plotted at 95% and 99% CI levels. These two CI levels were chosen because they offered a better representation of the average number of unique SRCIPs observed in the baseline datasets. Secondly, the two CI levels offered slight variations between them and thus can be used interchangeably. Lastly, they offered a wide range of CI from which the parameter of interest could be found (in our case, the average number of unique SRCIP addresses observed on an hourly basis).

What is shown in each of these plots is a range of potential values computed at different CI levels. Areas that look more concentrated than others show that most of the data points are concentrated there. For instance, in **Section 5.5** bootstrap samples were computed from a $/_e27$ subnet equivalent at 95% CI falls between **[71 - 44]** for non parametric bootstrapping while for parametric bootstrapping it lies between **[88 - 93]**. A look at **Figure 5.4**, shows that a majority of the data points fall within these ranges (**[71 - 44]** for non-parametric and **[88 - 93]** for parametric). The dispersion that is seen outside of this region goes to show that the other 5% could fall outside of the computed ranges of **[71 - 44]** and **[88 - 93]** as CI computation acknowledges the likelihood of the actual population parameter could fall outside of this region. i.e. the bars seen outside of this range account for the inherent errors that come with the data and bootstrap computation. Since CI gives a range of plausible values for a population parameter for any given random dataset, then from these two ranges one can safely say that they are 95% confident that the intervals **[71 - 44]** and **[88 - 93]** captured the true mean of the number of unique SRCIP addresses observed every hour within the $/_e27$ subnet equivalent bootstrap samples. In other words, this study is 95% confident that every hour that a network telescope that accommodates

32 unique DSTIP addresses is used within this IP address block, the average number of unique SRCIP addresses observed by those unique DSTIP addresses will lie between **[71 - 44]** for non-parametric and **[88 - 93]** for parametric respectively.

Using this basis, one can also extend for bootstrap samples that were computed from larger subnet equivalent samples. For instance, **Figure 5.5** shows a bootstrap sample plot computed at 99% CI with a majority of the data points concentrated around the range **[142 - 148]** for non-parametric bootstrap samples and **[177 - 187]** for parametric bootstrap samples. This is to say that this study is 99% confident that the average amount of unique SRCIP addresses registered per hour within a network telescope, configured to accommodate 64 unique DSTIP addresses, lies within the range **[142 - 148]** for non-parametric bootstrap samples. If more unique DSTIP addresses are added to the network, let's say 128 (See **Figure 5.6**), then with such values (coupled with the level of CI given), it is easy to come up with the range with which the allocated network telescope would be able to collect, on average, the number of unique SRCIP addresses registered on an hourly basis. It was in the range of **[285 - 293]** for a bootstrap sample computed at 95% CI for a subnet equivalent size of 128.

It is worth noting that these values are not always going to be the same for different networks. The volume of the number of unique SRCIP addresses registered within the network telescope is going to vary depending on the location of the network, the type of threat targeting an organisation or surge of threats on the network, the nature of business the organisation is involved in, and how persistent the threat actors are in getting into an organisation's network. However, the procedure followed here does not apply to every network telescope irrespective of the aforementioned reasons. If followed thoroughly with the appropriate CI levels, an organisation will be able to know with a certain degree of confidence how many unique SRCIP addresses they should have had if they had configured a bigger 'lens' network telescope.

Worth mentioning here is that the sizes of the subnet equivalents do not necessarily have to be identical to well-known subnets. One can choose any random number of unique DSTIP addresses and use that to compute what is missing in the reader's network. For instance, an organisation can compute for 100 unique DSTIP addresses, or 10, or 20. Whatever value, the user will still be able to get the answers they need. Secondly, in this study, it was observed that the average values of non-parametric bootstrapping resonated very well with those observed in the baseline bootstrap sample. Thus the study recommends the use of non-parametric bootstrapping over parametric bootstrapping, specifically because a user has more control over the variables in non-parametric bootstrapping than in parametric

bootstrapping. Note also that most of the non-parametric bootstrapping plots do follow a normal distribution curve, an important feature that has to be taken into consideration when computing CI.

5.7 Recommendations

This chapter has proven that bootstrapping can indeed be used to simulate IBR data to fill in the void left by those who do not have adequate resources to afford larger network telescopes. IBR data has shown results that were more reflective of the baseline bootstrap when working with non-parametric bootstrapping than parametric bootstrap thus the study recommends the use of non-parametric bootstrapping for future studies. The study has also confirmed observations made by Hesterberg (2015) in his study as mentioned in **Section 3.4**. The results are also consistent with what is expected in any inferential computation when it comes to the value of SEM.

The study also recommends the use of the data at least at 95% CI level because the variations between 95% and 99% CI level are not big, this way, there will be a high chance of reflecting the reflections observed in the baseline data. A 95% or 99% CI for any given sample size means that the data user is 95% or 99% confident that each of the allocated sample will contain a specific average number of unique SRCIP addresses in the available pool of DSTIP addresses. Confident to a point of knowing how much is not accounted for if the number observed are anything lower or higher than the indicated interval presented per network telescope size. The average number of unique SRCIP addresses observed in the DSTIP addresses of the host's network telescope is going to be different based on the duration of observation and the size of the network telescope. However, the proportions of the averages is going to be roughly the same depending on the SEM value of each sample. Thus given the % range of a CI per bootstrap sample, one should be able to compute the average number of unique SRCIP addresses observed in the DSTIP addresses of the host's network telescope. **Tables 5.19** and **5.20** presents the thesis's artefact to be used as guide in computing the averages per given sample at 95% CI.

The SEM in **Tables 5.19** and **5.20** presents the margin of error in the computation of the *mean* for each bootstrap sample. These were averaged for monthly and quarterly datasets to ensure fair representation. If one can compute the margin of error in their IBR data the CI range will be computed. To explain the tables, let us use bootstrap sample size **128**

- /e25 in **Table 5.19**. It shows the range from 47.83% to 52.51% that is, 50.17% plus or minus 2.34 percentage points. The researchers are confident that if another network telescope is used to collect average number of unique SRCIP using bootstrap sample size **128 - /e25** on monthly basis, then 95% of the time - or 19 times out of 20 - the findings would fall in this range. Using the percentage range gives the user the actual range of average number of unique SRCIP observed per hour for the sizes presented in **Tables 5.19** and **5.20**.

Longer observation period offer more precision in the CI range as the SEM is lower in quarterly analysis than in monthly datasets (see **Tables 5.19** and **5.20**). In addition to this longer observation periods offer high volume of unique SRCIPs contained in the pool of unique DSTIP addresses observed. This is to say that if a 30 minute observation interval showed a few unique SRCIP addresses as compared to hourly observation, then a monthly interval would show more unique sources than a weekly observation. The study recommends observation of longer periods as that accommodates more unique SRCIP per sample. Large network telescopes still contain more unique SRCIP addresses as compared to small ones so a network telescope with 32 DSTIP addresses will show less unique SRCIP addresses than a 128 network telescope. Thus, the study recommends bootstrapping larger samples as compared to smaller because bigger samples created better bootstrap samples which were more representative of the overall baseline bootstrap, thus the SEM got closer to the baseline bootstrap. In the context of this research, larger network telescopes showed more unique SRCIPs per give pool of DSTIP than smaller ones.

Table 5.19: Monthly Summary Table for CI in Percentage at 95% CI

Bootstrap Size	Avg. Bootstrap Mean %	Average SEM	% CI
256 - /e24	100.00	4.68	[95.32 - 104.68]
128 - /e25	50.17	2.34	[47.83 - 52.51]
64 - /e26	25.17	1.20	[23.97 - 26.37]
32 - /e27	12.50	0.64	[11.86 - 13.14]
16 - /e28	6.21	0.37	[5.75 - 6.49]

Table 5.20: Quarterly Summary Table for CI in Percentage at 95% CI

Bootstrap Size	Avg. Bootstrap Mean %	Average SEM	% CI
256 - /e24	100.00	1.76	[98.24 - 101.76]
128 - /e25	50.30	0.92	[49.38 - 51.22]
64 - /e26	25.45	0.51	[24.94 - 25.96]
32 - /e27	12.69	0.29	[12.40 - 12.98]
16 - /e28	6.78	0.27	[6.51 - 7.05]

5.8 Summary

This chapter explored bootstrapping as a statistical technique that can be used to simulate samples of IBR datasets with the aim of estimating the likelihood of finding missing unique SRCIP addresses that are currently present within a subnet equivalent. This chapter began by justifying why bootstrapping is needed for this study in **Section 5.1**. This section outlines how bootstrapping fits the objectives of this study and how it can help to attain these research goals. From here a need to define how the study will approach bootstrapping in order to achieve its objectives was needed, thus a laid out plan of how the study was conducted is explained in **Section 5.2**. This was immediately followed by the techniques of bootstrapping that were used to process the data. It is in **Section 5.3** that the two techniques of bootstrapping were used. More details about these two techniques were presented in **Section 3.2**.

This chapter also looked at the relationship that exists between the number of unique SRCIP and time. This relationship and its findings were presented in **Section 5.4**. From here on the study aimed to establish the confidence levels associated with each bootstrap sample. The study has proven that bootstrapping can indeed be used to simulate IBR data to fill in the void left by those who do not have adequate resources to afford a larger network telescope 'lens'. These findings are presented in **Section 5.5**. The study also presented a graphical representation of these CI computed and explained how they relate to the tables presented. This is shown in **Section 5.6**. Lastly, IBR data has shown results that were more reflective of the baseline bootstrap when working with non-parametric bootstrapping than with parametric bootstrap. As such, the study presents artefacts and recommendations to the reader in **Section 5.7**

6

Quantifying Variations in IBR Samples

This chapter focuses on the computation of the differences that exist between the baseline datasets and their samples, be it random or sequential. The chapter builds on the knowledge introduced and explained in **Sections 3.8**. Building on the work by Hyndman and Koehler, the chapter begins by introducing the mathematical models that have been derived to compute the differences that exist between baseline datasets and their samples in **Section 6.1**. This is followed by the research approach that was used to pre-process the data in **Section 6.2**. It is in this section that the two sampling techniques that were used to sample IBR data are explained. This is immediately followed by an evaluation of the derived models against MAPE, SMAPE, MAE and MASE in **Section 6.3**.

Having established a benchmark with which to work on, and having the models validated, the study shifted its focus to assess the performance of the models on random and sequential datasets. This is shown in **Section 6.4**. Recommendations on DSTIP monitoring and placement are presented in **Section 6.5**.This section led to the assessment of the feasibility of sampling IBR data in **Section 6.6**. An analysis of the overall performance of the models on IBR data is explained in **Section 6.7**. It was at this point that the

study had to look at the strengths and limitations of the developed models. This is explained in **Section 6.8**. The impact of sampling on destination ports using information retrieval techniques is explained in **Section 6.9**. Like in the preceding sections, this section explores two case studies: monthly and quarterly analysis. The chapter closes with a summary in **Section 6.10**. The practical applications of the models developed are discussed further in **Chapter 7**.

6.1 Mathematical Models Developed for IBR Datasets

To have a clear understanding of how the models came about, the reader is required to understand the literature review presented in **Section 3.9** which forms the core from which all the mathematical models presented in this section are based. A clear understanding of **Section 3.9** is assumed when presenting the models found in this chapter. Three major differences found between the models here and those developed by Hyndman and Koehler (2006) lie in the usability of the model, the data required to use the model, and what is being measured. Hyndman and Koehler (2006) designed MAPE, SMAPE, MAE, and MASE models to measure the errors found in forecasting time series data. Thus the core purpose of his models was forecasting, while on the other hand,the models presented in this chapter are designed to measure differences that exist between data samples based on a specified unit of standardisation. Standardisation is an integral element in the models developed in this chapter without which the model becomes unusable or gives erroneous results.

Secondly, Hyndman and Koehler's models were specifically designed to work with time-series data, hence the concept of forecasting which cannot happen without the data having time stamps. On the other hand, the models in this chapter work with both time-series data and data without a timestamp. However, the models require that the data samples being compared have a series of data points and their order does not really matter. Lastly, Hyndman and Koehler were more interested in the errors found between two or more time-series while, in this case, the study was more interested in the representativeness of one sample to the next. The focus was on how accurate can one sample compare to the next. More specifically, how accurately can subnets and subnet equivalents represent the baseline dataset from which they were drawn? The understanding of these differences will help the reader to see how the models developed fit in this study. With this knowledge and understanding, the study will present the models one at a time and show how they are used here.

6.1.1 Absolute Mean Accuracy Percentage Score (AMAPS)

Absolute Mean Accuracy Percentage Score (AMAPS) is a measure of an average absolute percentage score. AMAPS computes the level of accuracy (representativeness) of a subnet or subnet equivalent data sample to the actual (baseline) dataset as a score measured in percentage. It has to be absolute because this study was not interested in the direction of the difference. As such, a negative value coming out of the computation will negatively affect the score computed, hence the absolute mean. In all the cases, the baseline data was from /24 IPv4 subnet values taken from all the network telescopes. The subnets and subnet equivalents were taken from the same baseline dataset to form comparable samples. In this study, subnets and subnet equivalents ranged from .128/25 to .252/30 for subnets and $/_e 25$ *to* $/_e 30$ for subnet equivalents. This way, the study was able to compute the gap that exists between the baseline study and the subnets and subnet equivalents. For the remainder of this chapter that is how actual subnet (for sequential sampling) and subnet equivalent (for random sampling) have been defined.

Let $\boldsymbol{A_t}$ and $\boldsymbol{S_t}$ denote the baseline and subnet equivalent sample values of the same baseline at data point **t** respectively. If one is working with sequential sampling, $\boldsymbol{S_t}$ denote subnet equivalent sample values of the same baseline at data point $\boldsymbol{t}$ respectively. Let $\boldsymbol{a}$ and $\boldsymbol{s}$ denote the size of the baseline and subnet sample (respectively) being evaluated. $\boldsymbol{a}$ and $\boldsymbol{s}$ are the values that a model user need to define in order to normalise the values contained in $\boldsymbol{A_t}$ and $\boldsymbol{S_t}$. $\boldsymbol{t}$ in all time-series data denotes time. If a researcher is working with data that has no time stamps on it, **t** becomes the position of the data points within the datasets that one is working with. Considering that the study was interested in the accuracy score, a perfect score would be a **one** while a poor score will be a **zero**. Thus to find the accuracy score, one will need to first compute the error found between the baseline data sample and the subnet sample being investigated. The computer error will then be subtracted from **one**. Since the score is measured in percentage, the value from this computation has to be multiplied by 100. This is presented in **Equation 6.1**.

High values of AMAPS indicate a better representation of the subnet or subnet equivalent with the baseline, i.e. the higher the absolute mean accuracy percentage score, the better one's sample is at representing the baseline dataset. The opposite of this score also applies, low scores are indicative of poor representation of the subnet or subnet equivalent to represent a baseline dataset. Thus high AMAPS values in our study are proof of how the subnet (or subnet equivalent) under study is closer to the baseline dataset (/24 IPv4

subnet). AMAPS is defined as:

$$AMAPS = \left[1 - \left(\frac{1}{N}\sum_{t=1}^{N}\frac{\left|\left(\frac{A_t}{a}\right) - \left(\frac{S_t}{s}\right)\right|}{\frac{A_t}{a}}\right)\right] \times 100 \tag{6.1}$$

6.1.2 Symmetric Absolute Mean Accuracy Percentage Score

Symmetric Absolute Mean Accuracy Percentage Score (SAMAPS) is an alternative to AMAPS when there is zero or near-zero demand for items as expressed by Hyndman and Koehler in their original model of SMAPE. Let $\boldsymbol{A_t}$ and $\boldsymbol{S_t}$ denote the baseline and subnet equivalent sample values of the same baseline at data point **t** respectively. If one is working with sequential sampling, $\boldsymbol{S_t}$ denote subnet sample values of the same baseline at data point $\boldsymbol{t}$ respectively. Let $\boldsymbol{a}$ and $\boldsymbol{s}$ denote the size of the baseline and subnet samples being evaluated. $\boldsymbol{a}$ and $\boldsymbol{s}$ are the values that the model user needs to define in order to normalise the values contained in $\boldsymbol{A_t}$ and $\boldsymbol{S_t}$. In contrast to AMAPS, SAMAPS has both a lower bound and an upper bound. This symmetrical nature of SAMAPS gives it a higher level of accuracy in its computational value than AMAPS. Just like SMAPE delimits to an error rate of 200% in order to reduce the influence of low volume items (Kim and Kim, 2016; Franses, 2016), so does SAMAPS. Low volume items are problematic because they could otherwise have infinitely high error rates that skew the overall error rate (Hyndman and Koehler, 2006), which in turn affect the level of accuracy.

The interpretation is similar to that of AMAPS since they are all percentage-based. Thus to find the accuracy score, one will need to first compute the error found between the baseline data sample and the subnet sample being investigated, and then subtract that value from one. Since the score is measured in percentage, the value from this computation has to be multiplied by 100. Thus, SAMAPS is computed by computing the error from the normalised baseline data sample of /24 IPv4 minus the normalised values of subnet and subnet equivalent values divided by the sum of baseline value and subnet equivalent values as expressed in **Equation 6.2**:

$$SAMAPS = \left[1 - \left(\frac{2}{N}\sum_{t=1}^{N}\frac{\left|\left(\frac{A_t}{a}\right) - \left(\frac{S_t}{s}\right)\right|}{\left(\frac{A_t}{a}\right) + \left(\frac{S_t}{s}\right)}\right)\right] \times 100 \tag{6.2}$$

6.1.3 Standardised Mean Absolute Error (SMAE)

Standardised Mean Absolute Error (SMAE) measures the average magnitude of the errors between normalised baseline data samples of /24 IPv4 and the normalised values of subnet and subnet equivalent values that have equal weight just as expressed by (Varouchakis and Hristopulos, 2013). The key major difference here is the standardisation which is critical as the study will later show in results. In this expression, like in **Equation 6.2**, $\boldsymbol{A_t}$ and $\boldsymbol{S_t}$ denote the baseline and subnet equivalent sample values of the same baseline at data point **t**respectively. Let $\boldsymbol{a}$ and $\boldsymbol{s}$ denote the size of the baseline and subnet samples being evaluated. $\boldsymbol{a}$ and $\boldsymbol{s}$ are the values that a model user needs to define in order to normalise the values contained in $\boldsymbol{A_t}$ and $\boldsymbol{S_t}$. When measuring the magnitude of errors, SMAE does not consider the direction of the set pairs under observation, hence the *absolute* (| |) expression added to the mathematical model, just as explained by Willmott and Matsuura (2005). SMAE has negatively-oriented scores, meaning that lower values are better than higher values. The smaller the standardised mean absolute error, the closer the subnet (or subnet equivalent) under study is to the baseline dataset (/24 IPv4 subnet). Thus smaller values in our study are proof of how the subnet sample is closer to /24 IPv4 subnet. Like in AMAPS and SAMAPS, if a researcher is working with data that has no time stamps on it, **t** becomes the position of the data points within the datasets that one is working with. SMAE is mathematically expressed as **Equation 6.3**:

$$SMAE = \frac{1}{N}\sum_{t=1}^{N}\left|\left(\frac{A_t}{a}\right) - \left(\frac{S_t}{s}\right)\right| \tag{6.3}$$

6.1.4 Standardised Mean Absolute Scaled Error (SMASE)

Standardised Mean Absolute Scaled Error (SMASE) is unlike the other two quantifying techniques, SMASE uses a scaling error based technique instead of a relative measure just as expressed in MASE by Hyndman and Koehler (2006). SMASE can only be computed when there are multiple data samples (time series or otherwise) to compute against each other. SMASE uses a scale based on the in-sample SMAE as shown in **Equation 6.4**, which is independent of the scale of the data. The scale makes SMASE less sensitive to outliers and easy to interpret and use in the same lines as SAMAPS or AMAPS. According to Hyndman and Koehler, a scaled error is less than one if it arises from a better forecast than the average one-step naive forecast computed in-sample. On

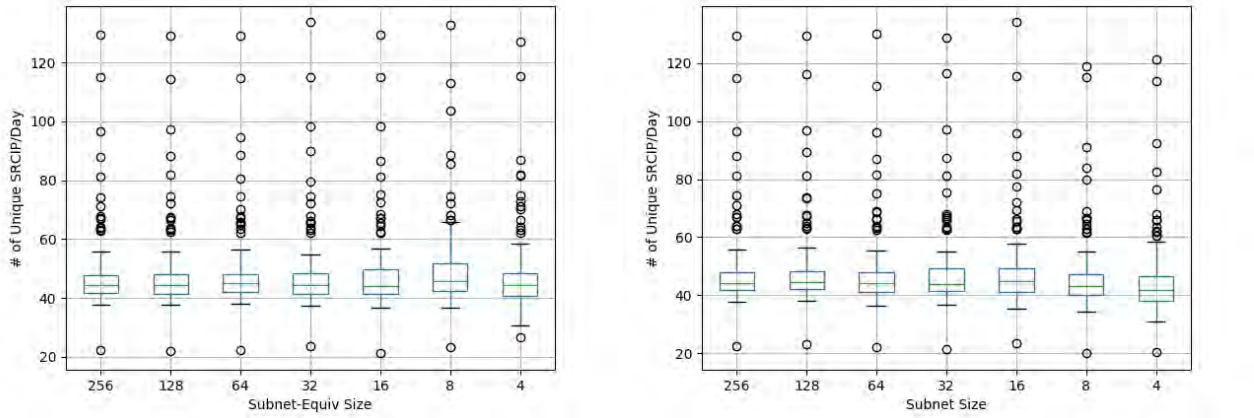
the other hand, if the forecast is worse than the average one-step naive forecast computed in-sample, then it is greater than one. Bringing it into our context, if the sample drawn from */24 IPv4 subnet* is better aligned to the original /24 IPv4, then the value will be less than one. Also, if there are significant differences, then the value will be higher than one. Thus, values of SMASE that are less than one are ideal to assess the representativeness of subnet or subnet equivalent in the place of a baseline dataset. In this expression, like in **Equation 6.2**, $\boldsymbol{A_t}$ and $\boldsymbol{S_t}$ denote the baseline and subnet equivalent sample values of the same baseline at data point **t** respectively. Let $\boldsymbol{a}$ and $\boldsymbol{s}$ denote the size of the baseline and subnet samples being evaluated. $\boldsymbol{a}$ and $\boldsymbol{s}$ are the values that a model user needs to define in order to normalise the values contained in $\boldsymbol{A_t}$ and $\boldsymbol{S_t}$. In **equation 6.5**, $\text{SMAE}_{\text{in-sample, naive}}$ is the standardised mean absolute error produced by a naive subnet sample.

$$SMAE_{in-sample,naive} = \frac{1}{N-1}\sum_{t=2}^{N}\left|\left(\frac{A_t}{a}\right) - \left(\frac{A_{t-1}}{a}\right)\right| \tag{6.4}$$

$$MMASE = \frac{SMAE}{SMAE_{in-sample,naive}} \tag{6.5}$$

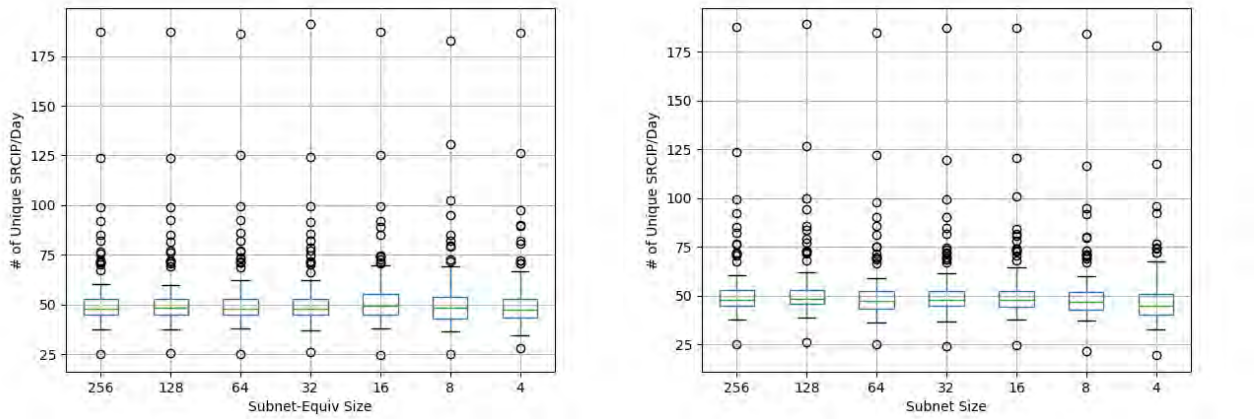
6.2 Research Approach in Data Analysis

This study did experiments on random and sequential datasets. **Figures 6.1, 6.2** and **6.3** show the data summary of the number of unique SRCIP addresses present in both the sequential and random data samples compared per subnet and subnet equivalent. All of these box plots are normalised for comparability i.e. each data from the subnet and subnet equivalents was divided by its corresponding size before plotting. The raw box plots (not normalised box plots) are found in **Appendix H**. This is for all three network telescope datasets used in this study with their data spanning from January to March 2021. On each plot, **Figures 6.1a, 6.2a** and **6.3a**, show the samples of random dataset while **Figures 6.1b, 6.2b** and **6.3b** show the sequential samples from the same datasets. For random samples, 10 random draws were made for each sample size as explained in Section **4.1.2**. One the other hand, for sequential sampling one sample was taken for each sample size as DSTIPs showed equitable distribution of unique SRCIPs. The reason for this approach and the exploratory work is presented in **Section 4.2.3**.



(a) Random Samples (b) Sequential Samples

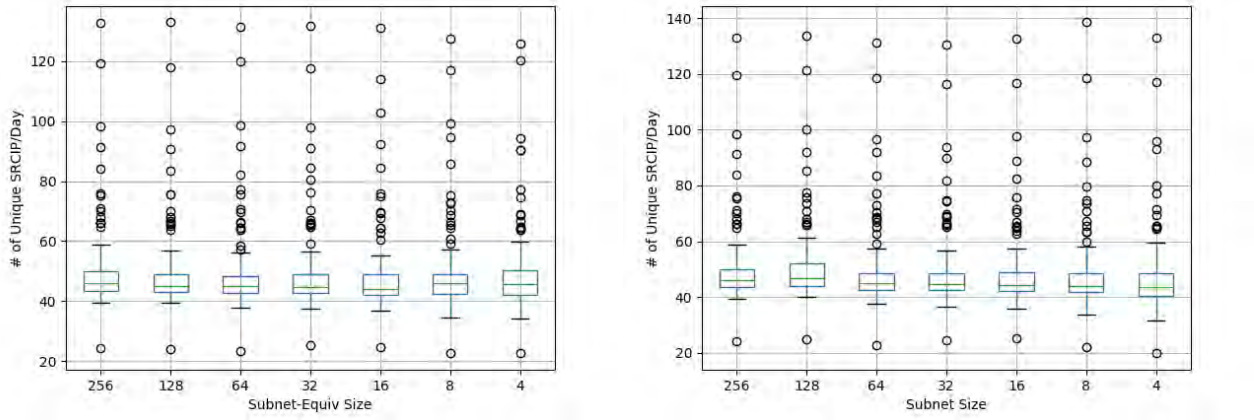
Figure 6.1: 146/8 -[Jan - Mar]: Data Summary of Unique SRCIP addresses/Day



(a) Random Samples (b) Sequential Samples

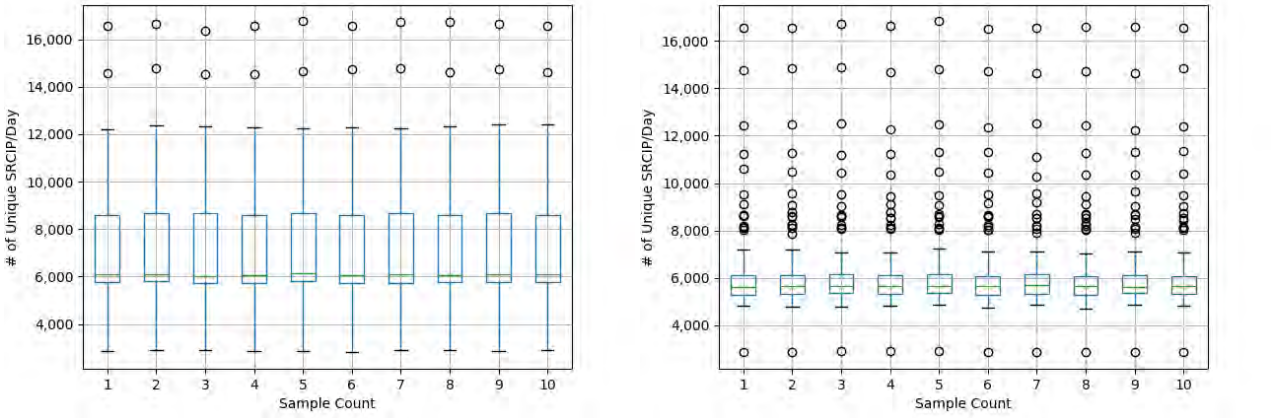
Figure 6.2: 196-A/8 - [Jan - Mar]: Data Summary of Unique SRCIP addresses/Day

What is common in all of these three box plots is the presence of outliers, which are indicative of the fact that the unique DSTIP addresses never received a uniform volume of unique SRCIP addresses. If that was the case the outliers would not be present. When it comes to the presence of outliers, there are no significant differences in each of the samples despite the fact that they were sampled using two different techniques. The DSTIP addresses that received the smallest amount of unique SRCIP addresses got as low as 20 unique SRCIPs in a three months period while those that received the most got as high as 187 unique SRCIPs. The sequential data were sampled based on the subnets that are well defined in the IP address blocks.



(a) Random Samples (b) Sequential Samples

Figure 6.3: 155/8 -[Jan - Mar]: Data Summary of Unique SRCIP addresses/Day



(a) January (b) [January - March]

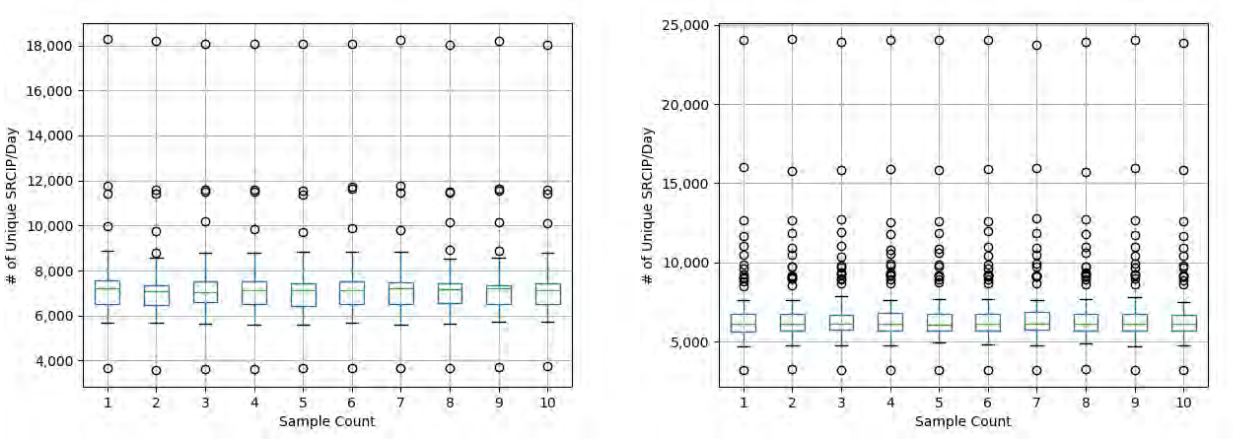
Figure 6.4: 146/8 -10 Random Sample Draws of Unique SRCIP/Day for $/_e26$ Subnet

Thus the default baseline dataset was a /24 subnet which contained all the 256 unique DSTIP addresses. Following this was a .128/25 subnet sample which contained 128 unique DSTIP addresses. This formed the first half of the unique DSTIP addresses in a /24 IPv4 subnet. Then the next sample contained 64 unique DSTIP addresses which formed a .192/26 subnet. This formed the third quarter of the last octet. **Table I.1** found in **Appendix I** shows the sequential sampling net-mask used.

Table 6.1: IP Address CIDR Network References for Sequential Sampling

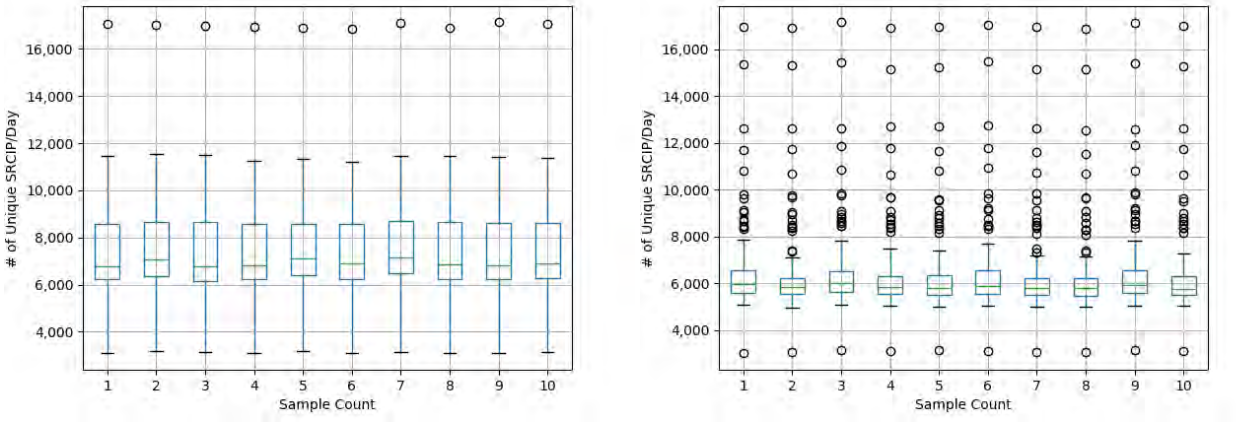
subnet	Subnet Mask	No. of DSTIP
/24	255.255.255.0	256
/25	255.255.255.128	128
/26	255.255.255.192	64
/27	255.255.255.224	32
/28	255.255.255.240	16
/29	255.255.255.248	8
/30	255.255.255.252	4

Table 6.1 shows the subnet mask used, the number of allocated DSTIP addresses per CIDR and the number of usable DSTIP addresses. It is the CIDR and Number of DSTIP address columns that the reader should focus on to understand the sample sizes used in both sequential and random sampling. On the other hand, the randomly sampled data was categorised into subnet equivalents as explained in **Section 4.1.2**. To make a single subnet equivalent, 10 randomly sampled draws were drawn from /24 net-block to come up with a new dataset for evaluation. This applies to $/_e25$ subnet equivalent to $/_e30$ subnet equivalent. An average of the unique SRCIPs per day in all the 10 samples was calculated to have a well represented sample. 10 samples offered a good representation of what could constitute a good a sample.



(a) February (b) [January - March]

Figure 6.5: 196-A/8 - 10 Random Sample Draws of Unique SRCIP/Day for $/_e26$ Subnet



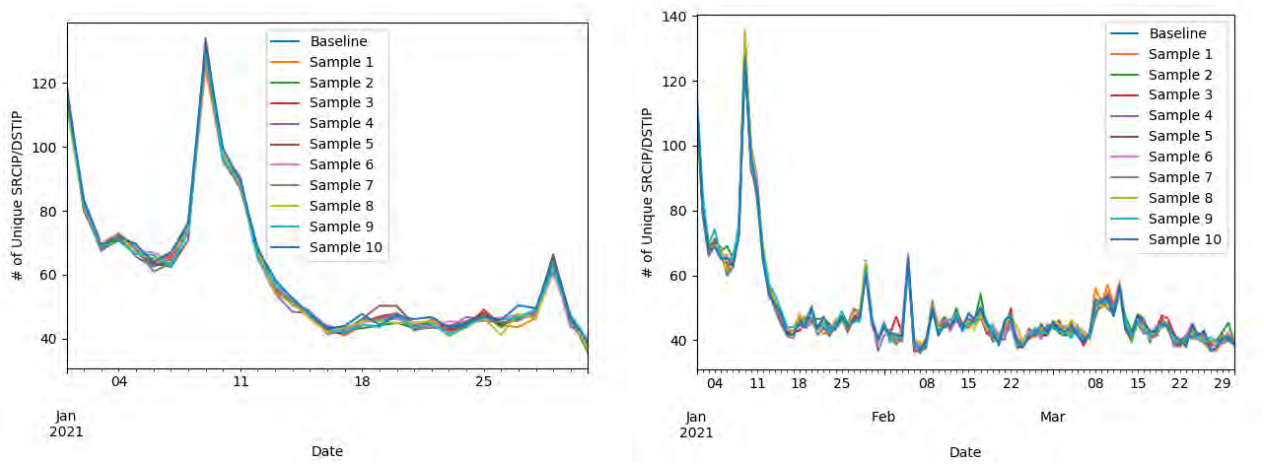
(a) March (b) [January - March]

Figure 6.6: 155/8 - 10 Random Sample Draws of Unique SRCIP/Day for $/_e26$ Subnet

What is critical from the box plots is to note the minimum and maximum values of the average number of unique SRCIP addresses observed per day per DSTIP address, which is shown on the *Y-axis* of the box plots. The minimum and the maximum values are shown by the ends of the whiskers. The *X-axis* of each of the box plots (**Figures 6.4, 6.5** and **6.6)** show the number of sample names taken to come up with a randomly sampled dataset. The average for each day from each of the 10 samples is what constituted the final dataset from a specific subnet. **Figures 6.4, 6.5** and **6.6**, show how each of the random draws compared to each other within a subnet equivalent.

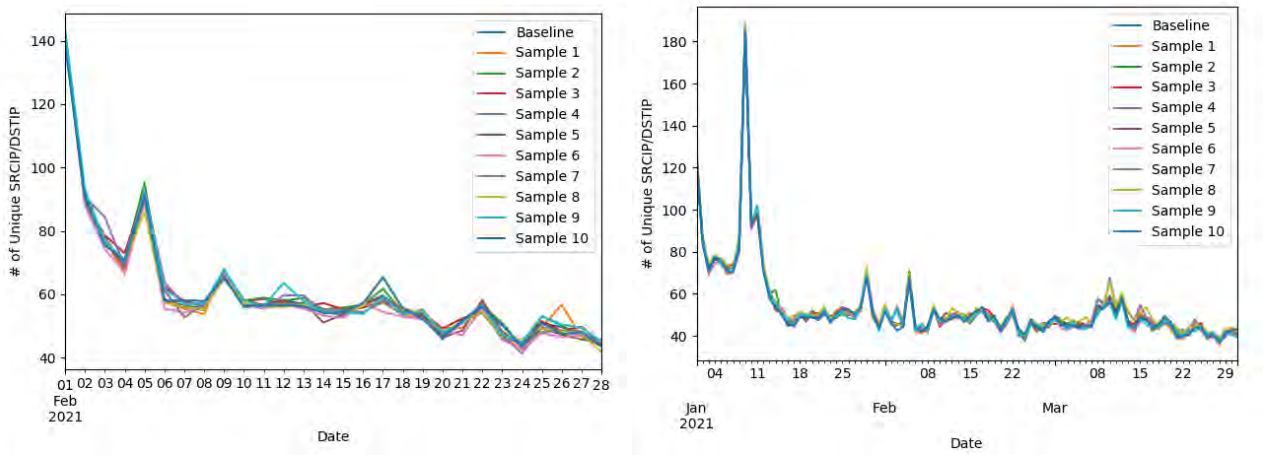
From **Figures 6.4, 6.5** and **6.6**, the reader can tell that there are not significant differences within the samples belonging to the same subnet equivalent. Each of these sample plots (**Figures 6.4, 6.5** and **6.6**) belong to $/_e26$ subnet equivalent. The data contained in each of the samples were normalised before computation and plotting to ensure that like terms are measured and to prevent unnecessary inconsistencies that come with data that is not normalised. More importantly, normalisation allows comparability of the samples from different subnet equivalents.

Since the study was interested in the activities observed per DSTIP address, to normalise the datasets, the number of unique SRCIP addresses observed on each day was divided by the number of unique DSTIP addresses contained in their respective subnet equivalent to come up with a *normalised subnet equivalent*. In other words, each subnet equivalent dataset was normalised using an actual subnet size that matches with its subnet equivalent to ensure comparability as explained in **Section 6.1.**



(a) January (b) [January - March]

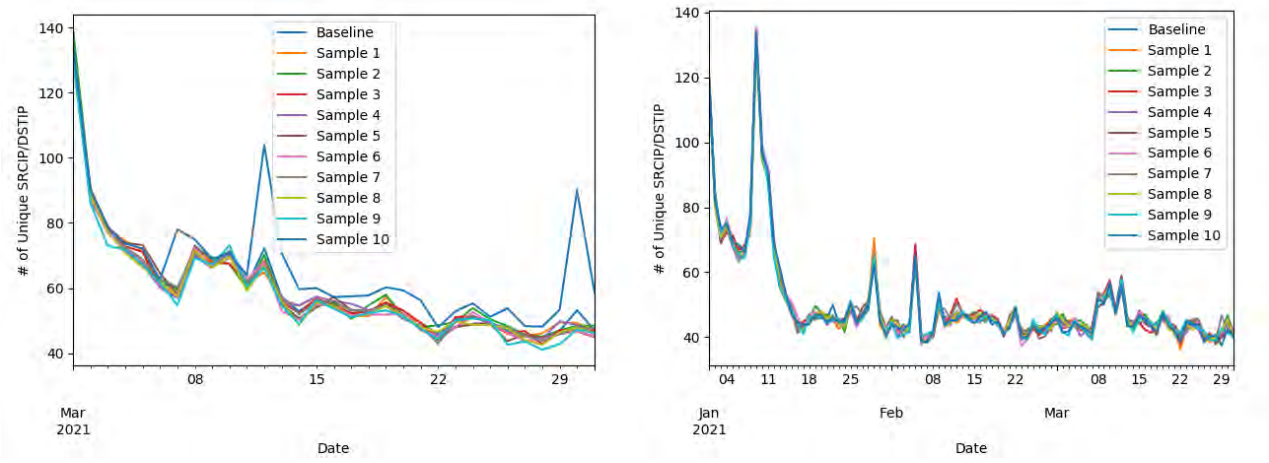
Figure 6.7: 146/8 -Time Series Plot of Unique SRCIP/DSTIP for /$_e$27 Subnet



(a) February (b) [January - March]

Figure 6.8: 196-A/8 -Time Series Plot of Unique SRCIP/DSTIP for /$_e$27 Subnet

For instance, to compare /24 IPv4 dataset (which contains 256 unique destination hosts) with a /$_e$27 subnet equivalent dataset (which contains 32 randomly sampled destination hosts), one would need to divide the traffic contained in the destination hosts by their respective subnet sizes. This is to say that since a /$_e$27 subnet equivalent is expected to have 32 IP addresses, then to come up with a normalised /$_e$27 subnet equivalent dataset, 10 random draws were made from the /24 IPv4 addresses, where each draw contained 32 unique DSTIP addresses that were randomly sampled.



(a) March (b) [January - March]

Figure 6.9: 155/8 -Time Series Plot of Unique SRCIP/DSTIP for $/_e27$ Subnet

As mentioned earlier, for comparability purposes, each of the subnet equivalents was normalised by the number of unique DSTIP addresses present in each dataset. This approach ensured that calculations are done per individual DSTIP address. This applies to all subnet equivalents used in this study. **Figures 6.7, 6.8** and **6.9** show the effect on normalisation on the 10 random samples from $/_e27$ subnet equivalent when compared with /24 net-block sample (represented in the legend as baseline). One can see from **Figures 6.7, 6.8** and **6.9** how comparable each individual random sample is to the baseline sample in a time series plot. The significance of normalisation (which will be used interchangeably with standardisation) will prove vital to the computation of the variations that are present between different subnets and subnet equivalents. Normalisation is key to the performance of the developed models as one can only compare likeable terms.

To sum up this section, what is established here is that the variations of the ranges (the difference between the minimum and the maximum value) are consistent in all the 10 samples of each random dataset. The data distribution of one sample is comparable to the next samples, more so with /24 IPv4 subnet, within their respective datasets. This just shows how similar the random samples are when they were drawn. There is also a significant resemblance between random and sequential data samples when it comes to displaying the number of unique SRCIP observed per day, even more so when the computation was done within individual DSTIPs. This is later on shown in **Section 6.3** where the accuracy scores are computed.

6.3 Evaluation of the Developed Models Against MAPE, SMAPE, MAE, MASE

This section demonstrates how AMAPS, SAMAPS, SMASE and SMAE were evaluated against MAPE, SMAPE, MAE and MASE. Simultaneously, the study got to evaluate how the proposed approach to derive the models has had an effect on well-known scores of Root Mean Scaled Error (RMSE) and Mean Scaled Error (MSE). It is important to recall that AMAPS, SAMAPS, SMASE and SMAE were derived from Hyndman and Koehler's forecasting models. Thus before going further, this study would like to clarify a very critical point: there is nothing wrong with Hyndman and Koehler's models in forecasting time series data. However, as they are, the models are limited in that one cannot use them to quantify differences that exist between data samples of the existing time series, something that this research aimed to achieve. Thus, through the study and analysis, it was derived that it is possible to optimise the models to accommodate, not only time-series data, but also any other dataset that has a unit of measure with which to standardise it. Thus, Hyndman and Koehler's models have been included for two reasons: first to show and acknowledge where the idea was derived, and, more importantly, to show the variation that the developed models have brought to Hyndman and Koehler's models while working with the same data to achieve a different goal. It is from this benchmark that this chapter should be understood. Practical applications of the models are found in **Chapter 7**.

The study analysis of the model was split into two cases: **IBR Data I** (presented in **Section 6.3.1**) was categorised as all the data that was collected within a month's period in all the three network telescopes while **IBR Data II** (presented in **Section 6.3.2**) is all the data that was collected in a three months period. In this section, the study focused more on how the derived models performed against MAPE, SMAPE, MAE, MASE and how the scores in these models affect the values of RMSE and MSE. What this section aims to achieve is to evaluate if the modifications made to the mathematical models are significant to stand on their own. More importantly, this section will also show the important role that normalisation does in the derived mathematical models and how different the results would be if standardisation was not taken into account.

6.3.1 Case Study: IBR Data I

Table 6.2: 146/8-032021: Accuracy Scores of Unique SRCIP/DSTIP

Subnet	AMAPS	SAMAPS	SMASE	SMAE	RMSE	MSE
Equiv	Acc.(%)	Acc.(%)	Score	Score	Score	Score
/24	**100**	**100**	**0**	**0**	**0**	**0**
/e25	99.32	99.32	0.03	0.38	0.50	0.25
/e26	98.84	98.84	0.05	0.65	0.89	0.80
/e27	97.46	97.48	0.12	1.47	1.86	3.48
/e28	97.30	97.34	0.12	1.52	1.97	3.91
/e29	97.19	97.20	0.13	1.58	2.10	4.41
/e30	94.97	95.00	0.21	2.62	3.55	12.63

Table 6.3: 146/8-032021: Error Scores of Unique SRCIP/DSTIP

Subnet	MAPE	SMAPE	MASE	MAE	RMSE	MSE
Equiv	Error(%)	Error(%)	Score	Score	Score	Score
/24	**0**	**0**	**0**	**0**	**0**	**0**
/e25	50.05	66.75	2.47	7,568.78	8,098.23	6.55e+07
/e26	75.07	120.19	3.72	11,356.03	1,2157.77	1.47e+08
/e27	87.41	155.28	4.33	13,217.87	1,4145.88	2.00e+08
/e28	93.64	176.10	4.64	14,163.62	1,5160.90	2.29e+08
/e29	96.85	187.81	4.79	14,649.68	1,5682.15	2.45e+08
/e30	98.43	193.85	4.87	14,890.59	1,5941.07	2.54e+08

Table 6.4: 155/8-032021: Accuracy Scores of Unique SRCIP/DSTIP

Subnet	AMAPS	SAMAPS	SMASE	SMAE	RMSE	MSE
Equiv	Acc.(%)	Acc.(%)	Score	Score	Score	Score
/24	**100**	**100**	**0**	**0**	**0**	**0**
/e25	98.30	98.33	0.10	0.94	1.47	2.16
/e26	98.01	97.96	0.12	1.16	1.88	3.54
/e27	97.46	97.43	0.15	1.47	1.94	3.78
/e28	96.16	96.12	0.23	2.21	2.65	7.07
/e29	94.84	94.72	0.31	2.94	3.80	14.48
/e30	93.87	93.86	0.36	3.37	4.15	17.27

Table 6.5: 155/8-032021: Error Scores of Unique SRCIP/DSTIP

Subnet Equiv	**MAPE Error(%)**	**SMAPE Error(%)**	**MASE Score**	**MAE Score**	**RMSE Score**	**MSE Score**
/24	**0**	**0**	**0**	**0**	**0**	**0**
/$_e$25	49.33	65.50	3.15	7,463.28	7,808.65	6.09e+07
/$_e$25	75.43	121.13	4.82	11,412.31	11,922.27	1.42e+08
/$_e$27	87.62	155.95	5.60	13,256.84	13,849.39	1.91e+08
/$_e$28	93.80	176.65	5.99	14,183.18	14,810.50	2.19e+08
/$_e$29	96.93	188.12	6.20	14,661.90	15,313.30	2.34e+08
/$_e$30	98.45	193.90	6.29	14,892.71	15,557.52	2.42e+08

Having detailed the research approach and more data characteristics in **Section 6.2**, the study evaluated how the mathematical models presented in **Equations 3.1, 3.2, 3.3** and **3.5** (shown in **Section 3.9**) performed on IBR data and compared their results to the models presented in **Equations 6.1, 6.2, 6.3** and **6.5**. Noteworthy is how MAPE, SMAPE, MAE and MASE have been used in this particular study to compute the margin of error between /24 IPv4 and subsequent subnet equivalents . A reminder that /24 IPv4 represents the actual values while subsequent subnet equivalents represent the predicted values. Thus, the score shown represents the errors between the baseline data and subnet equivalents. The percentage error scores easily show how deviated a subnet equivalent is from the baseline dataset (/24 net-block).

Tables 6.2 and **6.4** show the accuracy scores computed from AMAPS, SAMAPS, SMASE and SMAE with data collected from datasets **146/8-032021** and **155/8-032021**. For representatives purposes, this research has used datasets from different months and different network telescopes. This is so as the study wanted to make sure that every dataset described in **Chapter 4** was used. With this approach, data gets to be viewed in its entirety. This approach also helped to show that the results were independent of the month or network telescope being evaluated since every dataset gets to be evaluated. Other subsections of this chapter will present different datasets other than **146/8-032021** and **155/8-032021**.

Tables 6.3 and **6.5** show the error score summaries computed from MAPE, SMAPE, MASE, and MAE. All the data presented in this subsection is randomly sampled and the results show *mean* scores of the 10 samples that were randomly drawn for each subnet equivalent. In this section, the baseline datasets are represented by /24 subnet for all tables. Subsequent samples derived from these baseline datasets (referred to subnet equivalents) are represented by /$_e$25 subnet equivalent - /$_e$30 subnet equivalent. Each time a specific subnet equivalent is used, it will be mentioned along with the table name

being referred to. This way it is easier to follow through. A perfect subnet equivalent that mimics the baseline dataset is supposed to get 100% for AMAPS and SAMAPS with a score of zero indicating the same for SMASE, SMAE, RMSE and MSE. As it can be seen from **Tables 6.2** and **6.4**, only /24 subnets have perfect scores followed by $/_e25$ subnet equivalent.

The same naming convention was used when working with MAPE, SMAPE, MASE, and MAE. This is shown in **Tables 6.3** and **6.5** where baseline is represent by /24 subnet while the subnet equivalents are represented by $/_e25$ subnet equivalent to $/_e30$ subnet equivalent. A perfect subnet equivalent that mimics the baseline dataset is supposed to get a 0% error score for MAPE, SMAPE, MASE and MAE. As it can be seen from **Tables 6.3** and **6.5**, only /24 subnets have perfect scores followed by $/_e25$ subnet equivalents. Without normalisation, it is clear to see why the errors got bigger when moving from $/_e25$ heading towards $/_e30$. $/_e25$ is a subnet equivalent that represents half of the unique SRCIP addresses present in the 128 IP addresses it contains.

Every score computed from SMAPE, MASE and MAE were off the desired range and these errors are not acceptable for any test. MASE and MAE accept ranges between 0 and 1 where 0 is the desired score while 1 is the maximum tolerable score. Without standardisation, it can be seen in **Tables 6.3** and **6.5** that all the subnet equivalents are exceeding the limits set. When the study accommodated RMSE and MSE scores, which are standard ways to measure the error of a model in predicting quantitative data and computing how close a regression line is to a set of points, it helps to know how these models are not meeting the needs for IBR data under study as they are extremely large and show a serious deviation from the baseline. On the other hand, when the same datasets were normalised (as shown **Tables 6.2** and **6.4**), the study observed better scores for all of the metrics under study including RMSE and MSE, whose values are closer to the baseline datasets in both tables.

High accuracy scores presented in **Tables 6.2** and **6.4** show that any of these random samples can be used for placement and the results collected per unique DSTIP will not show significant differences. Smaller samples give lower scores as compared to larger ones, indicating that large samples are still better than smaller samples. The only difference this time is that such differences have a value added to it. When normalised, the differences are masked. What the reader should be aiming for is for high accuracy scores. Thus, 98.30 % accuracy score is preferred as compared to 93.86 % i.e. a $/_e25$ is better than a $/_e30$. More importantly, this section has shown that the developed models give better scores than MAPE, SMAPE, MASE and MAE. The purpose of this section was to evaluate the

performance. When presenting the models in **Section 6.1**, acceptable scores for SMASE and SMAE were supposed to be below *one*.

6.3.2 Case Study: IBR Data II

Having looked at monthly data to see how AMAPS, SAMAPS, SMASE and SMAE compared against MAPE, SMAPE, MASE and MAE, the study opted to extend the duration of observation to three months to see if it would have an impact on the computed scores. This was done to ensure that the study does not make conclusions based on limited data. Thus in this section, the tables contain accuracy scores for the total number of unique SRCIP/DSTIP per sample. The data used was collected between January - March 2021. Error scores fitting the same description are also presented.

The naming convention for the baseline and the subnet equivalent is the same as that displayed in **Section 6.3.1**. One significant difference between the monthly datasets and those found in **Tables 6.6 - 6.9** is that the accuracy scores of AMAPS, SAMAPS, SMASE and SMAE have slightly gone down, meaning the level of accuracy has declined over the three months period. This decline in accuracy, however, is only reflected in $/_e25$ subnet equivalent while there has been a rise in accuracy for $/_e30$ subnet equivalent. On the other hand, MAPE, SMAPE, MASE and MAE error scores have gone up, which in turn means that the level of accuracy has gone down too. This is apparent in $/_e25$ subnet equivalent. This entails that the negative effect that the three months duration has had on AMAPS, SAMAPS, SMASE and SMAE is also reflected in MAPE, SMAPE, MASE and MAE error scores.

Table 6.6: 196-A/8-2021 - [Jan-Mar]: Accuracy Scores of Unique SRCIP/DSTIP

Subnet	**AMAPS**	**SAMAPS**	**SMASE**	**SMAE**	**RMSE**	**MSE**
Equiv	**Acc.(%)**	**Acc.(%)**	**Score**	**Score**	**Score**	**Score**
/24	**100**	**100**	**0**	**0**	**0**	**0**
$/_e25$	99.15	99.15	0.05	0.43	0.60	0.36
$/_e26$	98.24	98.24	0.11	0.88	1.14	1.32
$/_e27$	97.97	97.99	0.12	1.03	1.45	2.13
$/_e28$	97.03	97.04	0.20	1.63	2.14	4.59
$/_e29$	96.22	96.23	0.25	1.99	2.79	7.83
$/_e30$	94.19	94.15	0.36	2.95	3.65	13.36

Table 6.7: 196-A/8-2021 - [Jan - Mar]: Error Scores of Unique SRCIP/DSTIP

Subnet	**MAPE**	**SMAPE**	**MASE**	**MAE**	**RMSE**	**MSE**
Equiv	**Error(%)**	**Error(%)**	**Score**	**Score**	**Score**	**Score**
/24	**0**	**0**	**0**	**0**	**0**	**0**
/e25	50.15	66.95	3.35	6,861.61	7,329.86	5.37e+07
/e26	74.98	119.98	5.01	10,254.71	10,948.50	1.19e+08
/e27	87.43	155.36	5.84	11,959.72	12,773.25	1.63e+08
/e28	93.73	176.40	6.26	12,817.24	13,685.45	1.87e+08
/e29	96.86	187.84	6.47	13,245.78	14,139.57	1.99e+08
/e30	98.44	193.86	6.58	13,461.87	14,372.95	2.06e+08

Table 6.8: 146/8-2021 - [Jan - Mar]: Accuracy Scores of Unique SRCIP/DSTIP

Subnet	**AMAPS**	**SAMAPS**	**SMASE**	**SMAE**	**RMSE**	**MSE**
Equiv	**Acc.(%)**	**Acc.(%)**	**Score**	**Score**	**Score**	**Score**
/24	**100**	**100**	**0**	**0**	**0**	**0**
/e25	99.15	99.15	0.06	0.40	0.51	0.26
/e26	98.62	98.62	0.10	0.65	0.86	0.74
/e27	97.44	97.47	0.19	1.24	1.59	2.54
/e28	96.70	96.74	0.25	1.57	2.12	4.53
/e29	96.41	96.40	0.26	1.68	2.10	4.42
/e30	94.49	94.48	0.40	2.54	3.23	10.45

Table 6.9: 146/8-2021 - [Jan - Mar]: Error Scores of Unique SRCIP/DSTIP

Subnet	**MAPE**	**SMAPE**	**MASE**	**MAE**	**RMSE**	**MSE**
Equiv	**Error(%)**	**Error(%)**	**Score**	**Score**	**Score**	**Score**
/24	**0**	**0**	**0**	**0**	**0**	**0**
/e25	50.08	66.81	3.93	6,298.67	6,608.88	4.36e+07
/e26	75.01	120.03	5.89	9,437.30	9,908.25	9.81e+07
/e27	87.37	155.17	6.85	10,990.59	11,535.05	1.33e+08
/e28	93.69	176.26	7.35	11,785.07	12,368.97	1.52e+08
/e29	96.88	187.91	7.60	12,187.20	12,791.77	1.63e+08
/e30	98.44	193.88	7.72	12,384.57	12,999.93	1.68e+08

As in **Sections 6.3.1**, AMAPS, SAMAPS, SMASE and SMAE scores for /e25 subnet

equivalent offer the best results. The standard of interpretation in /24 where AMAPS and SAMAPS give a perfect score of 100. On the other hand, SMASE and SMAE scores need to be below 1. Using this interpretation, /$_e$25 and /$_e$26 give the best results for consideration. This is primarily because the SMAE scores for /$_e$25 and /$_e$26 are below 1. This is observed in **Tables 6.6** and **6.8**. In addition to this, the RMSE and MSE are the closest to /24 subnet in these two tables. The high values of RMSE and MSE presented in **Tables 6.7** and **6.9** make results in these tables unacceptable. They are rejected because the errors are too high. This is seen in MAPE, SMAPE, MASE and MAE scores as well. The scores in **Tables 6.7** and **6.9** present big margins from the ideal /24 scores. What is significant about results in this section is that unlike in **Sections 6.3.1**, SMAE scores for /$_e$25 and /$_e$26 are below 1. This means longer observation periods present better results than short observation periods.

When the study compared the results found in **Sections 6.3.1** and **6.3.2**, it came to a conclusion that normalising the baseline dataset and its subnet and subnet equivalents is indeed significant in order to get accurate scores when comparing different samples. This is the primary reason why AMAPS, SAMAPS, SMASE and SMAE were formulated. This study has also shown the implications of the derivations that have been made on MAPE, SMAPE, MASE and MAE mathematical models and why it was important to make them, given the significant differences in the outputs. Thus from here onward, the study used AMAPS, SAMAPS, SMASE and SMAE to further analyse and assess the differences that exist between baseline datasets and their subnet and subnet equivalents. Accuracy (%) is shortened to *Acc. (%)* in all the tables. The scores are reflecting the average number of unique SRCIP observed per DSTIP.

6.4 Model Performance: Random *vs.* Sequential

So far the study has only focused on random IBR samples and how the samples performed under the derived mathematical models. Having established the need and justified the reasons for the developed models and changes made to Hyndman and Koehler (2006) models in the preceding sections, the study changed its focus to the techniques used in sampling the data. It established that the models are a good fit for IBR data, but there was a need to assess which of the sampling techniques used in this study performed better with the mathematical models. As with the preceding sections in this chapter, this section is split into two where monthly IBR data is looked at first (represented as IBR Data I) before looking at quarterly IBR data (represented as IBR Data II). This approach was

used to evaluate the impact that time has had on the accuracy of the data samples. In each of these dataset categories, the study looked at both random and sequential data samples. This section also provides graphical representation in form of plots to support the statistical values computed from AMAPS, SAMAPS, SMASE and SMAE models.

6.4.1 Case Study: IBR Data I

Tables 6.10, 6.12 and **6.14** show accuracy score summary for AMAPS, SAMAPS, SMASE and SMAE models computed for **196-A/8** and **155/8** Network telescopes for the months of January and February. These tables also show the value of two validation errors from RMSE and MSE to support the results found in the derived models. The baseline dataset for all randomly sampled data tables (**Tables 6.10, 6.12 and 6.14**) for computing AMAPS, SAMAPS, SMASE, SMAE, RMSE and MSE is represent by /24 IPv4 subnet while the subnet equivalents are represented by $/_e25$ subnet equivalent to $/_e30$ subnet equivalent. A perfect subnet equivalent that mimics the baseline dataset is supposed to get 100% for AMAPS and SAMAPS with a score of zero SMASE, SMAE, RMSE and MSE. As explained in **Section 4.2.3**, the equitable distribution of unique SRCIPs and packets sent to the DSTIP ensured that subnets of the same size, irrespective of being on the upper or lower end of the subnet would produce similar results. The actual unique SRCIP may be different but the value counts are similar in all subnets of equal sizes. This is why this section used one subnet to present its results as opposed to the approach that is used in random samples.

Table 6.10: 196-A/8-012021: Accuracy Scores of Unique SRCIP/DSTIP [Random]

Subnet	**AMAPS**	**SAMAPS**	**SMASE**	**SMAE**	**RMSE**	**MSE**
Equiv	**Acc.(%)**	**Acc.(%)**	**Score**	**Score**	**Score**	**Score**
/24	**100**	**100**	**0**	**0**	**0**	**0**
$/_e25$	99.25	99.25	0.02	0.47	0.66	0.44
$/_e26$	98.39	98.38	0.05	0.92	1.13	1.28
$/_e27$	97.95	97.96	0.07	1.19	1.47	2.18
$/_e28$	96.53	96.59	0.14	2.32	2.89	8.40
$/_e29$	96.63	96.64	0.14	2.27	3.52	12.43
$/_e30$	94.67	94.63	0.20	3.26	4.07	16.57

Table 6.11: 196-A/8-012021: Accuracy Scores of Unique SRCIP/DSTIP [Sequential]

Subnet	**AMAPS**	**SAMAPS**	**SMASE**	**SMAE**	**RMSE**	**MSE**
	Acc.(%)	**Acc.(%)**	**Score**	**Score**	**Score**	**Score**
/24	**100**	**100**	**0**	**0**	**0**	**0**
.128/25	98.43	98.44	0.06	0.99	1.15	1.33
.192/26	98.20	98.18	0.07	1.15	1.32	1.76
.224/27	98.11	98.09	0.07	1.14	1.48	2.20
.240/28	97.69	97.68	0.08	1.35	1.73	3.00
.248/29	94.67	94.46	0.19	3.20	4.11	16.95
.252/30	91.01	90.48	0.32	5.35	6.16	37.98

Table 6.12: 196-A/8-022021: Accuracy Scores of Unique SRCIP/DSTIP [Random]

Subnet	**AMAPS**	**SAMAPS**	**SMASE**	**SMAE**	**RMSE**	**MSE**
Equiv	**Acc.(%)**	**Acc.(%)**	**Score**	**Score**	**Score**	**Score**
/24	**100**	**100**	**0**	**0**	**0**	**0**
$/_{e}25$	99.04	99.04	0.04	0.54	0.65	0.42
$/_{e}26$	98.19	98.20	0.09	1.08	1.46	2.13
$/_{e}27$	97.50	97.50	0.12	1.42	1.98	3.94
$/_{e}28$	97.44	97.45	0.13	1.51	1.91	3.66
$/_{e}29$	96.96	96.99	0.14	1.69	2.07	4.32
$/_{e}30$	92.79	92.86	0.35	4.09	4.73	22.44

Table 6.13: 196-A/8-022021: Accuracy Scores of Unique SRCIP/DSTIP [Sequential]

Subnet	**AMAPS**	**SAMAPS**	**SMASE**	**SMAE**	**RMSE**	**MSE**
	Acc.(%)	**Acc.(%)**	**Score**	**Score**	**Score**	**Score**
/24	**100**	**100**	**0**	**0**	**0**	**0**
.128/25	98.56	98.58	0.07	0.80	0.94	0.89
.192/26	97.69	97.65	0.11	1.33	1.73	3.02
.224/27	97.73	97.74	0.11	1.27	1.98	3.94
.240/28	96.16	96.27	0.20	2.29	4.46	19.93
.248/29	94.65	94.49	0.26	3.06	3.51	12.32
.252/30	91.30	90.89	0.43	4.96	5.54	30.74

Table 6.14: 155/8-022021: Accuracy Scores of Unique SRCIP/DSTIP [Random]

Subnet	**AMAPS**	**SAMAPS**	**SMASE**	**SMAE**	**RMSE**	**MSE**
Equiv	**Acc.(%)**	**Acc.(%)**	**Score**	**Score**	**Score**	**Score**
/24	**100**	**100**	**0**	**0**	**0**	**0**
/$_e$25	98.67	98.68	0.07	0.76	1.04	1.08
/$_e$26	98.17	98.15	0.10	1.01	1.24	1.54
/$_e$27	97.68	97.66	0.12	1.26	1.49	2.24
/$_e$28	97.04	97.08	0.16	1.66	2.47	6.11
/$_e$29	96.13	96.14	0.19	2.01	2.28	5.23
/$_e$30	93.63	93.45	0.35	3.64	4.65	21.67

Table 6.15: 155/8-022021: Accuracy Scores of Unique SRCIP/DSTIP [Sequential]

Subnet	**AMAPS**	**SAMAPS**	**SMASE**	**SMAE**	**RMSE**	**MSE**
	Acc.(%)	**Acc.(%)**	**Score**	**Score**	**Score**	**Score**
/24	**100**	**100**	**0**	**0**	**0**	**0**
.128/25	98.69	98.70	0.06	0.69	0.82	0.68
.192/26	98.21	98.22	0.08	0.91	1.40	1.96
.224/27	97.18	97.13	0.15	1.62	1.91	3.68
.240/28	97.07	97.04	0.15	1.57	1.85	3.43
.248/29	95.21	95.10	0.25	2.62	3.00	9.04
.252/30	92.39	92.04	0.42	4.26	4.92	24.28

As it can be seen from **Tables 6.10, 6.12** and **6.14**, which represent the randomly sampled datasets, /24 subnet has perfect scores followed by /$_e$25 subnet equivalent. This applies to all the tables computing the metrics under study. Thus from **Tables 6.10, 6.12** and **6.14**, /$_e$25 subnet equivalent is the best representation of /24 subnet which essentially means that if a /24 IPv4 net-block is randomly sampled to draw a /$_e$25 subnet equivalent and its unique SRCIP addresses analysed, an accuracy of 99.25% for both AMAPS and SAMAPS will be expected from its outcome (for **Table 6.10**). This is a very high level of accuracy. What is even more profound is that if the lowest number of unique DSTIP addresses is to be sampled (in our case /$_e$30 subnet equivalent), one would get an accuracy of 94.64% and 94.63% for AMAPS and SAMAPS respectively for **Table 6.10**. Similar levels of accuracy (slightly lower on both ends) are observed in February presented in **Table 6.12** for Network telescope **196-A/8**, which ranges from 92.79% to 99.04% for MAPS and 92.86% to 99.04% for SAMAPS. **Table 6.14**, which represents randomly

sampled data from **155/8**, shows accuracy levels ranging from 93.63% to 98.76% for AMAPS and 93.45% to 98.68% for SAMAPS. All these findings are also graphically presented later on in this section.

All these accuracy scores are a representation of how many unique SRCIP addresses a user would find in any of the given subnet equivalents for these particular network telescopes. This is the level of confidence that one would have in the feedback of representation when using subnets of the original data. Keep in mind that these are computed from the number of unique SRCIP addresses observed per DSTIP since this is standardised data based on the number of DSTIP found in each sampled subnet. On the other hand, scores shown for SMASE and SMAE, especially for $/_e25$ subnet equivalent and $/_e26$ subnet equivalent are all below 1, falling within the desired range for this type of error. Even more interesting is how the scores for RMSE and MSE are not far off from the baseline score of 1. High values of SMAE, RMSE and MSE were observed when moving away from the baseline i.e. from $/_e28$ subnet equivalent to $/_e30$ subnet equivalent. This also correlates well with the accuracy scores shown in AMAPS and SAMAPS in that, as the value of the errors are going up, the level of accuracy is going down, something that can be attributed to the number of errors found in the smaller samples.

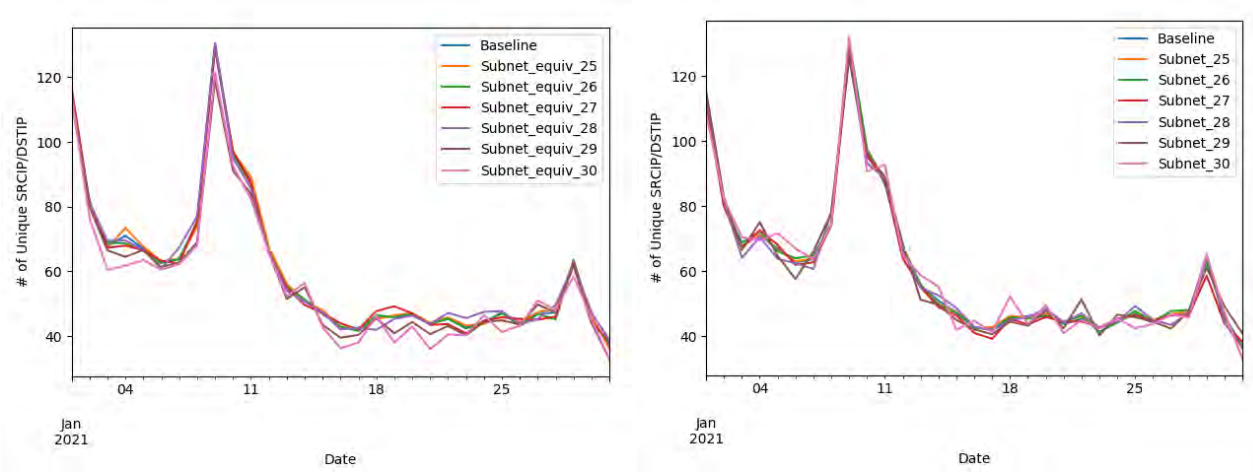
Let us now shift our attention to sequential sampling whose results have been presented in **Tables 6.11, 6.13** and **6.15.** The sequential sampling tables show computed scores of AMAPS, SAMAPS, SMASE, SMAE, RMSE and MSE. The baseline dataset is represented by .0/24 subnet while subsequent subnets are represented by .128/25 subnet to .252/30 subnet. When a review of sequential sampling datasets was made, and its accuracy reviewed, a similar pattern seen in the random sampling dataset is observed here, except this time the accuracy levels have dropped. The review and evaluation were made by ensuring that the same datasets that were randomly sampled were also sequentially sampled and given the same treatment. They are similar in the sense that sequentially sampled datasets have also proved to have high accuracy scores (see **Tables 6.11, 6.13** and **6.15**). Secondly, like in random samples, .128/25 subnet has high accuracy scores than any of the subsequent subnet done in this study. Having said that, let us look into each of the datasets closely.

Tables 6.11, 6.13 and **6.15** show accuracy scores for SRCIP addresses observed per DSTIP address in **196-A/8** and **155/8** network telescopes for the months of January and February. In all three tables, the accuracy score for the highest accuracy level is at least 98%. This is found in the .128/25 subnet. This is to say that if the **196-A/8** and **155/8** network telescope datasets have their baseline dataset (/24 IPv4 net-block)

sequentially sampled for a .128/25 subnet, the observed events and analysis will have an accuracy of at least 98%. The accuracy scores for sequential samples are slightly lower than when the same network telescope datasets were randomly sampled. Some differences are even closer to the minute detail as shown in **Tables 6.14** and **6.15**. In these two tables (**Tables 6.14** and **6.15**), /$_e$25 subnet equivalent and .128/25 subnet show a difference of 0.02 for both AMAPS and SAMAPS scores. However, looking at the remaining metrics of SMASE, SMAE, RMSE and MSE, the scores showed more variations in the value of the scores computed. This is not something that would have been picked if the analysis had just focused on AMAPS and SAMAPS scores. Thus, it is imperative to look at all metrics in each table per subnet equivalent and not isolate them.

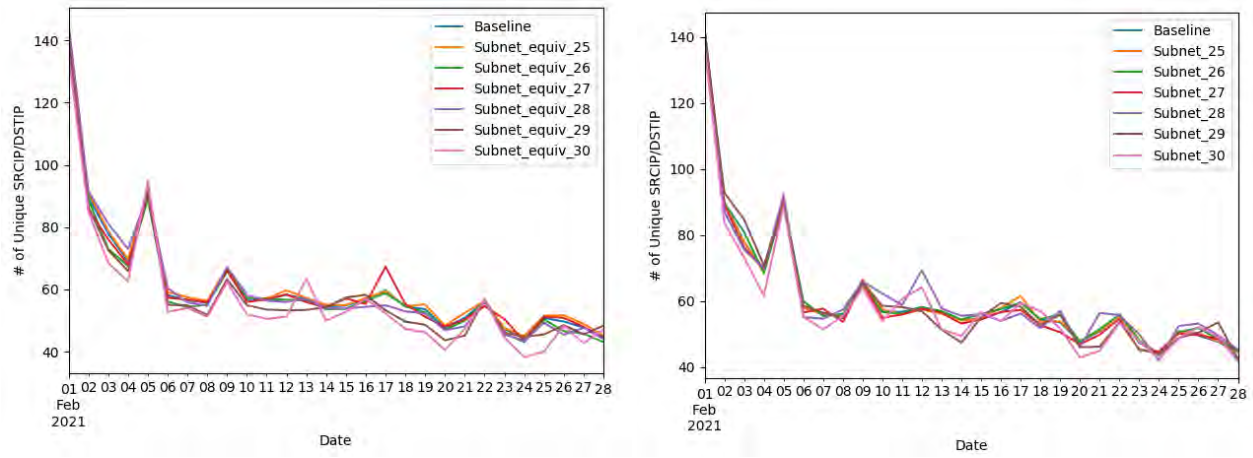
The subnets and subnet equivalents found at the bottom end of these tables (**Tables 6.14** and **6.15**), show differences between sequential and random sampling scores that are more apparent in /$_e$29 and /$_e$30 for random sampling, and .248/29 and .252/30 for sequential sampling. Just as in random sampling, as the subnet moved from .0/24 going towards .252/30, the accuracy scores gradually declined, as indicated in the columns showing AMAPS and SAMAPS accuracy scores, while the error metric scores went up (SMASE, SMAE, RMSE and MSE). This is true for all the network telescopes in different months (See **Tables 6.11, 6.13** and **6.15**). High error scores of SMASE, SMAE, RMSE and MSE as the subnet increases in number (move from .128/25 going to .252/30) are an indication of how deviated the involved subnets are from the /24 net-block. An anomaly was observed in the flow of errors when looking at .240/28 subnet in **Table 6.13** where RMSE and MSE for **6.13** is higher than that of .248/29 subnet in the same dataset. There is no reflection of this anomaly though in the other metrics. Neither does it affect the accuracy score for **196-A/8** network telescope for the month of February.

In all the data presented, for both random and sequential samples, the study observed that the lowest accuracy scores were found in **196-A/8** Network telescope particularly for .252/30 subnet (see **Tables 6.11** and **6.13**). **Figures 6.10, 6.11** and **6.12**, help to give a graphical view of how monthly random and sequential sampling relate to each other and, more importantly, how the baseline in each dataset interacts with its subnet and subnet equivalent.



(a) Subnet Equivalent (b) Subnet

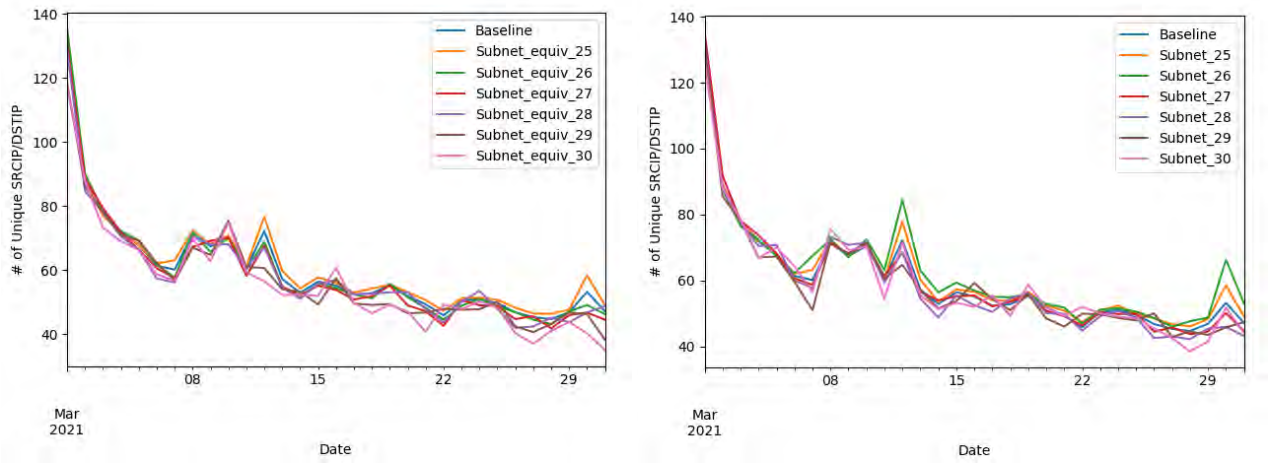
Figure 6.10: 146/8-012021 - Time Series Plot Showing No. of Unique SRCIP/DSTIP



(a) Subnet Equivalent (b) Subnet

Figure 6.11: 196-A/8-022021 - Time Series Plot Showing No. of Unique SRCIP/DSTIP

Each plot is showing the timeline with which the data was collected in the **X-axis** and the number of unique SRCIP addresses collected by individual DSTIP addresses in the **Y-axis**. For sequential sampling, the plots are labelled subnet as its legend shows an interaction between baseline for that dataset and its subnets. Throughout this study, sequential samples have named subnets hence the name subnet was preferred to label its plot. Each figure is a time series plot showing the number of unique SRCIP/DSTIP collected daily (the **X-axis** presents this clearly).



(a) Subnet Equivalent (b) Subnet

Figure 6.12: 155/8-032021 - Time Series Plot Showing No. of Unique SRCIP/DSTIP

On the other hand, all random sampling has been named *subnet equivalent* and the plot with subnet equivalent shows the interaction between the baseline and its subnet equivalents. Random sampling plots are presented in **Figures 6.10a, 6.11a** and **6.12a**, while sequential sampling plots are presented in **Figures 6.10b, 6.11b** and **6.12b**. The first thing that is apparent in all the three plots (**Figures 6.10, 6.11** and **6.12**) is how similar the shapes shown by both the random and sequential sampling data looks when plotted. For all three datasets, the pattern for random versus sequential is similar. The volume of unique SRCIP addresses per DSTIP collected is similar too (see the *Y-axis*). The peak points are similar and fall on the same dates. The way with which the subnet equivalents interact with the baseline datasets is slightly different from that of subnets collected for sequential samples, especially after mid way through the months. The differences in the levels of accuracy are reflected by the deviations observed in how subnets and subnet equivalents representing smaller subnets have deviated from the baseline.

Apart from January, (shown in **Figure 6.10**), the highest peak for all the datasets is found at the beginning of each month with the arrival of new SRCIP addresses into the network telescope. Looking at the interaction between the baseline datasets and their samples (both random and sequential) gives one more reason why quantifying the differences would show actual differences especially when the differences are minimal as is the case here. None of the samples shows significant deviation from the baseline data. That aside, it is clear to see that .252/30 subnet and $/_{e}30$ subnet equivalent (represented by light purples) is a bit further away from the baseline especially when the reader looks at **Figure 6.11**. More deviations are observed in **Figures 6.11** and **6.12** especially

when the reader looks at subnets and subnet equivalents /$_e$28, /$_e$29 and /$_e$30. It is these deviations from the baseline line that account for the high error scores and decline in the level of accuracy.

6.4.2 Case Study: IBR Data II

In this subsection, the study extends the analysis done on monthly datasets to quarterly analysis starting from January to March. When datasets from **196-A/8** and **155/8** Network telescopes were compared for monthly analysis and quarterly analysis, one significant difference between the monthly datasets and those found in **Tables 6.18** and **6.20** was found in the accuracy scores. The accuracy scores for AMAPS, SAMAPS, SMASE and SMAE have slightly gone down for **196-A/8** Network telescope in the quarterly analysis. What this means is that while the accuracy has gone down, the errors have slightly gone up over the three months period. On the other hand, network telescope **155/8** has shown a slight increase in accuracy while having the error scores gone down.

While this is the case for random sampling, in particular **155/8** network telescope data, all accuracy scores (AMAPS and SAMAPS) for both **196-A/8** and **155/8** network telescope datasets have gone down. In return, the error metrics (SMASE, SMAE, RMSE and MSE) have slightly gone down. What is more apparent with the increase of the observation period is that, overtime, it is clear to see that randomly sampled data performed better than sequentially sampled data in quarterly analysis. This is evident in **Tables 6.16, 6.18** and **6.20**. It is even clearer to see the difference between random and sequential samples when graphical representation is taken into account (see **Figures 6.13, 6.14** and **6.15**.

Table 6.16: 146/8-2021 - [Jan - Mar]: Accuracy Scores of Unique SRCIP/DSTIP

Subnet	**AMAPS**	**SAMAPS**	**SMASE**	**SMAE**	**RMSE**	**MSE**
Equiv	**Acc.(%)**	**Acc.(%)**	**Score**	**Score**	**Score**	**Score**
/24	**100**	**100**	**0**	**0**	**0**	**0**
/$_e$25	99.15	99.15	0.06	0.40	0.51	0.26
/$_e$26	98.62	98.62	0.10	0.65	0.86	0.74
/$_e$27	97.44	97.47	0.19	1.24	1.59	2.54
/$_e$28	96.70	96.74	0.25	1.57	2.12	4.53
/$_e$29	96.41	96.40	0.26	1.68	2.10	4.42
/$_e$30	94.49	94.48	0.40	2.54	3.23	10.45

Table 6.17: 146/8-2021 - [Jan - Mar]: Accuracy Scores of Unique SRCIP/DSTIP

Subnet	**AMAPS**	**SAMAPS**	**SMASE**	**SMAE**	**RMSE**	**MSE**
	Acc.(%)	**Acc.(%)**	**Score**	**Score**	**Score**	**Score**
/24	**100**	**100**	**0**	**0**	**0**	**0**
.128/25	98.79	98.80	0.08	0.56	0.68	0.47
.192/26	98.37	98.37	0.12	0.76	0.97	0.94
.224/27	97.88	97.87	0.15	0.99	1.27	1.63
.240/28	96.38	96.44	0.26	1.65	2.38	5.69
.248/29	94.73	94.56	0.40	2.51	3.10	9.61
.252/30	91.78	91.32	0.61	3.88	4.61	21.32

Table 6.18: 196-A/8-2021 - [Jan-Mar]: Accuracy Scores of Unique SRCIP/DSTIP

Subnet	**AMAPS**	**SAMAPS**	**SMASE**	**SMAE**	**RMSE**	**MSE**
Equiv	**Acc.(%)**	**Acc.(%)**	**Score**	**Score**	**Score**	**Score**
/24	**100**	**100**	**0**	**0**	**0**	**0**
$/_{e}25$	99.15	99.15	0.05	0.43	0.60	0.36
$/_{e}26$	98.24	98.24	0.11	0.88	1.14	1.32
$/_{e}27$	97.97	97.99	0.12	1.03	1.45	2.13
$/_{e}28$	97.02	97.03	0.20	1.63	2.14	4.59
$/_{e}29$	96.22	96.23	0.25	1.99	2.79	7.83
$/_{e}30$	94.19	94.15	0.36	2.95	3.65	13.36

Table 6.19: 196-A/8-2021 - [Jan-Mar]: Accuracy Scores of Unique SRCIP/DSTIP

Subnet	**AMAPS**	**SAMAPS**	**SMASE**	**SMAE**	**RMSE**	**MSE**
	Acc.(%)	**Acc.(%)**	**Score**	**Score**	**Score**	**Score**
/24	**100**	**100**	**0**	**0**	**0**	**0**
.128/25	98.39	98.41	0.10	0.83	0.98	0.96
.192/26	97.80	97.77	0.14	1.11	1.33	1.78
.224/27	97.76	97.76	0.14	1.13	1.56	2.43
.240/28	97.11	97.15	0.18	1.45	2.52	6.39
.248/29	94.81	94.66	0.33	2.69	3.40	11.60
.252/30	90.76	90.19	0.58	4.68	5.43	29.56

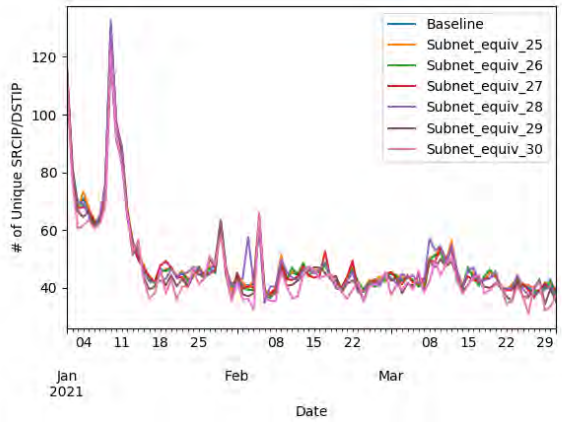
Table 6.20: 155/8-2021 - [Jan - Mar]: Accuracy Scores of Unique SRCIP/DSTIP

Subnet	**AMAPS**	**SAMAPS**	**SMASE**	**SMAE**	**RMSE**	**MSE**
Equiv	**Acc.(%)**	**Acc.(%)**	**Score**	**Score**	**Score**	**Score**
/24	**100**	**100**	**0**	**0**	**0**	**0**
/$_{e}$25	98.96	98.97	0.07	0.51	0.78	0.61
/$_{e}$26	98.10	98.07	0.14	0.94	1.26	1.59
/$_{e}$27	97.75	97.74	0.16	1.08	1.41	1.99
/$_{e}$28	97.03	97.03	0.23	1.52	2.13	4.54
/$_{e}$29	95.06	95.00	0.36	2.36	2.83	8.01
/$_{e}$30	94.00	93.89	0.46	3.01	3.89	15.16

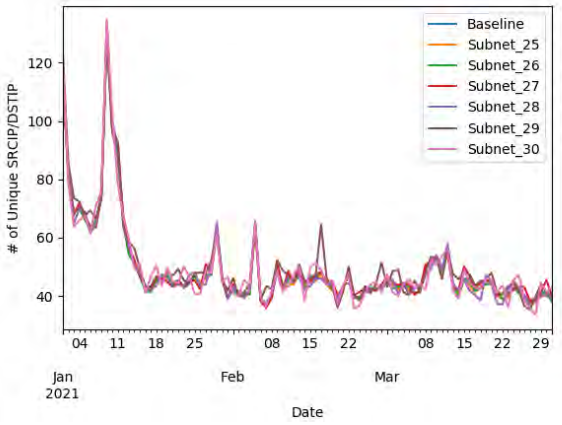
Table 6.21: 155/8-2021 - [Jan - Mar]: Accuracy Scores of Unique SRCIP/DSTIP

Subnet	**AMAPS**	**SAMAPS**	**SMASE**	**SMAE**	**RMSE**	**MSE**
	Acc.(%)	**Acc.(%)**	**Score**	**Score**	**Score**	**Score**
/24	**100**	**100**	**0**	**0**	**0**	**0**
.128/25	98.15	98.18	0.13	0.90	1.17	1.37
.192/26	98.00	97.97	0.15	0.97	1.24	1.54
.224/27	97.23	97.19	0.21	1.38	1.73	3.02
.240/28	96.82	96.77	0.24	1.55	1.92	3.69
.248/29	94.23	94.04	0.43	2.83	3.41	11.65
.252/30	92.03	91.56	0.59	3.83	4.82	23.30

As explained in **Section 6.4.1**, more deviations are observed when looking at subnet and subnet equivalents 28, 29 and 30. It is these deviations from the baseline line that account for the high error scores and decline in the level of accuracy. Another unintended observation and result that has been revealed in this quarterly analysis is the presence of the peak on the 9th of January in **Figures 6.13, 6.14** and **6.15**. This confirms how correlated the traffic and unique SRCIPs are in all the three network telescopes under study.

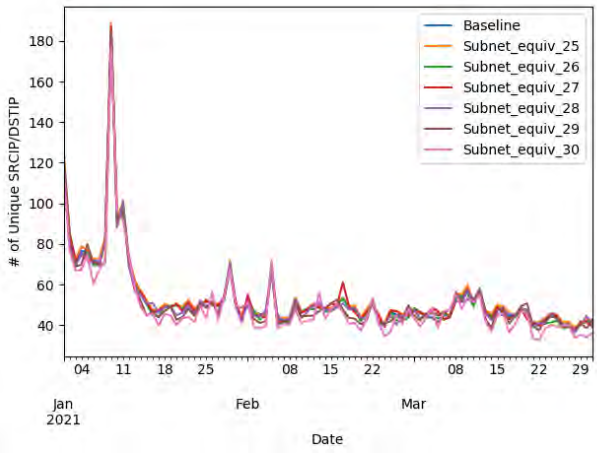


(a) Subnet Equivalent

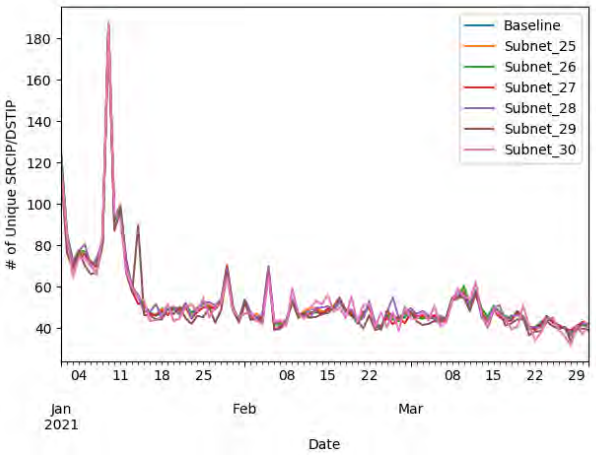


(b) Subnet

Figure 6.13: 146/8 - [Jan - Mar]: Time Series Plot of No. of Unique SRCIP/DSTIP

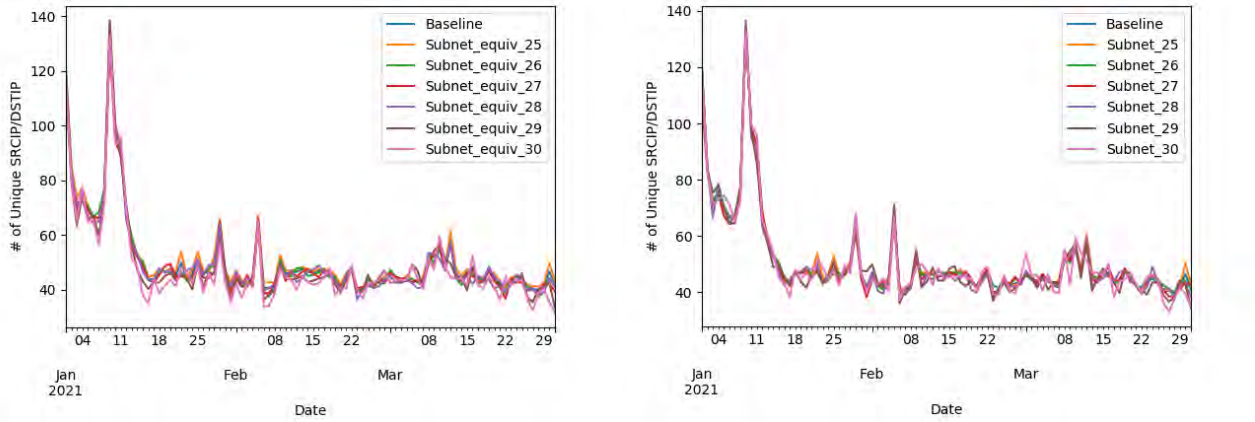


(a) Subnet Equivalent



(b) Subnet

Figure 6.14: 196-A/8 -[Jan - Mar]: Time Series Plot of No. of Unique SRCIP/DSTIP



(a) Subnet Equivalent (b) Subnet

Figure 6.15: 155/8 -[Jan - Mar]: Time Series Plot of No. of Unique SRCIP/DSTIP

6.5 Recommendations on DSTIP Monitoring and Placement

Work presented in **Sections 6.3**, **6.4.1** and **6.4.2** show that longer observations give better accuracy and error scores as compared to shorter observation periods. Shorter in this study was defined by monthly datasets while longer by quarterly datasets. Thus the study recommends longer observations to get better results. Secondly, by accommodating all scores under study (AMAPS, SAMAPS, SMASE, SMAE, RMSE and MSE), only $/_e25$ presents the best alternative to a /24 IPv4 baseline. This is the case because in all case studies SMAE score is the only one that is below the value of 1. This gets better when quarterly data is used in which case, results show that $/_e26$ and $/_e27$ has also scores that are below 1. Acceptable scores for SMAE ought to be below 1. All samples that have had a score below or around 1 have shown the lowest error scores from RMSE and MSE. Thus following $/_e25$ are $/_e26$ and $/_e27$. With this premise and results presented, this study would not recommend setting up a network telescope below 32 unique DSTIP. SMAE sets set a strict criteria of acceptable score, as such although the accuracy scores for $/_e28$ - $/_e30$ are above 90% its SMAE score is out of the acceptable range.

Furthermore, when the study compared random and sequential samples, random samples performed better. Thus this study recommends random placement of unique DSTIP in network telescopes as opposed to sequential. *This is true for both long term observation as well as short term observation of DSTIPs.* Lastly, high accuracy scores for AMAPS,

SAMAPS, SMASE and SMAE show that the models developed perform better especially when RMSE and MSE are taken into consideration. The study therefore recommends the use of RMSE and MSE when computing the difference (in unique SRCIP addresses per DSTIP) that exists between samples.

6.6 Feasibility of Sampling IBR Data

The study has successfully proved that it is feasible to sample and use a portion of a network telescope and still attain high levels of representation from the baseline dataset. From the findings presented in **Section 6.4**, it is apparent that randomly sampled datasets have higher accuracy scores than sequentially sampled data. Thus, the study recommends randomly sampling unique DSTIP addresses to an end user. However, it does not disregard the significance of sequential sampling as the differences in the level of accuracy were not very significant. Secondly, SRCIP addresses make a strong case of showing higher accuracy levels than packet dataset, with its lowest accuracy presented at at least 93% accuracy score for AMAPS and SAMAPS scores and going as high as 99%. These values are true for randomly sampled datasets. Work done on packets per unique DSTIP address (TCP network traffic) can be found in Chindipha *et al.* (2019b). It was from the findings of this paper that the study opted to work with unique SRCIP addresses as compared to packets when it comes to computing the accuracy of sample representativeness. This was supported by the fact that some unique SRCIP addresses sent more packets than others thereby affecting the normalisation of the data, which in turn negatively affected the quality if the results.

On the other hand, sequentially sampled datasets have shown the lowest accuracy score of at least 90% for AMAPS and SAMAPS scores and go as high as 98%. A .128/25 subnet and $/_{e}25$ subnet equivalent, are the best options thus far showing remarkable scores throughout the study when compared to the other subnets and subnet equivalents. However, this is only true if a network telescope user is interested in near perfect representation of the baseline. Otherwise, what this simply means is that more unique DSTIP addresses are still better than fewer DSTIP addresses. However, these models offer the levels of accuracy with which if a user is comfortable with can use to work with a smaller telescope. For instance, an accuracy of 97% for 16 unique DSTIP addresses is certainly good enough to offer insights and confidence to a network telescope user who only has 16 randomly spaced DSTIP addresses. This knowledge helps the user to know how much is most likely to be missing and make plans accordingly.

The study also shows that the use of subnets and subnet equivalents as a means of identifying activities happening from their original baseline dataset is viable. This is primarily attributed to the high accuracy scores observed in the bigger samples (like .128/25 subnet and /$_e$25 subnet equivalent). Acceptance of the level of accuracy varies with the intent of the use of the data. However, what is certain is that there is at least a 93% chance of getting the estimations of the original data correctly. The technique of random sampling of subnet equivalents can be employed to create a subnet equivalent with a high level of fidelity when compared to the original subnet. In the end, one ends up saving on the time needed to work on large chunks of dataset, or if IP address blocks are limited, without a lot of compromise on the accuracy of the results.

6.7 Analysis of Model Performance on IBR Data

Although processing and working with a /24 IPv4 net-block datasets gives better results by affording an end user a wider scope of events than what one would get with a smaller network telescope, this study has shown that it is possible to use accuracy metrics to sample out a baseline dataset and use it as a new dataset. Using the level of accuracy shown in samples of .128/25 to .252/30 subnets for sequential data samples and /$_e$25 to /$_e$30 subnet equivalent for random samples, it is evident that the error of margin observed while sampling out the dataset is relatively small particularly for bigger subnet equivalents.

Using AMAPS, SAMAPS, SMASE and SMAE models, randomly sampled datasets have proved to have very high accuracy levels when compared to sequentially sampled datasets. When the metric scores were used to test the accuracy levels of number of unique SRCIP addresses per DSTIP, the randomly sampled hosts showed relatively higher accuracy than the sequential samples. This was true for both monthly and quarterly datasets. The models have shown high accuracy levels of over 99% for randomly sampled datasets and 98% for sequentially sampled data. A mean accuracy score of at least 95% for both AMAPS and SAMAPS in both sequential and random datasets has shown that it is possible to sample out 16 unique DSTIP addresses from a baseline dataset and still achieve high levels of accuracy in one's threat intelligence gathering using a small aperture network telescope.

Between the two comparable metrics, SAMAPS produced slightly high accuracy scores than AMAPS. This is attributed to its symmetrical nature when computing the scores.

SMASE, on the other hand, operate on a different scale with all of the samples falling within the acceptable range of between 0 and 1. The derived models of AMAPS, SAMAPS, SMASE and SMAE have proved to be reliable tools to use when comparing different samples as long as the data samples are normalised with well know standardising mechanisms. The models have also been shown to operate along other mathematical models like RMSE and MSE to achieve the same goal. In addition to this, the models have proved to work with both sequential and random samples and still achieve very good results. More importantly, in addressing the main research question, the models have shown that it is feasible to sequentially and randomly sample IBR datasets to represent a baseline dataset. In the process of doing this, they have opened the way for the use of smaller scattered addresses within larger organisational networks to be utilised for threat intelligence and IBR collection, and offer an opportunity for those with small address blocks to use whatever they can afford to utilise for passive threat intelligence gathering.

6.8 Strengths and Limitations of the Developed Models

Considering the fact that the AMAPS, SAMAPS, MMAE, MMASE were derived from the MAPE, SMAPE, MAE and MASE, they do, in part, inherit some of the limitations that these models have. However, there are also some strengths that have been rectified with these models that are present in MAPE, SMAPE, MAE and MASE. It is important to note that these strengths and limitations presented in this section have mostly to do with the usability and functionality of the models. Thus, some of these limitations found in MAPE, SMAPE, MAE and MASE are present because these models are used for forecasting. However, this study was not interested in forecasting. Its main interest was on quantifying the variations present in the datasets already collected to understand how different each of the large samples were from smaller ones.

6.8.1 Strengths

In MAPE, SMAPE, MAE and MASE, if the value is not present on the actual values, then it becomes undefined in the forecast values. Essentially, the forecast value is going to be given a value of zero, which does affect the trend when it comes to computing the average but also when graphically presenting the data. AMAPS, SAMAPS, SMAE and SMASE do not have this drawback in that, if the data point is not present in the baseline, it follows without loss of generality that it will not be found in any of subnet or subnet

equivalents. Thus, when computing the mean between the baseline and the subnet (and subnet equivalent), such data points are not accounted for and do not affect the levels of accuracy presented. AMAPS, SAMAPS, SMAE and SMASE operate on the principle that if the data point is not present in the baseline dataset, there is no need to add it to the time series for computation. This is not the case with MAPE, SMAPE, MAE and MASE as one needs to account for every value present and not present in the computation.

While the models developed by Hyndman and Koehler (2006) only work with time series data, the models developed in this study have been tested with data that is either of a time series nature or just a series of data samples with no time attached to it. If the user is using time series data then one needs to make sure that the timeline is well adjusted to all data samples to avoid any bias. For instance, in this study, the fact that data was sampled out from the baseline data meant that, by default, some data points were left out of the sample draws. With this, the study had to keep in mind that if the analysis of the data was done on an hourly basis, then it is possible that the sample draws may not have some unique SRCIP present in their pool. This is the case because some SRCIP addresses were only present for certain hours within a particular day.

What this means is that, in this study, there were some cases where certain unique SRCIP addresses were registered by the network telescopes within a certain section of the day and never showed up again uniformly within the whole day. In those instances, they were not registered to all unique DSTIP addresses, thus not being uniformly available at the lowest levels of the unit of measure, which is time. Thus the lowest unit of measure would be seconds, minutes, and then hours. So, some unique sources would be available in certain unique DSTIP addresses within specific minutes of the day but not present within every hour.

Now, if the study's time-value(t) was put at hourly analysis instead of daily analysis, what that means is that the size of the data samples was going to be different since the unique SRCIP addresses were not uniformly distributed within the given day. Thus, though the computation will be possible, it would affect the level of accuracy of the results, hence the unit of time has to be taken into consideration. Obviously, the baseline will contain all the data points of interest, but hourly analysis on the samples will contain more variation against the baseline than when the *time-value* is scaled up to daily analysis. However, As the duration of the observation gets longer, these variations get obscured as some of the data points missed out earlier are recorded or caught at a later time within the sample subnet equivalent.

6.8.2 Limitations

There are some limitations that are unique to these developed models. To begin with, if a user of the model fails to find the appropriate unit or value to standardise the values contained in different samples, then the level of accuracy computed from the different samples is going to be wrong. This then makes it difficult to compare two different samples since the unit of measure has now changed, leading to results that could not only be misleading but also wrong. Thus a model user needs to make sure that different samples have the same unit of standardising the data points as the baseline data.

If it happens that a model user is working with data that does not have a unit with which to standardise the other data samples, then these models cannot be used to measure the degree to which they are varying. For IBR data, our unit of measure was to try and identify the number of unique sources received per individual DSTIP address. Thus, the size of the subnet or the subnet equivalent becomes this study's value of standardising its samples. If a mistake is made by using the wrong size for a subnet to standardise, then the results will be wrong and misleading. Standardisation is very critical to these models as they help to ensure that the different samples are comparable and are brought to the same scale.

6.9 Port Analysis Using JD, TF and IDF

This section shows how the techniques presented in **Section 3.10** are used in line with this study. As alluded to in **Section 3.10**, the formulae will be presented in the context of this study. This is to say, there will not be the mention of an item or word or document in the formula. Instead, ports and subnet equivalents will be used to show how the methods have been adapted to fit this study. This section will focus more on the computed scores for all the four Information Retrieval and Text Mining Techniques (IR-TMT) using the theoretical knowledge that has been presented already in **Section 3.10**. Initial work which formed the basis of this section was published in (Chindipha *et al.*, 2019a).

Table 6.22: 155/8-012021: DPORT IR-TMT Scores for TCP Traffic

Subnet	JD	TF	IDF	TF-IDF
/24	0	0	0	0
.128/25	0.0	1.52 x 10^{-05}	0.30	4.59 x 10^{-06}
.192/26	0.003	1.53 x 10^{-05}	0.30	4.59 x 10^{-06}
.224/27	0.004	1.53 x 10^{-05}	0.30	4.59 x 10^{-06}
.240/28	0.005	1.54 x 10^{-05}	0.30	4.59 x 10^{-06}
.248/29	0.006	1.54 x 10^{-05}	0.30	4.59 x 10^{-06}
.252/30	0.012	1.55 x 10^{-05}	0.30	4.59 x 10^{-06}

Table 6.23: 146/8-012021: DPORT IR-TMT Scores for TCP Traffic

Subnet	JD	TF	IDF	TF-IDF
/24	0	0	0	0
.128/25	0.0	1.52 x 10^{-05}	0.30	4.59 x 10^{-06}
.192/26	0.002	1.52 x 10^{-05}	0.30	4.59 x 10^{-06}
.224/27	0.003	1.53 x 10^{-05}	0.30	4.59 x 10^{-06}
.240/28	0.005	1.53 x 10^{-05}	0.30	4.59 x 10^{-06}
.248/29	0.007	1.54 x 10^{-05}	0.30	4.59 x 10^{-06}
.252/30	0.014	1.56 x 10^{-05}	0.30	4.59 x 10^{-06}

Tables 6.22 and **6.23** show summary of scores for *JD*, *TF*, *IDF* and *TF-IDF* with data collected from **155/8** and **146/8** Network telescopes respectively. All the data used in this section is composed of TCP Traffic. In addition to this, note that all the results from this section used sequential samples. The core idea behind using sequential sampling on this was to accommodate every possible occurrence of the DPORTs as the objective was to quantify any differences that may exist between the /24 IPv4 baseline and the subnets. With random sampling, it is possible to skip some ports in the samples. Sequential sampling offered the best approach to computing port differences in samples It is for this reason that this study will not show IR-TMT Scores for random sample datasets. Note also how similar the results have been thus far in different datasets from different network telescopes. Thus to demonstrate the differences in DPORTs that exist between datasets, the study used January datasets from **155/8** and **146/8** Network telescopes.

Table 6.24: 155/8-012021: Differences in DPORT Count for TCP Traffic

Subnet	Count	Diff (Count)	Diff (%)
/24	65,485	0	0
.128/25	65,485	0	0
.192/26	65,373	112	0.17
.224/27	65,305	180	0.27
.240/28	65,262	223	0.34
.248/29	65,105	380	0.58
.252/30	64,641	844	1.28

Table 6.25: 146/8-012021: Differences in DPORT Count for TCP Traffic

Subnet	Count	Diff (Count)	Diff (%)
/24	65,485	0	0
.128/25	65,485	0	0
.192/26	65,469	16	0.02
.224/27	65,345	140	0.21
.240/28	65,232	253	0.38
.248/29	65,112	423	0.64
.252/30	64,521	964	1.47

A Jaccard Distance ranges on a scale of 0 to 1 where 0 is a match of the baseline dataset (/24 IPv4) and a subnet. On the other hand, a value of 1 means that the baseline dataset (/24 IPv4) and a subnet under review are completely different. Both the **155/8-012021** and **146/8-012021** databases demonstrate that the DPORTs show that a .128/25 subnet is an accurate representation of IPv4 net-block address with a distance of zero. **Tables 6.24** and **6.25** support this argument by showing the difference in the number of unique DPORTs that exists between the baseline dataset and .128/25 subnet.

For both **155/8-012021** and **146/8-012021** databases, the DPORTs difference in the .128/25 subnets (see **Tables 6.24** and **6.25**) accurately represent this difference by showing the same number of unique DPORTs. However, moving from .128/25 subnet heading towards .252/30 subnet, the differences in the number of unique DPORTs become more apparent. While still on JD, as the differences in the number of unique DPORTs increase (shown in **Tables 6.24** and **6.25**), JD is increasing as well, reflecting this variation. The highest JD (0.014) is found **Table 6.25** for **146/8-012021** dataset which shows a

DPORT composition difference of 1.47% (see **Table 6.25**) from the baseline database. All the differences in the subnet are computed against the /24 IPv4 baseline i.e. the study subtracted the value found in individual samples from the baseline dataset. Using these differences, the study computed the percentage composition difference. This is how values in **Tables 6.24** and **6.25** were computed.

The term frequency (TF) scores show a slight variation in **Tables 6.22** and **6.23**. In fact, it is very easy to miss out on such differences because they would almost equate to zero if they are reduced to four decimal places. What is significant in these scores is to note that the scores are different because the sizes of the subnets are different, and thus when compared against the baseline dataset, it is reflecting these variations. This should have raised concerns had the TF scores been the same since this TF technique takes frequency into account and from **Tables 6.24** and **6.25**, it is apparent that the content of one subnet is slightly different from the next. A good judgment of this is shown in **Table 6.25** where the difference between .128/25 subnet and .192/26 subnet is only 16 and thus their frequency is identical (see **Table 6.25**).

The variations in DPORTs are not large, hence the TF scores being either identical or close to each other. What this means is that according to TF scores, these subnets are a good fit as a representation of the baseline dataset because the scores are very close to zero, which is a good indicator of ideal scores. However, in JD computation and interpretation, if a perfect representation is to be picked, then the bigger subnet will be chosen because the variations are smaller in these than they are in smaller subnets (see Tables **Tables 6.24** and **6.25**). What is more significant to note, however, is how small the differences are between the baseline datasets and .252/30 subnet. The difference in the composition of unique DPORTs found between the baseline datasets and .252/30 subnet is less than 1.5%. This is a very good variation for a worst-case scenario.

Finally, the study looked at IDF and TF-IDF. It is very easy for one to assume that there must have been something wrong with these scores since they are the same, however, they accurately represent the summary statistics of the variations that exist between the baseline and subnet equivalent datasets. Keep in mind that IDF in this study measures DPORTs that are not common in the two samples being evaluated i.e. the baseline dataset and any subnet under evaluation. Since all DPORTs are unique, then it is only logical that their weight be the same since each port is represented once in the subnet equivalent. Once because each port is unique. If we match /24 IPv4 against itself, it will give a value of zero because all the DPORTs are present in both datasets.

TF-IDF is computed using part of IDF, and as such, as long as one of the two variables

remains the same, then TF-IDF will reflect that. More importantly, the TF-IDF scores are the same in both **155/8-012021** and **146/8-012021** dataset because they are uniquely represented in their subnet. Since the score for IDF and TF-IDF are closer to zero than one, it is safe to say that, as far as DPORTs are concerned, all the subnets equally represent the baseline dataset. In **Section 4.2.2**, this study showed that most of the dominant DPORTs received roughly about the same amount of traffic from SRCIP addresses. In the same section, the study showed that such DPORTs were present to all unique DSTIP addresses. This occurrence helps to explain why there are very few variations in the number of unique DPORTs contained in the different subnet. These results basically show that most DPORTs are present in all the unique DSTIP addresses. All the top 10 DPORTs had the same traffic reception with the key difference being the volume of traffic that the number of DPORTs contained in each subnet received.

The study opted to use the number of unique DPORTs found in a given subnet instead of just accommodating the differences in the frequency of the DPORTs because different ports have different vulnerabilities. So, if a port is recorded once, there is no need for another occurrence to verify a specific vulnerability under study. A single occurrence is deemed to be sufficient evidence of its existence. In another study, it would be significant if the research is testing for specific vulnerability (such as Distributed Denial of Service (DDoS) attack), but in this study, the frequency of ports in a subnet was not significant.

6.9.1 Recommendations for DPORTs

With observations made in this study, this research recommends that the Information Retrieval and Text Mining Techniques (IR-TMT) be used as a method to compare the dissimilarity between the baseline dataset, and the datasets. The study has shown that when unique DPORTs are taken into consideration, there are negligible differences that exist between the baseline dataset and the subnets. The weighting scores show negligible difference when comparing data sets from different network telescopes. The study has also shown that there are negligible differences between the baseline dataset and the smallest subnet (1.47% difference with the poorest subnet scores). It is thus not surprising that the scores are reflecting this proximity, i.e. the scores are closer to zero than one. This is proof of how close the subnet equivalents are to the baseline database. The scores are almost identical because more than 75% of the unique DPORTs are present in all the subnet equivalents.

Another finding that stood out were the scores of IDF and TF-IDF which were constant

in all the subnet equivalents. The TF-IDF scores are the same in both databases because each of DPORT in all the subnet is unique, and thus, given the same weight throughout the subnets. This is significant because, if the scores were different, it would have raised many questions in the suitability of these techniques to this case study. AS in all other fields, unique items have the same weight as their dataset. Since most of the scores are identical, one measure that easily distinguishes the scores is the Jaccard Distance. As the subnet size decreased, the distance increased gradually. Thus it is confirmed that larger subnets are still a better representation of baseline datasets than smaller ones. However, in this case, such marginal differences can be overlooked as the differences are not huge, making each subnet a viable option for selection.

6.10 Summary

In this chapter, this study proposed a number of models that can be used to quantify the differences that exist between the baseline datasets and subnets (for sequentially sampled data) and subnet equivalents (for randomly sampled data). The overall objective behind this chapter was to quantify that it is possible to use a small sized network telescope and know the actual differences that exist between the large network samples and smaller network samples.

The chapter begins by introducing the mathematical models that have been derived to work with IBR data in **Section 6.1**. In this section, Absolute Mean Accuracy Percentage Score (AMAPS), Symmetric Absolute Mean Accuracy Percentage Score (SAMAPS), Standardised Mean Absolute Error (SMAE), and Standardised Mean Absolute Scaled Error (SMASE) are explained. The research approach used in analysing the IBR ready for use is explained in **Section 6.2**. Visual representations are utilised herein in order to show how the data looked like prior to being processed. This is done in the form of time series plots as well as box plots. This is for both sequential and random datasets. An evaluation of AMAPS, SAMAPS, SMAE and SMASE against MAPE, SMAPE, MAE, MASE is explained in **Section 6.3**. This is immediately followed by an assessment of the performance of the Models on Random vs Sequential IBR Samples in **Section 6.4**. In each of these assessment sections, both monthly and quarterly datasets are evaluated. In **Section 6.5**, the study made recommendations on monitoring and placement of DSTIP in order to get the best results from them.

Feasibility study of sampling IBR data is conducted in **Section 6.6**. The performance of the model on IBR data is explained in **Section 6.7**. The study also assessed the strengths

and limitations of the developed models in **Section 6.8**. At this point, the study had conducted tests on SRCIP and DSTIP addresses. Thus the study changed its focus and conducted a port analysis in **Section 6.9**. This was done using Information Retrieval and Text Mining Techniques (IR-TMT). This section also offered recommendations on its finding regarding DPORTs.

Having analysed IBR datasets using Bootstrapping and then evaluating them using some of the mathematical models (both developed and old techniques), the study opted to assess the practical aspects of its findings. More importantly, assessing how the size of the network telescope affects the proportion of how many unique SRCIP addresses are collected when it has been used over a period of time. All this is explored in **Chapter 7**

7

Practical Applications and Implications

Having undertaken analysis using Bootstrapping in **Chapter 5** and analysed the data using custom made mathematical models in **Chapter 6**, the study transitioned to evaluate some of the practical applications in each of the analyses conducted.

In **Section 7.1**, the study expands and clarifies the work done in **Chapter 5** by showing why bootstrapping answers the research questions and how it can be applied. This is the section that addresses the application of Bootstrapping IBR data. This is followed by **Section 7.2**, which shows the effect of time on the amount of threat intelligence data. To be more specific, the study analysed how long it takes for different sizes of network telescope to collect specific proportions. This section helps the reader to understand the effect of using a small-sized network telescope on the volume of unique SRCIP addresses collected over time. This leads to **Section 7.3** which focuses on cross disciplinary practical applications of the models developed outside of IBR data. In **Sections 6.3** and **6.4**, this research has laid a good argument on how the mathematical models developed can be used to quantify differences in terms of threat intelligence data. This chapter expands the application of these models outside of IBR data.

7.1 Applications of Bootstrapping IBR Data

Table 7.1: Monthly Summary Table for CI in Percentage at 95% CI

Bootstrap Size	Avg. Bootstrap Mean %	Average SEM	% CI
256 - /e24	100.00	4.68	[95.32 - 104.68]
128 - /e25	50.17	2.34	[47.83 - 52.51]
64 - /e26	25.17	1.20	[23.97 - 26.37]
32 - /e27	12.50	0.64	[11.86 - 13.14]
16 - /e28	6.21	0.37	[5.75 - 6.49]

Table 7.2: Quarterly Summary Table for CI in Percentage at 95% CI

Bootstrap Size	Avg. Bootstrap Mean %	Average SEM	% CI
256 - /e24	100.00	1.76	[98.24 - 101.76]
128 - /e25	50.30	0.92	[49.38 - 51.22]
64 - /e26	25.45	0.51	[24.94 - 25.96]
32 - /e27	12.69	0.29	[12.40 - 12.98]
16 - /e28	6.78	0.27	[6.51 - 7.05]

As mentioned in **Section 5.7**, IBR data has shown results that were more reflective of the baseline bootstrap when working with non-parametric bootstrapping than parametric bootstrap, and thus the study recommended the use of non-parametric bootstrapping for future studies. The study also recommended the use of the data at least at 95% CI level because the variations between 95% and 99% CI level are not big. **Tables 7.1** and **7.2** are repeated and made available for discussion in this section. These two tables are first presented in **Section 5.7** and presented here to show how these artefacts can be applied in in real world.

Each network telescope size has its associated proportion of unique SRCIP addresses, overall, when compared to the baseline. The two tables reproduced in this section show these proportions. An organisation that does not have large network telescopes can use these proportions in the table to compute how much their current size can collect in relation to the baseline. Currently, if a network telescope user bootstraps a /25 dataset, the average number of unique SRCIP/hour collected by the user's network telescope will range between 47.83% - 52.51%. This range presents confidence intervals at 95% CI that would allow the user to make decisions with that level of confidence. The confidence in the data allow a network telescope user to make decisions that they would be comfortable with.

More importantly, the tables show that smaller network telescopes can also be entrusted to collect IBR data. Having bootstrapped IBR data, value has now been added to show how much data can be collected by smaller network telescopes on hourly basis. Recall that bootstrapping enabled the computation of CI, and it is the CI that can now give confidence to any user who will use the IBR data. One of the research questions was to quantify how the baseline can compare to a smaller network telescope. Now that these values are known, decisions can be made from it.

As stated in **Section 5.7**, the actual count of average number of unique SRCIP/hour may be different but the proportional range will remain the same. As such, other researchers no longer need to have a larger telescope in order to observe the estimates of how proportional their smaller sized network telescopes can relate to larger network telescopes. Other researchers do not need to have a baseline to quantify the average number of unique SRCIP/hour observed in their freely random selected DSTIPs. The table proportions can be used to compute the range of values expected as long as they know their SEM. IBR data users can present their findings with 95% degree of certainty using **Tables 7.1** and **7.2**. In other words, **Tables 7.1** and **7.2** offer other IBR researchers the range of values that they should expect their estimates to fall between, 95% of the time if they run their experiment again or re-sample the IBR data in the same way.

Longer observation periods are highly recommended as narrow or small confidence interval indicates that if a network telescope user were to bootstrap a different IBR sample, then the researchers are reasonably sure they would get a similar result. A wide confidence interval indicates that we are less sure and perhaps information needs to be collected from a larger sample to increase the user's confidence. This is apparent in **Table 7.2** which shows narrow CI as compared to **Table 7.1**. The work presented here can act as a benchmark to future work. For instance, now that the proportion of representation for different network telescopes are known and the average number of unique SRCIP/hour can be computed from them, this knowledge that was not available prior to this research, directly answers the question of quantifying the differences that exist between small and large network telescope sensors. The next set of questions would be to quantify the threat intelligence data that these different sizes collect. Are the proportions observed here going to match with the value of the threat intelligence each network telescope size collects? Do the unique SRCIP collected by different network telescope sizes contain unique threat intelligence data as well? Or are there some commonalities in the threat collected by the identified pool of unique SRCIPs? These are some of the questions that future researchers can address having learnt of the CI computed from bootstrapping IBR data.

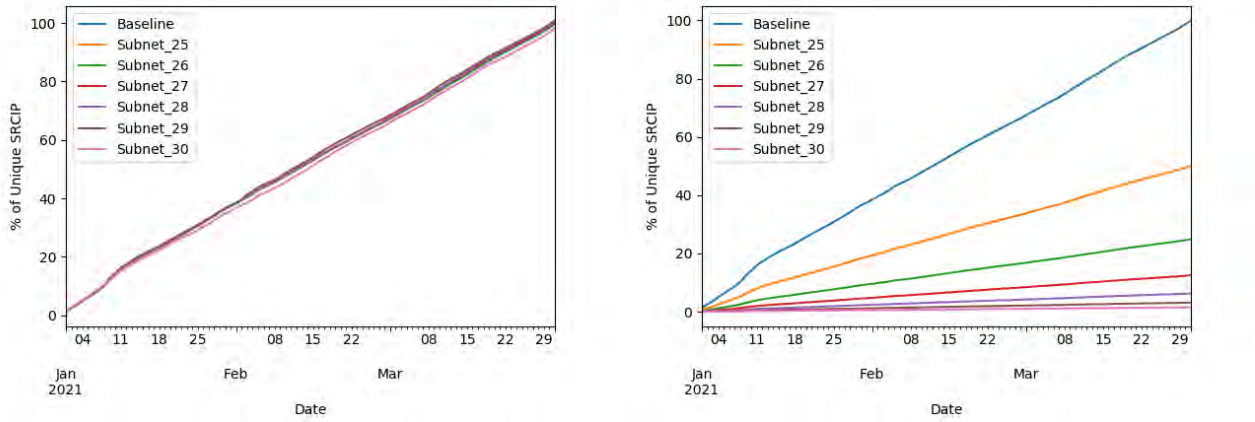
7.2 Unique SRCIP *vs.* Time

So far the study has quantified and graphically demonstrated that large samples, which represent large network telescopes, are a better option to replace a /24 IPv4 baseline dataset than smaller networks. This study has offered smaller network telescope users the details that they need to make informed decisions. The information comes with certain levels of confidence in the data that is collected by the small-sized network telescopes. This study provides the differences and the levels of confidence offered by the different samples which represent different network telescope lenses. This study has provided this information graphically and quantitatively.

The study has also shown that, given enough time, small network telescopes should be able to collect the same amount of threat intelligence, although it may require more time for them to attain the desired amount of data. However, one research question that has not been addressed yet is how long it would take different sizes of network telescopes to observe a certain proportion of the total amount of unique SRCIPs. For instance; how long would it take for a small-sized network telescope to observe 20% of the unique SRCIPs that contributed to the total traffic observed in any network telescope? Does the time taken to observe such a proportion differ when the size of the network telescope changes? These are the questions that are addressed in this section as they offer a realistic timeline for a network telescope user to work with. The study also conducted a time series analysis by analysing what proportion of unique source IP addresses are observed at different time frames. The two questions being addressed in this section are:

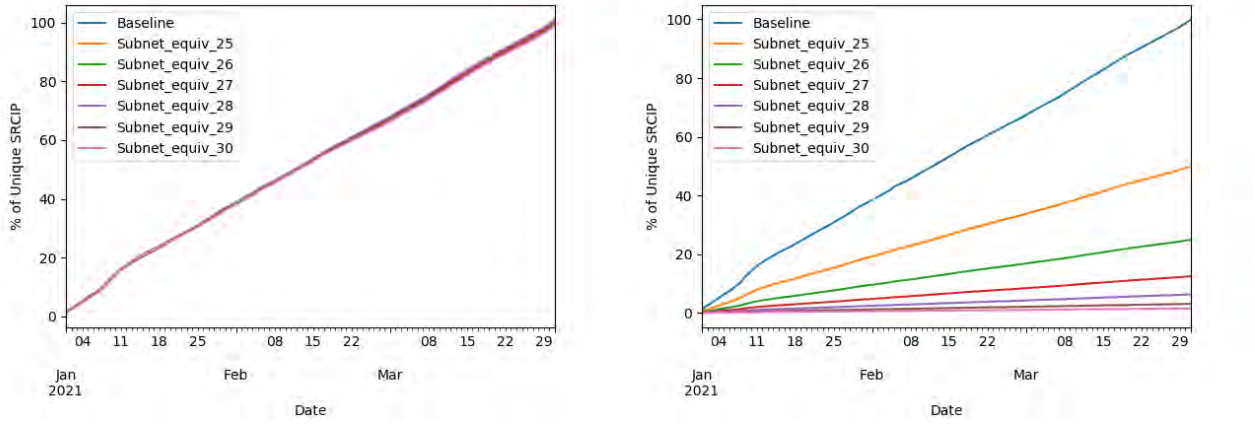
1. How long does it take to observe a specific proportion (e.g. 10%, 20% or 50%) of the unique sources?
2. What happens to these proportions when the size of the network telescope changes? Two approaches were used on this: firstly by just observing one telescope, and secondly, by observing multiple telescopes.

These questions offer practical applications to a network telescope user as they would know when to expect certain amounts of data to be collected. Using the data received from the unique SRCIPs, threat intelligence can be extracted. To address these questions, this study used monthly and quarterly data collected in 2021.



(a) Normalised (b) Not Normalised

Figure 7.1: 146/8 -[Jan - Mar]: Unique SRCIPs over time [Sequential]



(a) Normalised (b) Not Normalised

Figure 7.2: 146/8 -[Jan - Mar]: Unique SRCIPs over time [Random]

Figures 7.1 and **7.2** show cumulative time series plots for **146/8** Network telescope. **Figure 7.1a** and **7.2a** are normalised by computing the percentage composition against the total of the individual datasets. This is why each of the plots amount to 100 %. This is how all plots have been normalised in this section. These plots were included in this section to show how normalised datasets relate to each other. **Figures 7.1b** and **7.2b** are not normalised. Each data sample is computed against the baseline total as its denominator. The explanations in this section will primarily focus on the datasets that were not normalised as they show results that are relative to a larger network telescope. Thus offering a comparison with the base.

The most outstanding result from **Figures 7.1b** and **7.2b** is how similar random and sequential data samples look like. Both **Figures 7.1b** and **7.2b** show the proportion of the number of unique SRCIP observed at specific times for **146/8** network telescope with data collected from January to March. The data used for **Figures 7.1b** and **7.2b** is summarised in **Tables 7.3** and **7.4**. In this section, T_0 in the tables represent week zero while T_{LD} represent the last day of observation.

Table 7.3: 146/8-[Jan - Mar]: Sequential Cumulative % Summary Table

Timeline	/24	/25	/26	/27	/28	/29	/30
T_0	1.34	0.67	0.34	0.17	0.08	0.04	0.02
T_1	8.78	4.47	2.15	1.08	0.54	0.27	0.14
T_2	19.54	9.84	4.86	2.43	1.20	0.62	0.29
T_3	26.68	13.40	6.67	3.35	1.64	0.84	0.40
T_4	34.19	17.18	8.52	4.27	2.14	1.07	0.51
T_5	41.56	20.86	10.35	5.20	2.60	1.32	0.62
T_6	48.97	24.60	12.18	6.12	3.07	1.56	0.73
T_7	56.58	28.38	14.11	7.06	3.54	1.80	0.85
T_8	63.51	31.83	15.85	7.95	3.97	2.02	0.97
T_9	70.60	35.38	17.60	8.86	4.41	2.24	1.08
T_{10}	78.29	39.21	19.50	9.83	4.92	2.49	1.20
T_{11}	86.31	43.27	21.46	10.82	5.44	2.73	1.33
T_{11}	93.44	46.84	23.28	11.70	5.88	2.95	1.43
T_{LD}	100.00	50.16	24.91	12.52	6.28	3.15	1.53

Table 7.4: 146/8-[Jan - Mar]: Random Cumulative % Summary Table

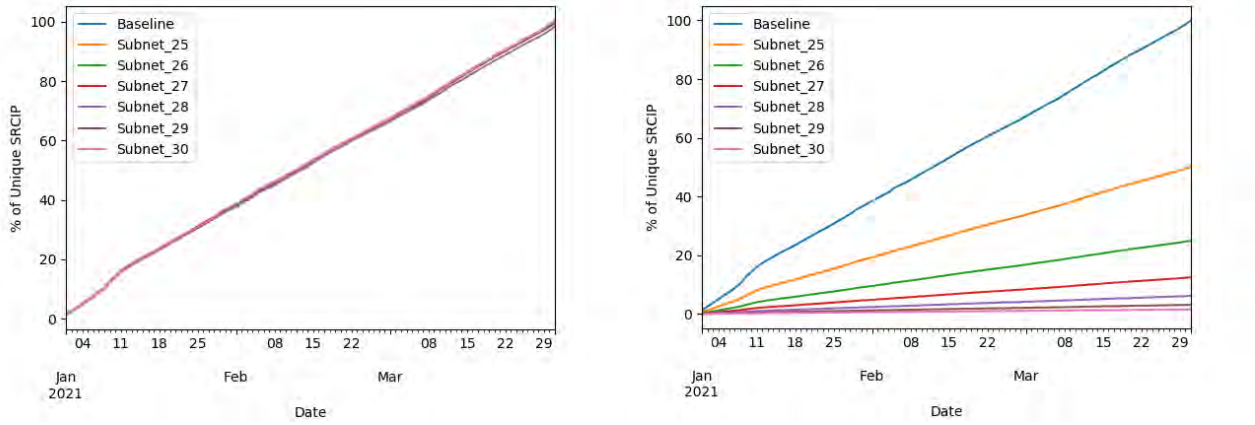
Timeline	/24	/e25	/e26	/e27	/e28	/e29	/e30
T_0	1.34	0.66	0.34	0.17	0.09	0.04	0.02
T_1	8.78	4.37	2.20	1.12	0.56	0.27	0.13
T_2	19.54	9.74	4.86	2.46	1.25	0.61	0.31
T_3	26.68	13.32	6.63	3.34	1.69	0.84	0.41
T_4	34.19	17.06	8.51	4.25	2.16	1.06	0.53
T_5	41.56	20.78	10.36	5.17	2.63	1.29	0.64
T_6	48.97	24.45	12.21	6.11	3.09	1.53	0.76
T_7	56.58	28.31	14.12	7.08	3.57	1.77	0.88
T_8	63.51	31.80	15.85	7.93	4.00	1.98	0.98
T_9	70.60	35.32	17.61	8.82	4.46	2.20	1.09
T_{10}	78.29	39.16	19.51	9.79	4.94	2.43	1.20
T_{11}	86.31	43.15	21.54	10.79	5.46	2.68	1.33
T_{12}	93.44	46.72	23.34	11.69	5.89	2.89	1.44
T_{LD}	100.00	49.97	24.97	12.52	6.32	3.10	1.54

Due to this duality of presentation, the explanation in this section will accommodate both the tables and the figures as they speak of the same data. Recall that the subnet and subnet equivalent represent different sizes of network telescopes, with subnets representing sequentially collected DSTIPs while subnet equivalent representing randomly sampled DSTIPs. Recall that T_0 in the tables represent week zero (also referred to as the first day of observation) while T_{LD} represent the last day of observation. Apart from T_{LD}, the spacing between the timeline in the tables (i.e. between T_0 and T_1) was placed to 7 days which represent a week. Thus when this study mentions T_3 for example, it is referring to observations made in week 3. T_{LD} was included because the study had to show when all the unique SRCIPs were observed. In all network telescopes, the total amount of unique SRCIPs was found on the last day of each month. In other words, any user who observes less than a month is likely to miss some unique SRCIPs as some of these unique SRCIPs only appeared once in the last day.

The proportion of the number of unique SRCIP observed at specific times for different network telescopes is similar. For instance, in all the cases presented in **Tables 7.3** and **7.4**, it took about two weeks (T_2) in each of the baseline datasets to observe about 20% of the total number of unique SRCIP addresses collected. To observe the same 20% proportion in a /25 or $/_e25$ network telescope, one would need to observe for at least five weeks (T_5). It takes almost 10 weeks (T_5) to observe the same proportion (20%) for /26 or $/_e26$ network telescope. Essentially, small network telescopes take a longer period to observe the same amount of traffic that can be observed in a larger network telescope. This can also be deduced from **Figures 7.1b** and **7.2b**. Only the baseline dataset collected a total number of unique SRCIPs. Thus, using **Tables 7.3** and **7.4**, a network telescope user can tell how long it would take to observe any specific proportion and be in a position to tell the effect that the size of a network telescope has on the amount of SRCIPs collected by each size. Any proportion can be used to explain the results. Here, 20% is just being used as an example.

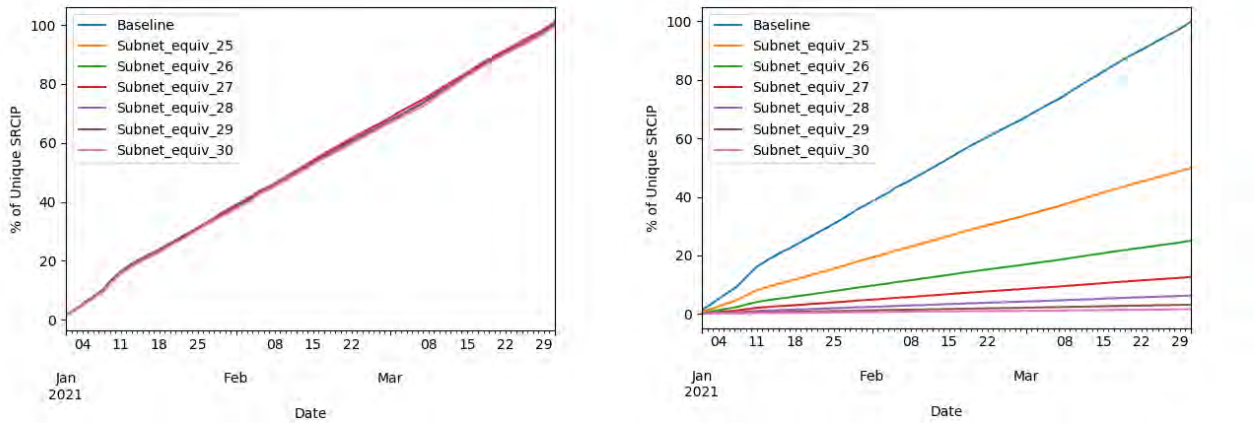
If these small sized network telescopes can be monitored for longer, they should be able to acquire the same capacity as the large sized network telescopes. If the observation period between large and same sized network telescopes is the same, then the small sized network telescopes are bound to collect significantly less data. This scenario is true when experimental set up was not normalised. However, the normalised datasets present these findings differently by presenting very identical results as can be seen in **Figures 7.1a** and **7.2a**. As explained earlier, the normalised datasets were computed against their total count.

Regarding the differences between random and sequential sampling of DSTIPs, the study did not observe significant differences as can been from both **Figures 7.1** and **7.2**, and **Tables 7.3** and **7.4**. However, following the recommendations made in **Section 6.5**, the study maintains its recommendation that random placement of unique DSTIPs should be the first priority. These findings observed in Network telescope **146/8** are not unique to this network telescope as the study also shows results observed in Network telescopes **196-A/8** and **155/8**. **Figures 7.3** and **7.4** show quarterly time series plots that portray a similar pattern as that observed in Network telescope **146/8**.



(a) Normalised (b) Not Normalised

Figure 7.3: 155/8 -[Jan - Mar]: Unique SRCIPs over time [Sequential]



(a) Normalised (b) Not Normalised

Figure 7.4: 155/8 -[Jan - Mar]: Unique SRCIPs over time [Random]

Table 7.5: 196-A/8-[Jan - March]: Sequential Cumulative % Summary Table

Timeline	/24	/25	/26	/27	/28	/29	/30
T_0	1.32	0.65	0.34	0.16	0.08	0.05	0.02
T_1	8.90	4.47	2.23	1.10	0.56	0.27	0.14
T_2	19.03	9.56	4.74	2.36	1.19	0.58	0.30
T_3	26.18	13.13	6.53	3.27	1.63	0.81	0.42
T_4	33.84	16.97	8.45	4.22	2.12	1.06	0.53
T_5	41.51	20.76	10.35	5.20	2.63	1.30	0.66
T_6	49.10	24.57	12.26	6.13	3.10	1.55	0.78
T_7	56.87	28.44	14.19	7.13	3.59	1.79	0.90
T_8	63.92	31.98	15.94	8.03	4.02	2.01	1.01
T_9	70.95	35.51	17.68	8.92	4.46	2.23	1.12
T_{10}	78.55	39.35	19.55	9.86	4.93	2.46	1.24
T_{11}	86.46	43.32	21.54	10.86	5.41	2.71	1.36
T_{12}	93.60	46.90	23.30	11.77	5.87	2.93	1.47
T_{LD}	100.00	50.12	24.89	12.55	6.26	3.14	1.58

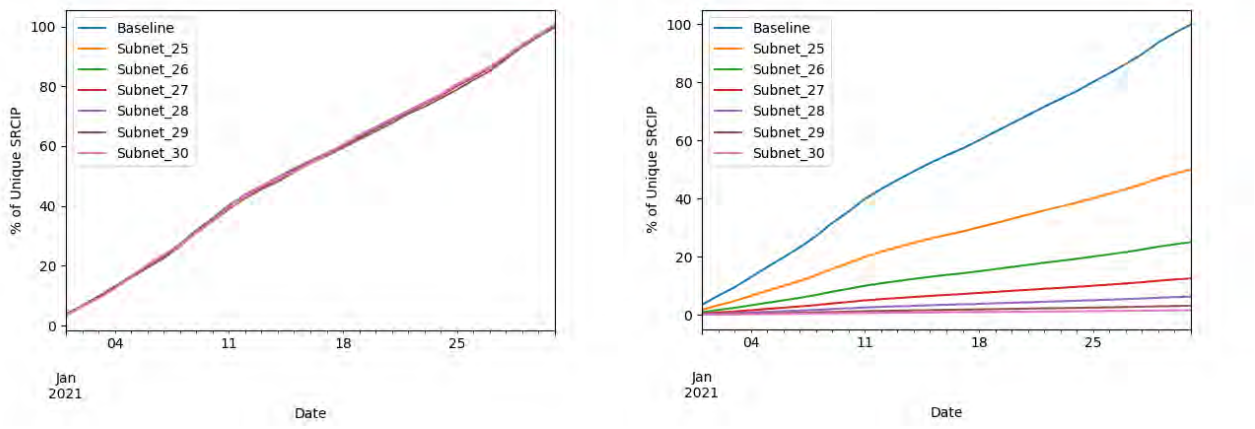
Table 7.6: 196-A/8-[Jan - March]: Random Cumulative % Summary Table

Timeline	/24	$/_e25$	$/_e26$	$/_e27$	$/_e28$	$/_e29$	$/_e30$
T_0	1.32	0.66	0.33	0.16	0.08	0.04	0.02
T_1	8.90	4.44	2.23	1.11	0.54	0.28	0.13
T_2	19.03	9.76	4.93	2.48	1.22	0.63	0.30
T_3	26.18	13.34	6.72	3.38	1.66	0.84	0.41
T_4	33.84	17.08	8.59	4.32	2.13	1.06	0.53
T_5	41.51	20.80	10.39	5.26	2.59	1.30	0.64
T_6	49.10	24.54	12.30	6.21	3.08	1.52	0.76
T_7	56.87	28.32	14.18	7.18	3.54	1.76	0.87
T_8	63.92	31.76	15.92	8.07	3.97	1.98	0.98
T_9	70.95	35.32	17.70	8.98	4.42	2.20	1.09
T_{10}	78.55	39.17	19.61	9.92	4.90	2.45	1.21
T_{11}	86.46	43.11	21.58	10.92	5.41	2.71	1.34
T_{12}	93.60	46.73	23.38	11.82	5.86	2.92	1.45
T_{LD}	100.00	50.02	25.01	12.65	6.27	3.14	1.56

These findings are true for both normalised datasets and those that were not normalised. On the other hand, **Tables 7.5** and **7.6**, show results observed in **196-A/8** Network telescope. By the ninth week (T_9), both **146/8** and **196-A/8** had observed about 70% of the total unique SRCIP received. In **Table 7.4**, the study showed that T_{LD} for $/_e27$ was 12.52 %, while in **Table 7.6**, T_{LD} for $/_e27$ is 12.65 %. Whatever value is picked from

any network telescope when compared against a different network telescope, on the same sized network telescope, the difference is less than 1%.

Now this is some actionable intelligence that one can work with when it comes to planning. Using the proportions presented and the timeline involved, any network telescope user can compute the actual count of unique SRCIPs collected in their network. The crucial part is also on how these findings, in particular the proportions, collaborate with the findings observed in **Section 7.1**. A network telescope user can use the timeline presented here and the estimated proportions coupled with the CI provided in **Section 7.1** to compute the estimated amount of unique SRCIPs they may have missed out and impute the difference to make estimates. The study extended the experiments by conducting a monthly analysis with the aim of understanding the relationship between the sample size and the timeline it takes for each of the samples to collect the maximum amount of unique SRCIPs. **Figure 7.5** shows the graphical representation of monthly data for **196-A/8** Network telescope. The pattern observed is similar to that observed in quarterly data. Adding the monthly datasets to the analysis of the time series plots shows that the duration of collection and size of the network telescope affects the proportion of the unique SRCIP addresses collected. A longer duration will still collect more unique SRCIPs than a smaller duration, and larger network telescopes will still collect more threat intelligence than smaller network telescopes. Two network telescopes of the same size that are monitored for the same duration can show a similar proportion but have different actual count of unique SRCIPs.



(a) Normalised (b) Not Normalised

Figure 7.5: 196-A/8-012021: Systematic Time Series Plot of Unique SRCIP/DSTIP

Table 7.7: 146/8-012021: Sequential Cumulative % Summary Table

Timeline	/24	/25	/26	/27	/28	/29	/30
T_1	3.45	1.73	0.86	0.44	0.20	0.10	0.06
T_1	22.89	11.63	5.60	2.83	1.41	0.71	0.37
T_2	50.43	25.39	12.52	6.29	3.12	1.60	0.75
T_3	69.53	34.89	17.36	8.75	4.28	2.16	1.04
T_4	90.02	45.20	22.41	11.26	5.64	2.81	1.35
T_{LD}	100.00	50.14	24.95	12.51	6.28	3.12	1.50

Table 7.8: 146/8-012021: Random Cumulative % Summary Table

Timeline	/24	/e25	/e26	/e27	/e28	/e29	/e30
T_0	3.45	1.70	0.87	0.44	0.23	0.11	0.05
T_1	22.89	11.38	5.74	2.91	1.45	0.70	0.35
T_2	50.43	25.11	12.54	6.34	3.22	1.58	0.80
T_3	69.53	34.65	17.29	8.69	4.39	2.19	1.09
T_4	90.02	44.91	22.42	11.19	5.69	2.80	1.40
T_{LD}	100.00	49.98	24.96	12.44	6.33	3.11	1.55

Table 7.9: 155/8-022021: Sequential Cumulative % Summary Table

Timeline	/24	/25	/26	/27	/28	/29	/30
T_0	4.47	2.23	1.14	0.54	0.29	0.14	0.07
T_1	26.01	13.03	6.50	3.20	1.61	0.84	0.44
T_2	50.30	25.22	12.65	6.20	3.09	1.63	0.78
T_3	75.01	37.66	18.78	9.26	4.68	2.37	1.17
T_4	100.00	50.19	25.03	12.38	6.21	3.13	1.57

Table 7.10: 155/8-022021: Random Cumulative % Summary Table

Timeline	/24	/e25	/e26	/e27	/e28	/e29	/e30
T_0	4.47	2.21	1.12	0.56	0.28	0.15	0.07
T_1	26.01	13.07	6.52	3.24	1.67	0.80	0.43
T_2	50.30	25.22	12.55	6.36	3.18	1.57	0.79
T_3	75.01	37.56	18.69	9.54	4.73	2.35	1.17
T_4	100.00	50.04	25.01	12.70	6.31	3.12	1.59

As with the quarterly data, **Figure 7.5** is supported by **Tables 7.7 - 7.10**. January and February data from **146/8** and **155/8** network telescope are used. Due to the short duration of observation, the proportion of unique SRCIP observed on weekly basis has slight increased. For instance, in quarterly observation T_2 showed less than 20 % of unique SRCIP observed, while in monthly data (**Tables 7.7 - 7.10**), T_2 shows at lease 50 % of the total amount. This does not mean that monthly data contain more unique SRCIPs as compared to quarterly data. It simply means that small periods of observation have high proportions per observable period. Observable period in this research was a week.

Furthermore, the study also observed that, at the beginning of each collection period (be it one month or three months), there were a handful of unique SRCIP addresses that were responsible for the large proportion of traffic sent to the network telescopes. The study had initially assumed that at the beginning of the data collection period, there will be a lot of unique SRCIP which will eventually phase out and that, before the last date of data collection, all the unique SRCIP will have been observed. However, from the plots, one can note that there was a steady increase in the unique SRCIPs added to the network telescope. New unique SRCIPs kept on appearing in the network until the last day of data collection. This is true for both monthly and quarterly observations. Three weeks into the data collection for the month of January and February for **146/8** and **155/8** network telescope, about 70% of the total number of unique SRCIPs had been observed. It took nine weeks for the same proportion of unique SRCIPs in quarterly datasets.

Thus, using the time series plots and the tables presented in this section, one can decide how long they would want to observe network traffic in their network in order to collect a certain proportion of the overall parameter of interest. Depending on the volume of unique SRCIPs that they would want to their research, they would observe for that duration using their network telescope. Thus the tables presented in this section offer the artefacts that any network telescope user can use to understand how long it would take to observe specific proportions of unique SRCIPs. In addition to this, the artefacts also help a network telescope user in understanding how the specific proportions are affected by the different sizes of the network telescope. Overall, the reader will note that having gone through the tables and plots, the major take away is that the progression over time is similar in all network telescopes at any given point

7.3 Cross Disciplinary Application of the Models

Having looked at the models developed for the IBR datasets and how they performed in **Chapter 6**, the study opted to lay out the cross disciplinary applications of these models in the real world. One of the main discoveries during the experimental stages was a clear understanding that these models are multi-disciplinary in nature. This means that the applications of the models are not limited to computer science or IBR data, but they can also be used in other fields like Biology, Chemistry, Physics, Statistics, population studies, geology, and in the medical field, among others. Any specialised field that works with statistics, sampling, and data samples has the opportunity to benefit from these models and assess how the various samples they have taken from a baseline (original data) relate to each other, and the primary data. It does not even have to be time-series data, however, it needs to have a series of data points per sample. The guiding principle here is the ability of the researchers to be able to standardise the various data samples to a well-known unit of measurement. In this study, the unit of standardisation was the number of unique DSTIP addresses found in a subnet or subnet equivalent.

Thus in this section, the study looked at how the derived models can be applied to some of the research fields. In establishing the usability of the models in the various fields described in this section, the primary investigator conducted a series of informal discussions with researchers who are experts in their respective fields. These are researchers who are either doing their post-doctoral research or lecturing, or are in their final year of PhD studies and have had their work peer-reviewed (published articles). This approach was opted for because of the knowledge base of the participants being engaged, which cannot be found in one or two papers, but after years of experimental work. That way, the author got to appreciate what role the developed models can do in each of these fields. The author also had to verify the methods used, hence adding the footnote to support their narrative. The fields presented in this section are not exhaustive, but they are just a sample of the fields the models developed can work in and they demonstrate the usability of the models in a real-world scenario.

7.3.1 Application of the Models in Demography

Demography is the scientific study of human populations primarily with respect to changes in their size, structure and development over time (Rees, 2020). The people involved in this study are called demographers. They use data collected about a specific population

or groups at a point in time with respect to well-defined characteristics (this data is referred to as census data). The overall objective is to understand populations' dynamics (structure and change), the factors behind their dynamics, and the consequences of these changes to the entire population.

At any given point in time, each population is distinguished by the number of people it has (size) and the population composition, which constitute the age of the population, the number of males and females, employment status, and level of education, among other features (Rees, 2020). All these attributes within a population change constantly as people arrive (through birth or immigration), while others depart from the same population through death or emigration. There are a lot of other statistical variables that demographers analyse in order to make well-informed decisions. Among such variables include: calculating fertility, mortality and migration rates by sex and age in order to identify the functions (in the mathematical sense) of fertility, mortality, and migration that determine change in a population[1].

By understanding these variables, coupled with population models, demographers come to a clear understanding of the relationship between the demographic structure of different populations given their distinctive features (Rees, 2020). Thus, demographers use census data, surveys, and statistical models to analyze the size, movement, and structure of populations at that particular point in time in order to make predictions and projections in line with the resources available to that population[2]. Decisions are made by countries or organisations based on such findings.

The census data can equally be treated the same way as IBR data since it also has distinct features that define it. For instance, census data has a time stamp making it fit in the category of time series data. Secondly, demographers use this data to make predictions, thus MAPE, SMAPE, MASE and MAE can be used to make forecasts with this data. However, there is one component that may need to be looked at; how representable different samples within a population are to the overall census data. This is where surveys come in since they only focus on sample populations. Usually, this happens because of financial limitations, and so a handful of people in a country are picked. It could be based on districts, villages, or cities. In order to verify how accurately the survey data collected represents the interest of the entire population, AMAPS, SAMAPS, SMAE and SMASE can be deployed to this data.

Firstly, they will need to identify which unit of standardization they will need to use for

[1]https://iussp.org/en/about/what-is-demography
[2]https://iussp.org/en/about/what-is-demography

this data. They could use the number of individuals per household to standardise survey data and verify how this compares with the census data. They could also use the number of deaths per square kilometre in order to assess how death is affecting the population. They could use the number of educated people either within a household or a specified area as a standardisation unit. Once they find the standardisation unit, they can apply it to see how well the sample contained in the survey data represents the census data. There are many features that the demographers can use to standardise survey data to make it comparable to census data, but once this step is accurately done, AMAPS, SAMAPS, SMAE and SMASE can be deployed. Due to the statistical nature of demography, by virtue of using this example, the models do apply and fit in the discipline of statistics, a field that has proven to be vital across all scientific fields.

7.3.2 Application of the Models in Ichthyology

Ichthyology is the branch of zoology that is dedicated to the study of various kinds of fish with the aim of understanding their biological make-up, taxonomy, and conservation (Kapoor and Khanna, 2004). The study also extends to husbandry and commercial fisheries. Like all animals, fish also have their own dynamics that get to affect their own livelihood. These include: diseases that are very specific to certain breeds, extinction of certain species from specific water bodies, crossbreeding, and the life expectancy of different breeds, among others (Kapoor and Khanna, 2004). These are just some of the dynamics that researchers in ichthyology get involved in. In order to tackle some of these problems, a full study has to happen.

In this study, the primary researcher had an informal discussion with one of the PhD students who was involved in collecting data from different countries in order to understand why certain breeds of fish are disappearing in certain areas while the same breeds are thriving in other parts. In this interview, the researcher went on to argue that different areas get to be affected differently depending on the variables they have found in a specific area. For instance, some ponds or water bodies tend to have a lot of species of fish while others have very few of the same species. In the process of crossbreeding, new kinds tend to formulate while others tend to disappear. The size of the pond also has an effect on the survival and extinction of fish as well as their life span. He went on to state that there are also certain kinds of diseases that can only affect certain species while others tend to be immune to the same disease. So in the process of collecting data from various locations, they are able to understand how all these variables affect the fish in all of these locations.

Our main interest in this conversation was to try to understand how he is able to compare these different samples given that the areas are different. The interviewee iterated that, in order to compare samples collected from different locations, certain parameters have to be uniform. For instance, the number of fish of a specific kind collected per given size of the bond has to be the same; meaning that the size of the pond, as well as the species of fish collected, has to be the same. Secondly, the fish involved in the cross-breeding need to be of the same age as well as species i.e. if a fish belonging to family A and B were collected at one location, he needs to find another pond (or water body) that has the same kind of species and the same size of the water body in order to make the results comparable. The last question the student was asked was on whether these results can be interpolated to represent a whole area. He cautiously responded that if they are limited in terms of finances, then they are forced to interpolate the findings from one water body to represent the whole area.

Using this interview with this researcher, we also saw how the developed models can be of use in Ichthyology. For starters, the data collected here by the student had no time stamp. In other words, time did not have a bearing on his findings. Secondly, in the interview it was clear which unit of standardisation can be used specifically for his study. He mentioned that the size of the pond, as well as the number of fish that belong to a specific species, need to be taken into account in order for different samples to be comparable. Thus using AMAPS, SAMAPS, SMAE and SMASE, an ichthyologist can standardise the data collected and evaluate how the different samples collected from the different locations compare to each other. That way, before making extrapolations, the researcher can know with some degree of certainty (accuracy scores from AMAPS and SAMAPS) how the different samples compare to each other and the errors involved. Using SMAE and SMASE, the researcher will know for sure if the samples properly represent the data collected from the area of origin if SMAE and SMASE scores are below the value of 1.

7.3.3 Application of the Models in Biochemistry

Biochemistry is a branch of science that explores the chemical processes happening at a molecular level within living organisms by conducting a series of laboratory experiments and joining knowledge and techniques from biology and chemistry in order to solve real-world biological problems (Banani *et al.*, 2017). Biochemistry focuses on what's happening inside our cells by studying components like proteins, lipids, and organelles[3]. It

[3]https://www.mcgill.ca/biochemistry/about-us/information/biochemistry

also looks at how cells communicate with each other, for example, during growth or when fighting illness. Biochemists (people who study biochemistry) need to understand how the structure of a molecule relates to its function, allowing them to predict and understand how molecules will interact (Banani *et al.*, 2017). Biochemistry covers a range of scientific disciplines, including genetics, microbiology, forensics, plant science, and medicine[4]. Because of its breadth, biochemistry is very important, and advances in this field of science over the past 100 years have been staggering[5].

This research's primary investigator engaged with one of the researchers in the field of biochemistry, who specialised in developing chemical compounds (drugs) for diabetes, and also does intensive work with cancer research. In the interview with this researcher, the primary interest was in understanding how data sampling works in their field: where they get the samples, how many samples constitute a full study, and more importantly, how they compare multiple samples. This researcher explained his position using an example of diabetes where they collect saliva, for example, as a single sample. They would collect multiple samples from different individuals with the same type of diabetes. However, he did mention that even when the samples are from the same diabetes type, things like the time the patient ate their last meal before the sample was collected, age, sex, among other variables, need to be taken into account. These all form part of the analysis. However, assuming that all these variables are the same, there is also the part where each individual is different from the next, and that creates new variables altogether.

This now led us to ask the main question: how then does one compare the different samples from different individuals? How is the data collected from the experiments standardised? He went on to explain that, the standardisation happens with the chemical compound (drug) they are developing. They make sure that all the data samples collected are treated equally by ensuring the concentration of the chemical compound used as the treatment is the same. From the concentration, they also assess the time taken by the compound to achieve the intended results using the same concentration for all samples. They also ensure that the volume of the chemical compound created (which is the main output of the experiment) is kept constant for all the samples being tested.

Thus for biochemistry, one way of ensuring that the data collected from different samples is comparable is by making sure that the different samples are given the same volume of the chemical compound being developed as treatment, but also ensuring that the level of concentration is the same for all samples. In other cases, they collect multiple samples

[4]https://biochemistry.org/education/careers/becoming-a-bioscientist/what-is-biochemistry/
[5]https://biochemistry.org/education/careers/becoming-a-bioscientist/what-is-biochemistry/

for the same individual, but ensure that they assess how the volume of each chemical compound developed is treated by the cells of this individual. Alternatively, they assess how the different levels of concentration affect the treatment being offered to the specimen under study. In all of these cases, there is standardisation involved, and once a well-known method of standardisation is applied to the data, it will offer biochemists an opportunity to find new ways of quantifying the performance of the developed treatment on various samples. This allows them to compare multiple samples by using AMAPS, SAMAPS, SMAE and SMASE to quantify the differences. Note that this is just one way with which data is standardised in this field and also one way with which they deal with data samples. Biochemistry is a broad field and thus the applications of this model in this field are only as variant as the ability of the biochemist to find ways of standardizing the data.

7.3.4 Application of the Models in Geology

Geology is the study of the Earth, the materials of which it is made, the structure of those materials, and the processes acting upon them (Dai and Finkelman, 2018). It includes the study of organisms that have inhabited our planet. One of the core elements in the study is how Earth's materials, structures, processes and organisms have changed over time. However, their time is not measured as is done in other fields that work with changes in 20 or 5 years, or months. In geology, they deal with millions or billions of years (Friederich and van Leeuwen, 2017), referred to as geologic time[6]. In such a setup, time series analysis would not fit. However, through the processing of their data, they get to have a series of data points that take even years to process.

In the process of assessing how the developed models work, we engaged one of the geologists who has worked in the field for over 15 years before moving into academia. He gave us an example of how the sampling works when they are processing coal as he believed it would be less complicated to explain without a lot of jargon. He gave an example of a process they refer to as proximate analysis, which tends to measure four core elements: calorific value, moisture content, ash content, and volatile matter (Nunes *et al.*, 2018). Usually, this analysis is done when an investor is trying to assess which coal has the best of the four elements among samples from multiple sites.

All these variables are critical because an investor would set specific values for each of these elements. The value they set becomes the standard with which the geologist will

[6]https://www.livescience.com/why-geolotic-time-periods.html

have to work with in all the multiple sites. In each area, they look for things like the amount of weight needed to get to use for a series of experiments. The size of the area where the coal was taken from is also considered. After a series of experiments from the numerous samples they collected from different areas, they will need to do a comparative analysis to assess how the different samples compare to each other before a final decision is reached for the investor. This is where AMAPS, SAMAPS, SMAE and SMASE can be used to quantify the differences in different areas. Thus the models become an extra statistical tool to aid in the analysis. One of the reasons geologists are heavily involved in sampling is to cut on costs, and thus being certain of their decision is key. As such, having more tools to quantify the difference adds more confidence to both the investor and the geologist.

7.4 Summary

In this chapter, some of the practical applications of the research findings were presented. An important element to note is how the applications presented herein are not only limited to the use of network telescopes, but go beyond that. This chapter covered concepts from **Chapters 5** and **6**.

The chapter starts off with **Section 7.1** by presenting how bootstrapping can be applied to IBR data with the aim of offering levels of confidence to users. The confidence revealed by the dataset is what the network telescope user would need in order to make informed decisions. From here, the study presented different scenarios relating to the efficiency of different network telescope lenses in collecting threat intelligence data over different time frames. This is presented in **Section 7.2**. In this section, the study was offering practical scenarios that come with using different network telescope lenses and the effect that time had over such collection periods. The chapter closed by offering an external application of mathematical models that were developed to quantify differences that exist between different network telescope lenses. The idea behind this is to open a wider perspective to the applicability of the models developed and the knowledge gap that it closes. All of this is presented in **Section 7.3**.

8

Conclusion

This research evaluated the effectiveness of using small aperture network telescopes as IBR Data sources. This was achieved by assessing three different network telescopes that recorded a total of **108,309,459** events from *January 2021* to *March 2021*. Note that the results presented in this document only accommodate the findings of the final analysis of the study. To achieve the objectives set at the beginning this research, the study used both randomly and sequentially sampled datasets in order to create subnets and subnet equivalents. These represented different sizes of network telescope sensors. The researcher was fully aware that any data sample taken from the baseline dataset cannot fully replace the baseline data. In this case, a sample of /24 IPv4 cannot replace its baseline dataset. However, finding alternative small sized network telescopes that offer a high level of confidence by drawing samples from the baseline would be a better fit as compared to completely stopping the use of network telescopes for those who do not have adequate IP addresses. The argument in this research document has been that a small sized network telescope can be used to collect IBR data which can later be processed into threat intelligence. Largely, this motivation was inspired by inadequate IPv4 addresses that can be used for passive monitoring.

To achieve the research objectives of this study, a number of analyses were conducted in order to quantitatively state how well the smaller network telescope lenses represented the /24 IPv4 baseline dataset. In order to successfully quantify the differences that exist between the baseline and subnet/subnet equivalents, four mathematical models were applied (AMAPS, SAMAPS, SMAE and SMASE). In **Section 6.4** the study showed that smaller subnets can represent the content of the baseline datasets with up to 95% accuracy. The study also simulated IBR data to add levels of confidence to the users. Computations of CI range from 80% to 99%, confident that if the experiment set was repeated multiple times the outcome will fall with the same interval. The study also presented a model that shows the effect that time has when it comes to threat intelligence gathering.

This chapter thus concludes this research by offering a quick recap of what has been covered in **Section 8.1**. This is immediately followed by an evaluation of the research objectives in **Section 8.2**. This was done to assess the extent to which these goals were met in addressing the research questions. The study's recommendations are also added in this section. Research contributions are presented in **Section 8.3** and thereafter, future work based on the work done in this study is presented in **Section 8.4**

8.1 Document Summary

This document began by introducing the research problem and offered adequate background to the research questions in **Chapter 1**. This chapter also laid down the research objectives and research approach to be followed. This is followed by **Chapter 2** which provides a literature review on network telescopes and the necessary terminology needed to justify the use of a small-sized network telescope. This chapter also introduced the nature of the data used in this study i.e. time-series data. It explains how IBR data is collected and explains why it is important. The chapter builds further details on the research problem by explaining the current crisis of IPv4 exhaustion.

In **Chapter 3**, the study introduced the statistical techniques that were used in this research. Mostly, this chapter laid down the theoretical foundation from which the mathematical models were derived. Data sampling, bootstrapping, confidence interval, mathematical modelling, regression analysis, and information retrieval techniques were all introduced and an explanation was offered on how they were used. Similarity scoring techniques, which form the core of the quantification techniques in this study, were explained here as well.

Chapter 4 introduced and explained all the datasets that were used and where the data was collected from. How the data was processed and the data sampling techniques used in this study were also explained in this chapter. Summary statistics of the data and its graphical representations were presented here as well. A breakdown of the datasets and their composition also formed part of this chapter.

Chapter 5 presented a simulation technique called bootstrapping. An explanation of why this was important to this study was presented. A research approach of how bootstrapping was conducted was also explained here. This chapter further presented a regression analysis and how confidence intervals computed through bootstrapping can offer confidence to the network telescope user. This is also where parametric and non-parametric bootstrapping techniques were implemented and their results explained.

Chapter 6 started with the mathematical models that were developed to answer some of the research questions. An approach that was used to process and test the data against these developed models was presented here. Intense testing and evaluation against existing models were also conducted in this chapter. This study primarily focused on DPORTs and unique SRCIP addresses. Sequential and random datasets, collected both monthly and quarterly, were presented here. The chapter also presented the strengths and limitations of the mathematical models developed.

Having conducted multiple analyses on the datasets using various techniques, the study presented some of the practical applications that this study has offered to the research community in **Chapter 7**.

8.2 Evaluation of Research Goals

In **Section 1.3** this document laid out the objectives that it aimed to achieve in order to measure its success or failure. These included evaluating small aperture network telescopes for threat intelligence gathering using IBR data. This section thus evaluated the degree to which these objectives have been met.

1. The first goal set out was to assess whether there is a continual direct relationship between the number of unique SRCIP addresses observed against the number of unique DSTIP addresses after normalization. It was found that large subnets (which represent larger, more traditional network telescope sensors) still collect more unique

SRCIP addresses than small-sized network telescopes. This has been presented in **Sections 6.4**, **6.5** and **7.2**. However, when the datasets are normalised based on the size of the subnet, this study found that although larger subnets and subnet equivalents collected more unique SRCIP addresses than small-sized network telescopes, the proportion of the number of unique SRCIP addresses is *independent of the size* of the network telescope.

2. The second research objective was to compute the time frame needed to acquire specific proportions of the unique SRCIP addresses from the baseline data. The study presented plots that measured (show) the proportion of unique SRCIP addresses collected over time in **Section 7.2**. The plots show both random and sequential samples and different network telescope lenses. The study presented two different datasets; one for a month and another spanning over three months. To support the time series plots (see **Figures 7.1 - 7.5** in **Section 7.2**), the study presented its findings in **Tables 7.3 - 7.6** for quarterly data and **Tables 7.7 - 7.10** for monthly data as artefacts that address this research objective. In each case, a network telescope user can know the proportion of unique SRCIPs collected by their network telescopes at any given point in time of data collection. Thus a network telescope user can compute the time frame needed to collect a specific proportion of unique SRCIPs for any network telescope sensor size. The study found that in both monthly and quarterly datasets, new unique SRCIPs were present until the last day of observation. Larger network telescopes took less time to observe specific proportions of the unique SRCIPs than small-sized network telescopes. i.e. larger network telescope sensors collect more data than small-sized network telescopes. Although the percentage proportions of unique SRCIPs are very similar for different network telescopes in their observation period, the actual count of unique SRCIPs in the network telescopes is different. This is true for different networks as well. With these findings, *the study recommended in **Section 6.5** that longer period of observation should be considered as the findings shows that given more time small-sized network telescopes can amass unique SRCIPs.*

3. The third objective of this research was to identify how accurate a small-sized network telescope lens is at representing /24 IPv4 network telescope. The study identified that different network telescope lenses offer different levels of confidence in the data collected. Small-sized network telescope lenses offer small confidence intervals than larger network telescope lenses. Thus, based on the level of confidence offered by different sized network telescopes, a user can pick which one is convenient for them so long as they know how much data they would be missing out on. The confi-

dence interval and level shows the network telescope user that if they were to repeat the same experiment multiple times, this study is confident that using the identified confidence level the CI will fall within the same range. In **Section 5.5**, the study presented different CIs computed at different levels with 95% CI - 99% CI being recommended. The recommendations for bootstrapping IBR data and the confidence level to be used are presented in **Section 5.7**. An artefact from bootstrapping was presented using **Tables 5.19** and **5.20** in **Section 7.1**. The CI computed offers the desired confidence a user needs to have prior knowledge to decide which size is convenient for them. It is this CI that necessitates whatever decision needs to be made. In **Chapter 6**, the study quantified the differences that exist between different network telescope sizes. The smallest sized network telescopes offered at least 92% similarity to the baseline, while the largest (/25 subnet or $/_e25$) offered at least 98% similarity with the baseline. This is when the datasets are normalised using the size of their subnet. Randomly sampled unique DSTIPs gave better scores than sequential samples, thus in **Section 6.5**, t*he study recommended random placement of DSTIPs in a network telescope as opposed to sequential placement.* The actual differences have been presented in **Section 6.5**. In the same section, the study recommended that network telescopes should not contain less than 32 DSTIPs because acceptable scores for SMAE and MAE ought to be below 1. The advantage of random placement is that a network telescope user does not need reserve contiguous block to use for monitoring traffic.

4. Lastly, this study planned to evaluate the differences that exist when the IPv4 addresses in the network sensors are randomly selected compared to when the IPv4 addresses are selected in contiguous blocks. In order to achieve this, mathematical models were developed to quantify the differences that exist between the baseline and the subnet or subnet equivalent. Using the number of unique SRCIP/DSTIP as the basic unit of analysis, the study concluded that randomly sampled datasets performed slightly better than contiguously sampled datasets (sequential samples). In **Section 6.4**, the study showed both graphically and quantitatively that subnet equivalents performed slightly better than subnets. This, in part, could largely be attributed to the almost even distribution of unique SRCIP in the subnets. For AMAPS and SAMAPS scores of accuracy, randomly sampled DSTIP addresses recorded as high as 99.15% accuracy match with the baseline dataset for $/_e25$ while /25 subnet for the same datasets recorded a 98.41% match with the /24 IPv4 baseline dataset. The smallest samples from /30 subnet recorded 90.76% accuracy match with the baseline for AMAPS and SAMAPS while $/_e30$ recorded an accuracy score of

94.19%. Thus in **Section 6.5** the *study recommended random placement of DSTIPs.*

8.3 Research Contribution

The findings of this research have been published in a number of conferences. The papers published include Chindipha and Irwin (2017); Chindipha *et al.* (2018, 2019a,b); Chindipha and Irwin (2021). The primary research contributions made by this study are:

1. The development of four mathematical models (AMAPS, SAMAPS, SMASE and SMAE) that quantify the variations that exist between different samples. These models have been presented in **Section 6.1**. The benchmark for their formulation was laid out by Hyndman and Koehler (2006) when they designed mathematical models to be used to forecast time series data. However, in this research document, this study derived novel models from this that can now be used on both time series and data without time stamps. **Section 6.8** details how both time series and non time series data can use these novel models. In addition to this, these models have a different applications than those intended by Hyndman and Koehler (2006). This information has been presented in **Section 6.1**.

2. In **Section 7.2**, the study has produced time series plots (see **Figures 7.1 - 7.5**) that show the expected proportion of unique SRCIP addresses over time. These time series plots are supported by **Tables 7.3 - 7.10**. **Section 7.2** also explains how the time series plot and the tables can used by other researchers in the field. Both the time series plots and the tables are vital for planning purposes as one would know how many unique SRCIPs can be collected by their different network telescopes in a specified timeline. Using these artefacts, network telescope users can now calculate how long it would take them to collect 30% of the unique SRCIPs for example, and work with that knowing fully well what the data represent. They would also know how their currently small-sized network would collect in comparison to the baseline in that same timeline. This would help them to know how long they would need to monitor their network telescope to get the same amount of data as the larger network telescopes. In **Section 2.4**, this research document presented different use cases of IBR data, thus using the collected unique SRCIPs a network telescope user can use such the data collected and extract threat intelligence data they ought to expect in the given timeline. The author of this research document published similar work of IBR use cases that can be found in Chindipha and Irwin (2017). In this

paper, Chindipha and Irwin (2017) did an analysis on the re-emergence of the SQL Slammer worm using IBR data.

3. The study has successfully quantified the differences that exist between large network telescopes and smaller network telescopes in **Section 6.3** and **6.4**. It was known that large network telescopes collect more data than small-sized network telescopes. However, no study, to the best of the author's knowledge, has ever quantified such a knowledge gap. With this additional knowledge to the research body, small-sized network telescope users can use their network telescopes with full knowledge of the data gap that exists between different network telescopes.

4. An extension to the aforementioned contribution is the applicability of confidence interval to the different sizes of network telescopes. With this knowledge, network telescope users can plan accordingly, knowing what to expect in their different network telescope sizes. **Section 5.7** has presented recommendations that this has made in regard to bootstrapping IBR data while **Section 7.1** has presented some of the practical applications of bootstrapping IBR data. **Tables 5.19** and **5.20** presents the thesis' artefact to be used as guide in computing the averages per given sample at 95% CI. The confidence level attached to the CI assures the network telescope user that if they were to repeat the same bootstrapping process to their data, 95% of the times they bootstrap their data, their results will fall within the given range (CI).

8.4 Future Work

At the time of conducting this research, datasets from IPv6 network telescopes did not contain adequate data demanding the study's attention. Future work could look into assessing if it is possible to implement the research findings of this study to IPv6. Most of the work done herein focused on the TCP dataset because it contained the majority of captured events as discussed in **Section 4.1**. It would be interesting to observe if a similar pattern can be observed in UDP and ICMP datasets. In addition to this, future studies can also look into:

- The current network telescopes that are set up at Rhodes University were designed to be passive i.e. when TCP traffic comes in, it is not involved in the three-way handshake as in passive network configuration. The DSTIP addresses are only

responsible for receiving traffic and nothing more. Future work will aim at ensuring that the three-way handshake is enabled by ensuring that some DSTIPs respond to in coming network traffic. This is what Moore *et al.* (2004) proposed and stated that an active network telescope configuration is very much possible and collects more data than a passive network telescope. Moore *et al.* (2004) referred to these network telescopes as honeyfarm telescopes because they actively responds to some or all of the event request traffic using honeypots. This way, more data will be collected, giving us more data than what the current set-up offers. Note that, currently, about 85% of the traffic is TCP. Thus this set-up would improve the quality of the traffic collected.

- More analysis could be conducted if a network telescope larger than /24 IPv4 is presented. This can also be extended to accommodate other Open Source Intelligence datasets and test the tools developed in this study. It would also be interesting to compare the results found in Rhodes University's network telescopes with other network telescopes. During this study, the researchers attempted to acquire more testing data from other network telescope operators (like CAIDA) but were not successful. Financial constraints also prevented the buying in of more IBR data from other network telescope users to test such variation.

- Given resources, it would be worthwhile to test the research findings of this study with interspersed 'live' network telescopes of different sizes operated in a different region. The overall idea behind this study was to ensure that users are able to collect actionable data using the least possible number of unique DSTIP. This may enable them to present an actual working concept that could cement the viability of these findings in a real-world scenario. Questions like how long it takes to collect **x** amount of threat intelligence data can be tested in a real-world environment. Building on the findings of this study, assess the differences in the amount of threat intelligence data that can be collected by the differently sized network telescope. So far the study has present unprocessed data that can be collected by these differently sized network telescopes. Analysing and processing this data to assess if certain threats identified in the baseline dataset can still be identified in all the small sized network sensors under study could make a good research study.

References

Abramowitz, M. and Stegun, I. A. Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables. *National Bureau of Standards Applied Mathematics Series 55. Tenth Printing*, 1972.

Aggarwal, C. C. Towards Systematic Design of Distance Functions for Data Mining Applications. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 9–18. ACM, 2003. doi:10.1145/956750.956756.

Aizawa, A. An Information-Theoretic Perspective of TF–IDF Measures. *Information Processing & Management*, 39(1):45–65, 2003. doi:10.1016/S0306-4573(02)00021-3.

Akter, J. Bootstrapped Durbin-Watson Test of Autocorrelation for Small Samples. *ABC Journal of Advanced Research*, 3(2):137–142, 2014. doi:10.18034/abcjar.v3i2.39.

Al-Tamimi, H. A. H., Alwan, A. A., and Abdel Rahman, A. Factors affecting stock prices in the uae financial markets. *Journal of Transnational Management*, 16(1):3–19, 2011. doi:10.1080/15475778.2011.549441.

Archontoulis, S. V. and Miguez, F. E. Nonlinear Regression Models and Applications in Agricultural Research. *Agronomy Journal*, 107(2):786–798, 2015. doi:10.2134/agronj2012.0506.

Arkko, J. and Townsley, M. RFC 6127: IPv4 Run-Out and IPv4-IPv6 Co-Existence Scenarios. Technical Report 6127, IETF, May 2011. doi:10.17487/RFC6127.

Arlot, S. and Celisse, A. A Survey of Cross-Validation Procedures for Model Selection. *Statistics Surveys*, 4:40–79, 2010. doi:10.1214/09-SS054. ISSN: 1935-7516.

Atifi, A. and Bou-Harb, E. On Correlating Network Traffic for Cyber Threat Intelligence: A Bloom Filter Approach. In *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC)*, pages 384–389. IEEE, Washington, DC, USA, June 2017. ISSN 2376-6506. doi:10.1109/IWCMC.2017.7986317.

Bagdonavicius, V., Kruopis, J., and Nikulin, M. S. Nonparametric Tests for Complete Data. John Wiley & Sons, 2013. ISBN 978-1-84821-269-5.

Bahuguna, A., Bisht, R. K., and Pande, J. Country-level Cybersecurity Posture Assessment: Study and Analysis of Practices. *Information Security Journal: A Global Perspective*, 29(5):250–266, 2020. doi:10.1080/19393555.2020.1767239.

Bai, J. and Ng, S. Forecasting Economic Time Series Using Targeted Predictors. *Journal of Econometrics*, 146(2):304–317, 2008. doi:10.1016/j.jeconom.2008.08.010.

Bailey, M., Cooke, E., Jahanian, F., Myrick, A., and Sinha, S. Practical Darknet Measurement. In *2006 40th Annual Conference on Information Sciences and Systems*, pages 1496–1501. IEEE, 2006. doi:10.1109/CISS.2006.286376.

Bailey, M., Cooke, E., Jahanian, F., Nazario, J., and Watson, D. The Internet Motion Sensor-A Distributed Blackhole Monitoring System. In *Network and Distributed System Security Symposium(NDSS)*. Internet Society, Reston, USA, 2005. doi:10.1109/CISS.2006.286376.

Banani, S. F., Lee, H. O., Hyman, A. A., and Rosen, M. K. Biomolecular condensates: Organizers of cellular biochemistry. *Nature Reviews Molecular Cell Biology*, 18(5):285–298, 2017. doi:10.1038/nrm.2017.7.

Barakat, M. R., Elgazzar, S. H., and Hanafy, K. M. Impact of Macroeconomic Variables on Stock Markets: Evidence from Eemerging Markets. *International Journal of Economics and Finance*, 8(1):195–207, 2016. doi:10.5539/ijef.v8n1p195.

Barbosa, J. C. What is Mathematical Modelling? In **Lamon, S. J., Parker, W. A., and Houston, K.**, editors, *Mathematical Modelling*, pages 227–234. Woodhead Publishing, 2003. ISBN 978-1-904275-03-9. doi:10.1533/9780857099549.5.227.

Barnes, B. and Fulford, G. R. Mathematical Modelling with Case Studies: a Differential Equations Approach Using Maple and MATLAB. Chapman and Hall/CRC, 2011. ISBN 978-1-4822-4775-6.

Baty, F., Ritz, C., Charles, S., Brutsche, M., Flandrois, J.-P., Delignette-Muller, M.-L. *et al.* A Toolbox for Nonlinear Regression in R: The Package nlstools. *Journal of Statistical Software*, 66(5):1–21, 2015. https://EconPapers.repec.org/RePEc:jss:jstsof:v:066:i05

Beeharry, J. and Nowbutsing, B. Forecasting IPv4 Exhaustion and IPv6 Migration. In *2016 IEEE International Conference on Emerging Technologies and Innovative Business Practices for the Transformation of Societies (EmergiTech)*, pages 336–340. IEEE, 2016. doi:10.1109/EmergiTech.2016.7737362.

Bender, R. Calculating confidence intervals for the number needed to treat. *Controlled Clinical Trials*, 22(2):102–110, 2001. doi:10.1016/S0197-2456(00)00134-3.

Benson, K., Dainotti, A., claffy, k., Snoeren, A. C., and Kallitsis, M. Leveraging Internet Background Radiation for Opportunistic Network Analysis. In *Proceedings of the 2015 Internet Measurement Conference*, pages 423–436. 2015. doi:10.1145/2815675.2815702.

Benson, K., Dainotti, A., Claffy, K. C., and Aben, E. Gaining Insight into AS-Level Outages Through Analysis of Internet Background Radiation. In *2013 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pages 447–452. IEEE, 2013. doi:10.1109/INFCOMW.2013.6562915.

Bertino, E. and Islam, N. Botnets and Internet of Things Security. *Computer*, 50(2):76–79, 2017. doi:10.1109/MC.2017.62.

Blaise, A., Bouet, M., Conan, V., and Secci, S. Detection of Zero-day Attacks: An Unsupervised Port-based Approach. *Computer Networks*, 180:107391, 2020. doi:10.1016/j.comnet.2020.107391.

Blomhøj, M. Different Perspectives in Research on the Teaching and Learning Mathematical Modelling. In *Proceedings from Topic Study Group 21 at the 11th International Congress on Mathematical Education in Monterrey, Mexico, July 6-13, 2008*, volume 1, pages 1–18. 2009. ISSN 0106-6242.

Blum, W. Quality teaching of mathematical modelling: What do we know, what can we do? In *Proceedings of the 12th International Congress on Mathematical Education*, pages 73–96. Springer, Cham, 2015. ISBN 978-3-319-12688-3.

Bora, A. and Ahmed, S. Mathematical modeling: An important tool for mathematics teaching. *International Journal of Research and Analytical Reviews (IJRAR)*, 6(2):252–256, 2019. ISSN ISSN-2349-5138.

Bou-Harb, E., Debbabi, M., and Assi, C. A Novel Cyber Security Capability: Inferring Internet-scale infections by Correlating Malware and Probing Activities. *Computer Networks*, 94:327–343, 2016. ISSN 1389-1286. doi:10.1016/j.comnet.2015.11.004.

Bou-Harb, E., Fachkha, C., Debbabi, M., and Assi, C. Inferring Internet-scale Infections by Correlating Malware and Probing Activities. In *2014 IEEE International Conference on Communications (ICC)*, pages 640–646. 2014. doi:10.1109/ICC.2014.6883391.

Bou-Harb, E., Ghani, N., Erradi, A., and Shaban, K. Passive inference of attacks on cps communication protocols. *Journal of Information Security and Applications*, 43:110–122, 2018. ISSN 2214-2126. doi:10.1016/j.jisa.2018.10.002.

Bourke, P. Cross Correlation. Technical report, University of Western Australia, 1996. Date Accessed: 24 May 2020.
http://paulbourke.net/miscellaneous/correlate/

Bronars, S. G. The Power of Nonparametric Tests of Preference Maximization. *Econometrica: Journal of the Econometric Society*, 55(3):693–698, 1987. doi:10.2307/1913608.

Bush, R. RFC 6346: The Address plus Port (A+P) Approach to the IPv4 Address Shortage. Technical Report 6346, IETF, August 2011. doi:10.17487/RFC6346.

Bush, R. R. and Mosteller, F. A Mathematical Model for Simple Learning. *Psychological Review*, 58(5):313, 1951.

CAIDA. CAIDA's Annual Report for 2017. Technical report, Center for Applied Internet Data Analysis (CAIDA), 2017. Date Accessed: 24 October 2021.
http://www.caida.org/home/about/annualreports/2017/

Callan, J. Distributed Information Retrieval. In *Advances in Information Retrieval*, pages 127–150. Springer, 2002. ISBN 978-0-306-47019-6. doi:10.1007/0-306-47019-5_5.

Candès, E. J. and Wakin, M. B. An Introduction to Compressive Sampling - A Sensing/Sampling Paradigm that Goes Against the Common Knowledge in Data Acquisition. *IEEE Signal Processing Magazine*, 25(2):21–30, 2008. ISSN 1558-0792. doi:10.1109/MSP.2007.914731.

Casado-Vara, R., Martin del Rey, A., Pérez-Palau, D., de-la Fuente-Valentín, L., and Corchado, J. M. Web Traffic Time Series Forecasting Using LSTM Neu-

ral Networks with Distributed Asynchronous Training. *Mathematics*, 9(4):421, 2021. doi:10.3390/math9040421.

Chamandy, N., Muralidharan, O., and Wager, S. Teaching Statistics at Google scale. *The American Statistician*, 69(4):283–291, 2015. doi:10.1080/00031305.2015.1089790.

Chan, Y. Biostatistics 102: Quantitative Data - Parametric & Non-parametric Tests. *Singapore Medical Journal*, 140(24.08):391–396, 2003. ISSN 0037-5675.

Chatterjee, S. and Hadi, A. S. Regression Analysis by Example. John Wiley & Sons, 2015. ISBN 978-1-118-45624-8.

Chatziadam, P., Askoxylakis, I. G., and Fragkiadakis, A. A Network Telescope for Early Warning Intrusion Detection. In *International Conference on Human Aspects of Information Security, Privacy, and Trust*, pages 11–22. Springer, 2014. ISBN 978-3-319-07620-1.

Chen, D., Hu, F., Nian, G., and Yang, T. Deep Residual Learning for Nonlinear Regression. *Entropy*, 22(2):193, 2020. doi:10.3390/e22020193.

Chen, Q. and Bridges, R. A. Automated behavioral analysis of malware: A case study of wannacry ransomware. In *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 454–460. 2017. doi:10.1109/ICMLA.2017.0-119.

Chindipha, S. D. and Irwin, B. An Analysis on the Re-emergence of SQL Slammer Worm Using Network Telescope Data. In *The New Digital Economy - How to transform the Telco Networks*. Southern Africa Telecommunication Networks and Applications Conference (SATNAC), 2017.

Chindipha, S. D. and Irwin, B. Feasibility Study: Computing Confidence Interval for IBR Data Using Bootstrapping Technique. In *Accelerated Digitisation - Current and Future Ways of Working*. Southern Africa Telecommunication Networks and Applications Conference (SATNAC), 2021.

Chindipha, S. D., Irwin, B., and Herbert, A. Effectiveness of Sampling a Small Sized Network Telescope in Internet Background Radiation Data Collection. In *The Data Tsunami: Enabled Through Software Defined Transformation*. Southern Africa Telecommunication Networks and Applications Conference (SATNAC), 2018.

Chindipha, S. D., Irwin, B., and Herbert, A. An Evaluation of Text Mining Techniques in Sampling of Network Ports from IBR Traffic. In *The Changing Face of Telcos in a Digital World*. Southern Africa Telecommunication Networks and Applications Conference (SATNAC), 2019a.

Chindipha, S. D., Irwin, B., and Herbert, A. Quantifying the Accuracy of Small Subnet-Equivalent Sampling of IPv4 Internet Background Radiation Datasets. In *Proceedings of the South African Institute of Computer Scientists and Information Technologists 2019*, SAICSIT '19, pages 1–8. Association for Computing Machinery, 2019b. doi:10.1145/3351108.3351129.

Chum, O., Philbin, J., and Zisserman, A. Near Duplicate Image Detection: Min-Hash and TF-IDF Weighting. In *British Machine Vision Conference*, volume 810, pages 812–815. 2008. doi:10.5244/C.22.50.

Cogley, T. and Nason, J. M. Effects of the Hodrick-Prescott Filter on Trend and Difference Stationary Time Series Implications for Business Cycle Research. *Journal of Economic Dynamics and Control*, 19(1-2):253–278, 1995. doi:10.1016/0165-1889(93)00781-X.

Contreras, J., Espinola, R., Nogales, F. J., and Conejo, A. J. ARIMA Models to Predict Next-day Electricity Prices. *IEEE Transactions on Power Systems*, 18(3):1014–1020, 2003. doi:10.1109/TPWRS.2002.804943.

Cooke, E., Bailey, M., Mao, Z. M., Watson, D., Jahanian, F., and McPherson, D. Toward Understanding Distributed Blackhole Placement. In *Proceedings of the 2004 ACM Workshop on Rapid Malcode*, WORM '04, pages 54–64. ACM, New York, NY, USA, 2004. ISBN 1-58113-970-5. doi:10.1145/1029618.1029627.

Cortez, P., Rio, M., Rocha, M., and Sousa, P. Internet Traffic Forecasting Using Neural Networks. In *The 2006 IEEE International Joint Conference on Neural Network Proceedings*, pages 2635–2642. IEEE, 2006. doi:10.1109/IJCNN.2006.247142.

Cortez, P., Rio, M., Rocha, M., and Sousa, P. Multi-scale Internet Traffic Forecasting Using Neural Networks and Time Series Methods. *Expert Systems*, 29(2):143–155, 2012. doi:10.1111/j.1468-0394.2010.00568.x.

Cotton, B. B. Prospecting or Cybersquatting: Registering Your Name Before Someone Else Does. *John Marshall Law Review*, 35:287, 2001. https://repository.jmls.edu/lawreview/vol35/iss2/6

Cotton, M. and Vegoda, L. RFC 5735: Special Use IPv4 Addresses. Technical Report 5735, IETF, January 2010. doi:10.17487/RFC5735.

Czyz, J., Lady, K., Miller, S. G., Bailey, M., Kallitsis, M., and Karir, M. Understanding IPv6 Internet Background Radiation. In *Proceedings of the 2013 Conference on Internet Measurement Conference*, pages 105–118. 2013. ISBN 9781450319539. doi:10.1145/2504730.2504732.

Dai, S. and Finkelman, R. B. Coal geology in china: An overview. *International Geology Review*, 60(5-6):531–534, 2018. doi:10.1080/00206814.2017.1405287.

Dainotti, A., Benson, K., King, A., claffy, k., Kallitsis, M., Glatz, E., and Dimitropoulos, X. Estimating Internet Address Space Usage through Passive Measurements. *SIGCOMM Comput. Commun. Rev.*, 44(1), 2014. ISSN 0146-4833. doi:10.1145/2567561.2567568.

Dainotti, A., Benson, K., King, A., Huffaker, B., Glatz, E., Dimitropoulos, X., Richter, P., Finamore, A., and Snoeren, A. C. Lost in Space: Improving Inference of IPv4 Address Space Utilization. *IEEE Journal on Selected Areas in Communications*, 34(6):1862–1876, 2016. doi:10.1109/JSAC.2016.2559218.

Dang, V. T., Huong, T. T., Thanh, N. H., Nam, P. N., Thanh, N. N., and Marshall, A. SDN-Based SYN Proxy—A Solution to Enhance Performance of Attack Mitigation Under TCP SYN Flood. *The Computer Journal*, 62(4):518–534, 11 2018. ISSN 0010-4620. doi:10.1093/comjnl/bxy117.

Das, K. R. and Imon, A. A Brief Review of Tests for Normality. *American Journal of Theoretical and Applied Statistics*, 5(1):5–12, 2016. doi:10.11648/j.ajtas.20160501.12.

Davison, A. and Kuonen, D. An Introduction to the Bootstrap with Applications in R. *Statistical Computing & Statistical Graphics Newsletter*, 13(1):6–11, 2002. doi:10.1.1.15.5807.

Davison, A. C. and Hinkley, D. V. Bootstrap Methods and their Application. Cambridge University Press, 1997. ISBN 9780521574716.

de Myttenaere, A., Golden, B., Le Grand, B., and Rossi, F. Mean Absolute Percentage Error for Regression Models. *Neurocomputing*, 192:38–48, 2016. ISSN 0925-2312. doi:10.1016/j.neucom.2015.12.114. Advances in Artificial Neural Networks, Machine Learning and Computational Intelligence.

DiCiccio, T. J. and Efron, B. Bootstrap Confidence Intervals. *Statistical Science*, 11(3):189–228, 1996. doi:10.1214/ss/1032280214.

Dixon, P. M. Bootstrap Resampling. In *Encyclopedia of Environmetrics*. Wiley Online Library, 2006. ISBN 9780470057339. doi:10.1002/9780470057339.vab028.

Durand, A., Droms, R., Lee, Y., and Woodyatt, J. RFC 6333: Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion. Technical Report 6333, IETF, August 2011. doi:10.17487/RFC6333.

Efron, B. Estimating the Error Rate of a Prediction Rule: Improvement on Cross-Validation. *Journal of the American Statistical association*, 78(382):316–331, 1983.

Efron, B. Bootstrap Methods: Another Look at the Jackknife. In **Kotz, S. and Johnson, N. L.**, editors, *Breakthroughs in Statistics: Methodology and Distribution*, pages 569–593. Springer, New York, NY, 1992. ISBN 978-1-4612-4380-9. doi:10.1007/978-1-4612-4380-9_41.

Efron, B. and Hastie, T. Computer Age Statistical Inference: Algorithms, Evidence, and Data Science. Cambridge University Press, 2016. ISBN 1107149894.

Efron, B. and Tibshirani, R. Statistical Data Analysis in the Computer Age. *Science*, 253(5018):390–395, 1991. doi:10.1126/science.253.5018.390.

Fachkha, C., Bou-Harb, E., Boukhtouta, A., Dinh, S., Iqbal, F., and Debbabi, M. Investigating the Dark Cyberspace: Profiling, Threat-Based Analysis and Correlation. In *2012 7th International Conference on Risks and Security of Internet and Systems (CRiSIS)*, pages 1–8. IEEE, Washington DC, USA, Oct 2012. ISSN 2151-4763. doi:10.1109/CRISIS.2012.6378947.

Fachkha, C., Bou-Harb, E., Keliris, A., Memon, N. D., and Ahamad, M. Internet-scale Probing of CPS: Inference, Characterization and Orchestration Analysis. In *24th Annual Network and Distributed System Security Symposium, NDSS 2017, San Diego, California, USA, February 26 - March 1, 2017*. 2017. https://www.ndss-symposium.org/ndss2017/

Fachkha, C. and Debbabi, M. Darknet as a Source of Cyber Intelligence: Survey, Taxonomy, and Characterization. *IEEE Communications Surveys Tutorials*, 18(2):1197–1227, 2016. doi:10.1109/COMST.2015.2497690.

Field, A. P. and Wilcox, R. R. Robust Statistical Methods: A Primer for Clinical Psychology and Experimental Psychopathology Researchers. *Behaviour Research and Therapy*, 98:19–38, 2017. doi:10.1016/j.brat.2017.05.013.

Fisher, N. I. and Kordupleski, R. E. Good and Bad Market Research: A Critical Review of Net Promoter Score. *Applied Stochastic Models in Business and Industry*, 35(1):138–151, 2019. doi:10.1002/asmb.2417.

Fneish, F. Calculating Confidence Interval in R. Technical report, R-bloggers, Apr 2021. Date Accessed: 23 August 2021. https://www.r-bloggers.com/2021/04/calculating-confidence-interval-in-r/

Franses, P. H. A Note on the Mean Absolute Scaled Error. *International Journal of Forecasting*, 32(1):20–22, 2016. doi:10.1016/j.ijforecast.2015.03.008.

Friederich, M. C. and van Leeuwen, T. A review of the history of coal exploration, discovery and production in indonesia: The interplay of legal framework, coal geology and exploration strategy. *International Journal of Coal Geology*, 178:56–73, 2017. doi:10.1016/j.coal.2017.04.007.

Friedman, J., Liu, P., Troeger, C. E., Carter, A., Reiner, R. C., Barber, R. M., Collins, J., Lim, S. S., Pigott, D. M., Vos, T. *et al.* Predictive Performance of International COVID-19 Mortality Forecasting Models. *Nature communications*, 12(1):1–13, 2021. doi:10.1038/s41467-021-22457-w.

Granger, C. W. and Joyeux, R. An Introduction to Long-Memory Time Series Models and Fractional Differencing. *Journal of Time Series Analysis*, 1(1):15–29, 1980. doi:10.1111/j.1467-9892.1980.tb00297.x.

Guillot, A., Fontugne, R., Winter, P., Merindol, P., King, A., Dainotti, A., and Pelsser, C. Chocolatine: Outage Detection for Internet Background Radiation. In *2019 Network Traffic Measurement and Analysis Conference (TMA)*, pages 1–8. IEEE, 2019. doi:10.23919/TMA.2019.8784607.

Hamarsheh, A., Abdalaziz, Y., Nashwan, S. *et al.* Recent Impediments in Deploying IPv6. *Advances in Science Technology and Engineering Systems*, 2021. http://repository.aaup.edu/jspui/handle/123456789/1348

Harder, U., Johnson, M. W., Bradley, J. T., and Knottenbelt, W. J. Observing Internet Worm and Virus Attacks with a Small Network Telescope. *Electronic Notes in Theoretical Computer Science*, 151(3):47–59, 2006. doi:10.1016/j.entcs.2006.03.011.

Haukoos, J. S. and Lewis, R. J. Advanced statistics: Bootstrapping confidence intervals for statistics with "difficult" distributions. *Academic Emergency Medicine*, 12(4):360–365, 2005. doi:10.1197/j.aem.2004.11.018.

Held, L., Meyer, S., and Bracher, J. Probabilistic Forecasting in Infectious Disease Epidemiology: the 13th Armitage Lecture. *Statistics in Medicine*, 36(22):3443–3460, 2017. doi:10.1002/sim.7363.

Hesterberg, T. C. What Teachers Should Know About the Bootstrap: Resampling in the Undergraduate Statistics Curriculum. *The American Statistician*, 69(4):371–386, 2015. doi:10.1080/00031305.2015.1089789.

Hewage, P., Behera, A., Trovati, M., Pereira, E., Ghahremani, M., Palmieri, F., and Liu, Y. Temporal Convolutional Neural (TCN) Network for an Effective Weather Forecasting Using Time-Series Data from the Local Weather Station. *Soft Computing*, 24(21):16453–16482, 2020. doi:10.1007/s00500-020-04954-0.

Hiemstra, D. A Probabilistic Justification for Using TF Times IDF Term Weighting in Information Retrieval. *International Journal on Digital Libraries*, 3(2):131–139, 2000. doi:10.1007/s007999900025.

Higgins, J. J. An Introduction to Modern Nonparametric Statistics. Brooks/Cole Pacific Grove, CA, 2004. ISBN 9780534387754.

Hoffmann, J. P. and Shafer, K. Linear Regression Analysis. Washington, DC: NASW Press, 2015. ISBN 978-0871014573.

Horvatic, D., Stanley, H. E., and Podobnik, B. Detrended Cross-Correlation Analysis for Non-Stationary Time Series with Periodic Trends. *Europhysics Letters (EPL)*, 94(1):18007, 2011. doi:10.1209/0295-5075/94/18007.

Houmz, A., Mezzour, G., Zkik, K., Ghogho, M., and Benbrahim, H. Detecting the Impact of Software Vulnerability on Attacks: A Case Study of Network Telescope Scans. *Journal of Network and Computer Applications*, 195:103230, 2021. doi:10.1016/j.jnca.2021.103230.

Howell, D. C. Statistical Methods for Psychology. Cengage Learning, 2012. ISBN 1111835489.

Huang, N. E., Shen, Z., Long, S. R., Wu, M. C., Shih, H. H., Zheng, Q., Yen, N.-C., Tung, C. C., and Liu, H. H. The Empirical Mode Decomposition and the

Hilbert Spectrum for Non-linear and Non-stationary Time Series Analysis. In *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, volume 454, pages 903–995. The Royal Society, 1998. doi:10.1098/rspa.1998.0193.

Hunter, S. O. A framework for Malicious Host Fingerprinting Using Distributed Network Sensors. Master's thesis, Faculty of Science, Computer Science, Rhodes University, 2018.
http://hdl.handle.net/10962/60653

Hunter, S. O., Irwin, B., and Stalmans, E. Real-Time Distributed Malicious Traffic Monitoring for Honeypots and Network Telescopes. In *2013 Information Security for South Africa*, pages 1–9. IEEE, Washington DC, USA, Aug 2013. ISSN 2330-9881. doi:10.1109/ISSA.2013.6641050.

Hyndman, R. J. and Koehler, A. B. Another Look at Measures of Forecast Accuracy. *International Journal of Forecasting*, 22(4):679 – 688, 2006. ISSN 0169-2070.

Irwin, B. A Network Telescope Perspective of the Conficker Outbreak. In *2012 Information Security for South Africa*, pages 1–8. IEEE, Washington DC, USA, Aug 2012. ISSN 2330-9881. doi:10.1109/ISSA.2012.6320455.

Irwin, B. A Baseline Study of Potentially Malicious Activity Across Five Network Telescopes. In *2013 5th International Conference on Cyber Conflict (CYCON 2013)*, pages 1–17. NATO CCDCOE, Tallinn, Estonia, June 2013. ISSN 2325-5374.

Irwin, B. A Source Analysis of the Conficker Outbreak from a Network Telescope. *SAIEE Africa Research Journal*, 104(2):38, 2013. doi:10.23919/SAIEE.2013.8531865.

Irwin, B. and Nkhumeleni, T. M. Observed Correlations of Unsolicited Network Traffic Over Five Distinct IPv4 Netblocks. In *Proceedings of the10th International Conference on Cyber Warfare and Security (ICCWS 2015)*, pages 135–43. IGI Global, Hershey, PA 17033, USA, 2015. ISBN 978-1-910309-96-4.

Irwin, B. V. W. A Framework for the Application of Network Telescope Sensors in a Global IP Network. Ph.D. thesis, Rhodes University, 2011.
http://hdl.handle.net/10962/d1004835

Ismay, C. and Kim, A. Y. Statistical Inference via Data Science: A ModernDive into R and the Tidyverse. CRC Press, 2019. ISBN 0367409828.

Kaiser, G. Mathematical modelling and applications in education. *Encyclopedia of Mathematics Education*, pages 553–561, 2020. doi:10.1007/978-3-030-15789-0_101.

Kalekar, P. S. Time Series Forecasting Using Holt-winters Exponential Smoothing. *Kanwal Rekhi School of Information Technology*, pages 1–13, 2004.

Kapoor, B. G. and Khanna, B. Ichthyology Handbook. Springer Science & Business Media, 2004. ISBN 978-3540428541.

Karevan, Z. and Suykens, J. A. Transductive LSTM for Time-series Prediction: An Application to Weather Forecasting. *Neural Networks*, 125:1–9, 2020. doi:10.1016/j.neunet.2019.12.030.

Kass, R. E. Nonlinear Regression Analysis and its Applications. *Journal of the American Statistical Association*, 85(410):594–596, 1990.

Kim, S. and Kim, H. A New Metric of Absolute Percentage Error for Intermittent Demand Forecasts. *International Journal of Forecasting*, 32(3):669–679, 2016. ISSN 0169-2070.

Kirby, K. N. and Gerlanc, D. BootES: An R Package for Bootstrap Confidence Intervals on Effect Sizes. *Behavior Research Methods*, 45(4):905–927, 2013. doi:10.3758/s13428-013-0330-5.

Kohn, A. F. Autocorrelation and Cross-Correlation Methods. In *Wiley Encyclopedia of Biomedical Engineering*, pages 260–283. Wiley Online Library, 2006. ISBN 0-471-24967-X.

Kreiss, J.-P. and Lahiri, S. N. Bootstrap Methods for Time Series. In *Handbook of Statistics*, volume 30, pages 3–26. Elsevier, 2012. doi:10.1016/B978-0-444-53858-1.00001-6.

Kumar, A., Shankar, R., and Aljohani, N. R. A Big Data Driven Framework for Demand-driven Forecasting with Effects of Marketing-mix Variables. *Industrial Marketing Management*, 90:493–507, 2020. doi:10.1016/j.indmarman.2019.05.003.

Laitinen, M. A. Net Promoter Score as Indicator of Library Customers' Perception. *Journal of Library Administration*, 58(4):394–406, 2018. doi:10.1080/01930826.2018.1448655.

Lencse, G. and Kadobayashi, Y. Comprehensive survey of ipv6 transition technologies: A subjective classification for security analysis. *IEICE Transactions on Communications*, 102(10):2021–2035, 2019. doi:10.1587/transcom.2018EBR0002.

Levin, S. L. and Schmidt, S. IPv4 to IPv6: Challenges, Solutions, and Lessons. *Telecommunications Policy*, 38(11):1059–1068, 2014. doi:10.1016/j.telpol.2014.06.008.

Lewis, D. D. Naive (Bayes) at Forty: The Independence Assumption in Information Retrieval. In *European Conference on Machine Learning*, pages 4–15. Springer, 1998. doi:10.1007/BFb0026666.

Livadariu, I., Elmokashfi, A., and Dhamdhere, A. Measuring IPv6 Adoption in Africa. In *International Conference on e-Infrastructure and e-Services for Developing Countries*, pages 345–351. Springer, 2017. doi:10.1007/978-3-319-98827-6_32.

Maglaras, L. A., Kim, K.-H., Janicke, H., Ferrag, M. A., Rallis, S., Fragkou, P., Maglaras, A., and Cruz, T. J. Cyber security of critical infrastructures. *ICT Express*, 4(1):42–45, 2018. ISSN 2405-9595. doi:10.1016/j.icte.2018.02.001.

Mamushiane, L., Shozi, T., and MANQELE, L. IPv6 Adoption in South Africa: Barriers, Benefits and Government Intervention. In *2021 IST-Africa Conference (IST-Africa)*, pages 1–10. 2021. ISBN 978-1-905824-67-0.

Marcaccioli, R. and Livan, G. Maximum Entropy Approach to Multivariate Time Series Randomization. *Scientific Reports*, 10:10656, 2020. doi:10.1038/s41598-020-67536-y.

Martin, M. A. On Bootstrap Iteration for Coverage Correction in Confidence Intervals. *Journal of the American Statistical Association*, 85(412):1105–1118, 1990. doi:10.1080/01621459.1990.10474982.

Matalas, N. C. Time Series Analysis. *Water Resources Research*, 3(3):817–829, 1967. doi:10.1029/WR003i003p00817.

McCombes, S. An Introduction to Sampling Methods. Technical report, Scribbr, Sep 2019. Date Accessed: 09 December 2021. https://www.scribbr.com/methodology/sampling-methods/

McElhinney, D. and Curran, K. The rise of ransomware aided by vulnerable iot devices. In *Security and Organization within IoT and Smart Cities*, pages 221–242. CRC Press, 2020. ISBN 9781003018636.

McKenzie, J. Mean Absolute Percentage Error and Bias in Economic Forecasting. *Economics Letters*, 113(3):259–262, 2011. doi:10.1016/j.econlet.2011.08.010.

Milhoj, A. Practical Time Series Analysis Using SAS. SAS Institute, 2013. ISBN 9781612906249.

Montgomery, D. C., Peck, E. A., and Vining, G. G. Introduction to Linear Regression Analysis. John Wiley & Sons, 2012. ISBN 978-1-119-57875-8.

Moore, D. Network Telescopes: Observing Small or Distant Security Events. In *Proceedings of the 11th USENIX security symposium*, pages 167–174. 2002.

Moore, D., Shannon, C., Voelker, G. M., and Savage, S. Network telescopes: Technical report. Technical report, University of California, San Diego, 2004. Date Accessed: 20 August 2021.
https://www.caida.org/catalog/papers/2004_tr_2004_04/tr-2004-04.pdf

Motulsky, H. J. and Brown, R. E. Detecting Outliers When Fitting Data with Nonlinear Regression - A New Method Based on Robust Nonlinear Regression and the False Discovery Rate. *BioMed Central Bioinformatics*, 7(1):1–20, 2006. doi:10.1186/1471-2105-7-123.

Nagai, R., Kurihara, W., Higuchi, S., and Hirotsu, T. Design and Implementation of an OpenFlow-Based TCP SYN Flood Mitigation. In *2018 6th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud)*, pages 37–42. 2018. doi:10.1109/MobileCloud.2018.00014.

Nkhumeleni, T. M. Correlation and Comparative Analysis of Traffic Across Five Network Telescopes. Master's thesis, Rhodes University, 2014.
http://hdl.handle.net/10962/d1011668

Nunes, L. J. R., Carlos, J., and Paulo, J. Chapter 1 - introduction. In *Torrefaction of Biomass for Energy Applications*, pages 1–43. Academic Press, 2018. ISBN 978-0-12-809462-4. doi:10.1016/B978-0-12-809462-4.00001-8.

Nyirenda-Jere, T. and Biru, T. Internet Development and Internet Governance in Africa. Technical report, Internet Society, 2015. Date accessed: 18 March 2021.
https://www.internetsociety.org/resources/doc/2015/internet-development-and-internet-governan

Ochieng, N., Mwangi, W., and Ateya, I. Optimizing computer worm detection using ensembles. *Security and Communication Networks*, 2019, 2019. doi:10.1155/2019/4656480.

O'Neill, E. T., McClain, P. D., and Lavoie, B. F. A Methodology for Sampling the World Wide Web. *Journal of Library Administration*, 34(3-4):279–291, 2001. doi:10.1300/J111v34n03_07.

Ostashchuk, O. Time Series Data Prediction and Analysis. Master's thesis, Czech Technical University in Prague Faculty of Electrical Engineering Department of Computer Science, May 2017. Date Accessed: 22 August 2021.
http://hdl.handle.net/10467/70524

Ostertagová, E. Modelling Using Polynomial Regression. *Procedia Engineering*, 48:500–506, 2012. ISSN 1877-7058. doi:10.1016/j.proeng.2012.09.545.

Pang, R., Yegneswaran, V., Barford, P., Paxson, V., and Peterson, L. Characteristics of internet background radiation. In *Proceedings of the 4th ACM SIGCOMM Conference on Internet Measurement*, IMC '04, pages 27–40. ACM, New York, NY, USA, 2004. ISBN 1-58113-821-0. doi:10.1145/1028788.1028794.

Parthasarathy, M. E. O. S. A dissimilarity measure for comparing subsets of data: Application to multivariate time series. In *Proceedings of a Workshop held in Conjunction with 2005 IEEE International Conference on Data Mining*, pages 101–112. IBM Research, Florida International University, USA, 2005. ISBN 0-9738918-3-1.

Pearson, D. T. An Exploration of the Overlap Between Open Source Threat Intelligence and Active Internet Background Radiation. Master's thesis, Faculty of Science, Computer Science, 2020.
http://hdl.handle.net/10962/103802

Pemberton, D., Komisarczuk, P., and Welch, I. Internet Background Radiation Arrival Density and Network Telescope Sampling Strategies. In *2007 Australasian Telecommunication Networks and Applications Conference*, pages 246–252. IEEE, Washington DC, USA, Dec 2007. doi:10.1109/ATNAC.2007.4665254.

Perkins, C. RFC 5944: IP Mobility Support for IPv4, Revised. Technical Report 5944, IETF, November 2010. doi:10.17487/RFC5944.

Perkins, C. E., Arkko, J., and Johnson, D. B. RFC 3775: Mobility Support in IPv6. Technical Report 3775, IETF, June 2004. doi:10.17487/RFC3775.

Piotr, B., Adrian, K., and Pawel, P. Network Telescopes Revisited: From Loads of Unwanted Traffic to Threat Intelligence. Technical report, Research and Academic

Computer Network (NASK, Poland), 2019. Date Accessed: 06 July 2021. https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=539961

Polakis, I., Kontaxis, G., Ioannidis, S., and Markatos, E. P. Dynamic Monitoring of Dark IP Address Space (Poster). In *International Workshop on Traffic Monitoring and Analysis*, pages 193–196. Springer, 2011. doi:10.1007/978-3-642-20305-3_20.

Rawlings, J. O., Pantula, S. G., and Dickey, D. A. Applied Regression Analysis: A Research Tool. Springer Science & Business Media, 2001. ISBN 978-0-387-22753-5.

Rees, P. Demography. In **Kobayashi, A.**, editor, *International Encyclopedia of Human Geography (Second Edition)*, pages 239–256. Elsevier, Oxford, second edition edition, 2020. ISBN 978-0-08-102296-2. doi:10.1016/B978-0-08-102295-5.10252-5.

Reina, D., Toral, S., Johnson, P., and Barrero, F. Improving Discovery Phase of Reactive Ad Hoc Routing Protocols Using Jaccard Distance. *The Journal of Supercomputing*, 67(1):131–152, 2014. doi:10.1007/s11227-013-0992-x.

Richter, P., Allman, M., Bush, R., and Paxson, V. A Primer on IPv4 Scarcity. *ACM SIGCOMM Computer Communication Review*, 45(2):21–31, 2015. doi:10.1145/2766330.2766335.

Richter, P. and Berger, A. Scanning the Scanners: Sensing the Internet from a Massively Distributed Network Telescope. In *Proceedings of the Internet Measurement Conference*, IMC '19, page 144–157. Association for Computing Machinery, New York, NY, USA, 2019. ISBN 9781450369480. doi:10.1145/3355369.3355595.

Ritz, C. and Streibig, J. C. Nonlinear Regression with R. Springer Science & Business Media, 2008. ISBN 978-0-387-09616-2.

Rousselet, G., Pernet, C., and Wilcox, R. R. A Practical Introduction to the Bootstrap: A Versatile Method to Make Inferences by Using Data-driven Simulations. *PsyArXiv*, 2019. doi:10.31234/osf.io/h8ft7.

Rousselet, G. A., Pernet, C. R., and Wilcox, R. R. Beyond Differences in Means: Robust Graphical Methods to Compare Two Groups in Neuroscience. *European Journal of Neuroscience*, 46(2):1738–1748, 2017. doi:10.1111/ejn.13610.

Rousselet, G. A., Pernet, C. R., and Wilcox, R. R. The Percentile Bootstrap: A Teaser with Step-by-Step Instructions in R. *Advances in Methods and Practices in Psychological Science*, 4(1), 2021. doi:10.1177/2515245920911881.

Rüping, S. SVM Kernels for Time Series Analysis. Technical Report 2001,43, Universität Dortmund, Sonderforschungsbereich 475 - Komplexitätsreduktion in Multivariaten Datenstrukturen, 2001.
http://hdl.handle.net/10419/77140

Sanft, R. and Walter, A. Exploring Mathematical Modeling in Biology Through Case Studies and Experimental Activities. Academic Press, 2020. ISBN 9780128195956.

Santhanakumar, M., Columbus, C. C., and Jayapriya, K. Multi Term Based Co-term Frequency Method for Term Weighting in Information Retrieval. *International Journal of Business Information Systems*, 28(1):79–94, 2018. doi:10.1504/IJBIS.2018.091164.

Schütze, H., Manning, C. D., and Raghavan, P. Introduction to Information Retrieval, volume 39. Cambridge University Press Cambridge, 2008. ISBN 0521865719.

Seabold, S. and Perktold, J. Statsmodels: Econometric and Statistical Modeling with Python. In *9th Python in Science Conference*. 2010.
http://conference.scipy.org/proceedings/scipy2010/pdfs/seabold.pdf

Shameem, M.-U.-S. and Ferdous, R. An Efficient K-Means Algorithm Integrated with Jaccard Distance Measure for Document Clustering. In *2009 First Asian Himalayas International Conference on Internet*, pages 1–6. IEEE, 2009. doi:10.1109/AHICI.2009.5340335.

Shannon, C. and Moore, D. The Spread of the Witty Worm. *IEEE Security & Privacy*, 2(4):46–50, 2004. doi:10.1109/MSP.2004.59.

Shelatkar, T., Tondale, S., Yadav, S., and Ahir, S. Web traffic time series forecasting using arima and lstm rnn. In *International Technical Meeting (ITM) Web of Conferences*, volume 32, page 03017. International Conference on Automation, Computing and Communication 2020 (ICACC-2020), 2020. doi:10.1051/itmconf/20203203017.

Simar, L. and Wilson, P. W. Sensitivity Analysis of Efficiency Scores: How to Bootstrap in Non-parametric Frontier Models. *Management Science*, 44(1):49–61, 1998. doi:10.1287/mnsc.44.1.49.

Simmon, E., Kim, K.-S., Subrahmanian, E., Lee, R., De Vaulx, F., Murakami, Y., Zettsu, K., Sriram, R. D. *et al.* A Vision of Cyber-physical Cloud Computing for Smart Networked Systems. US Department of Commerce, National Institute of Standards and Technology (NIST), 2013. doi:10.6028/NIST.IR.7951.

Sinan, A. and Alkan, B. B. A Useful Approach to Identify the Multicollinearity in the Presence of Outliers. *Journal of Applied Statistics*, 42(5):986–993, 2015. doi:10.1080/02664763.2014.993369.

Smadja, F. Retrieving Collocations From Text: Xtract. *Computational Linguistics*, 19(1):143–177, 1993. doi:10.5555/972450.972458.

Smyth, G. K. Nonlinear Regression. In *Encyclopedia of Environmetrics*, volume 4. Wiley Online Library, 2006. doi:10.1002/9780470057339.van017.

Thomas, J. and Galligher, G. Improving Backup System Evaluations in Information Security Risk Assessments to Combat Ransomware. *Computer and Information Science*, 11(1):1–12, 2018. doi:10.5539/cis.v11n1p14.

Tibshirani, R. J. and Efron, B. An Introduction to the Bootstrap. CRC press, 1993. ISBN 978-0412042317.

Torabi, S., Bou-Harb, E., Assi, C., Karbab, E. B., Boukhtouta, A., and Debbabi, M. Inferring and Investigating IoT-Generated Scanning Campaigns Targeting A Large Network Telescope. *IEEE Transactions on Dependable and Secure Computing*, pages 1–18, 2020. doi:10.1109/TDSC.2020.2979183.

Twomey, P. and Kroll, M. How to Use Linear Regression and Correlation in Quantitative Method Comparison Studies. *International Journal of Clinical Practice*, 62(4):529–538, 2008. doi:10.1111/j.1742-1241.2008.01709.x.

Varouchakis, E. A. and Hristopulos, D. T. Improvement of Groundwater Level Prediction in Sparsely Gauged Basins Using Physical Laws and Local Geographic Features as Auxiliary Variables. *Advances in Water Resources*, 52:34–49, 2013. doi:10.1016/j.advwatres.2012.08.002.

Wang, Z. and Bovik, A. C. Mean Squared Error: Love it or Leave it? A New Look at Signal Fidelity Measures. *IEEE Signal Processing Magazine*, 26(1):98–117, Jan 2009. doi:10.1109/MSP.2008.930649.

Wei, W. W. Time Series Analysis. In *The Oxford Handbook of Quantitative Methods in Psychology: Vol. 2*. OXFORD University Press, 2006. doi:10.1093/oxfordhb/9780199934898.001.0001.

Wiener, E., Pedersen, J. O., Weigend, A. S. *et al.* A Neural Network Approach to Topic Spotting. In *Proceedings of SDAIR-95, 4th Annual Symposium on Document*

Analysis and Information Retrieval, volume 317, page 332. Las Vegas, NV, 1995. https://ci.nii.ac.jp/naid/10004655550/en/

Wilcox, R. R. Introduction to Robust Estimation and Hypothesis Testing. Academic Press, 2011. ISBN : 978-0-12-386983-8.

Willmott, C. J. and Matsuura, K. Advantages of the Mean Absolute Error (MAE) Over the Root Mean Square Error (RMSE) in Assessing Average Model Performance. *Climate Research*, 30(1):79–82, 2005. doi:10.3354/cr030079.

Wood, M. Bootstrapped Confidence Intervals as an Approach to Statistical Inference. *Organizational Research Methods*, 8(4):454–470, 2005. doi:10.1177/1094428105280059.

Wright, D. B., London, K., and Field, A. P. Using Bootstrap Estimation and the Plug-in Principle for Clinical Psychology Data. *Journal of Experimental Psychopathology*, 2(2):252–270, 2011. doi:10.5127/jep.013611.

Wustrow, E., Karir, M., Bailey, M., Jahanian, F., and Huston, G. Internet Background Radiation Revisited. In *Proceedings of the 10th ACM SIGCOMM Conference on Internet Measurement*, IMC '10, pages 62–74. ACM, New York, NY, USA, 2010. ISBN 978-1-4503-0483-2. doi:10.1145/1879141.1879149.

Xu, S. Cybersecurity Dynamics: A Foundation for the Science of Cybersecurity. In *Proactive and Dynamic Network Defense*, pages 1–31. Springer, 2019. doi:10.1007/978-3-030-10597-6_1.

Yegneswaran, V., Barford, P., and Plonka, D. On the Design and Use of Internet Sinks for Network Abuse Monitoring. In **Jonsson, E., Valdes, A., and Almgren, M.**, editors, *Recent Advances in Intrusion Detection*, pages 146–165. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004. doi:10.1007/978-3-540-30143-1_8.

Yuan, Y., Chao, M., and Lo, Y.-C. Automatic Skin Lesion Segmentation Using Deep Fully Convolutional Networks with Jaccard Distance. *IEEE Transactions on Medical Imaging*, 36(9):1876–1886, 2017. doi:10.1109/TMI.2017.2695227.

Zander, S., Andrew, L. L. H., Armitage, G., and Huston, G. Estimating IPv4 Address Apace Usage with Capture-Recapture. In *38th Annual IEEE Conference on Local Computer Networks - Workshops*, pages 1010–1017. 2013. doi:10.1109/LCNW.2013.6758545.

Zebende, G. DCCA Cross-Correlation Coefficient: Quantifying Level of Cross-Correlation. *Physica A: Statistical Mechanics and its Applications*, 390(4):614 – 618, 2011. ISSN 0378-4371. doi:10.1016/j.physa.2010.10.022.

Zeghache, L. and Yacine, H. Small Network Telescope for Advanced Security Monitoring. In *ECCWS 2020 20th European Conference on Cyber Warfare and Security*, page 428. Academic Conferences and publishing limited, 2020. ISBN 9781912764624.

Zhan, Z., Xu, M., and Xu, S. A Characterization of Cybersecurity Posture from Network Telescope Data. In *International Conference on Trusted Systems*, pages 105–126. Springer, 2014. doi:10.1007/978-3-319-27998-5_7.

Zhang, C., Zhou, S., and Chain, B. M. Hybrid Epidemics: A Case Study on Computer Worm Conficker. *PLoS One*, 10(5):1–17, 2015. doi:10.1371/journal.pone.0127478.

Zou, C. C., Gao, L., Gong, W., and Towsley, D. Monitoring and Early Warning for Internet Worms. In *Proceedings of the 10th ACM Conference on Computer and Communications Security*, CCS '03, page 190–199. Association for Computing Machinery, New York, NY, USA, 2003. ISBN 1581137389. doi:10.1145/948109.948136.

Zou, C. C., Gong, W., Towsley, D., and Gao, L. The Monitoring and Early Detection of Internet Worms. *IEEE/ACM Transactions on Networking*, 13(5):961–974, 2005. doi:10.1109/TNET.2005.857113.

Zoubir, A. M. and Iskandler, D. R. Bootstrap Methods and Applications. *IEEE Signal Processing Magazine*, 24(4):10–19, 2007. doi:10.1109/MSP.2007.4286560.

A

Top 20 SRCIP Address

This appendix details the top 20 unique SRCIPs that were observed in this study. Note that all the tables presented in this appendix have been presented in **Section 4.2.1**, however, in this appendix, the study has added the actual count of each of the unique SRCIP. as in **Section 4.2.1**, SRCIPs in bold shows that these SRCIPs were present in all three Datasets for the month that they are presented in. The tables in this appendix used data that was collected between January - March 2021. The percentage computed is based in the total amount of traffic for each sensor. Thus for TCP traffic, the total was based on the total TCP traffic and for UDP traffic it used the total UDP traffic for the sensor being evaluated. The total count of packets contributed by each network telescope are also displayed in tables found in **Section 4.2.1**.

Table A.1: Top 20 SRCIP Breakdown Based on Volume of TCP Traffic [Jan 2021]

Rank	146/8	146_count	%	155/8	155_count	%	196-A/8	196_count	%
1	**92.63.197.97**	1,289,048	4.26	**92.63.197.97**	1,288,809	3.64	**92.63.197.97**	1,288,581	3.07
2	**185.175.93.24**	1,103,201	3.65	**185.175.93.24**	1,154,006	3.26	**185.175.93.24**	1,153,787	2.75
3	**79.124.62.74**	720,383	2.38	**79.124.62.74**	720,384	23	**79.124.62.74**	720,384	1.72
4	**194.26.25.125**	687,665	2.27	**194.26.25.125**	687,669	1.94	**194.26.25.125**	687,639	1.64
5	194.147.140.41	390,040	1.29	194.147.140.41	395,079	1.12	**64.95.96.217**	259,399	0.62
6	194.147.140.42	265,500	0.88	178.33.221.97	270,687	0.76	194.147.140.8	252,818	0.60
7	194.147.140.6	252,641	0.84	194.147.140.42	265,595	0.75	**193.27.229.47**	224,935	0.54
8	45.129.33.128	237,290	0.78	194.147.140.6	252,069	0.71	**103.145.13.58**	220,622	0.53
9	**193.27.229.47**	223,317	0.74	45.129.33.128	238,778	0.67	**74.106.249.155**	219,152	0.52
10	**74.106.249.155**	219,976	0.73	205.220.231.26	235,611	0.67	**45.146.164.211**	193,561	0.46
11	**103.145.13.58**	214,890	0.71	**193.27.229.47**	223,714	0.63	**103.195.100.208**	193,465	0.46
12	45.129.33.47	213,379	0.71	**74.106.249.155**	221,535	0.63	**141.98.10.138**	183,639	0.44
13	**103.195.100.208**	193,478	0.64	45.129.33.47	209,556	0.59	194.26.25.13	134,762	0.32
14	**45.146.164.211**	190,755	0.63	**45.146.164.211**	193,549	0.55	89.248.160.178	132,405	0.32
15	**141.98.10.138**	183,410	0.61	**103.195.100.208**	193,510	0.55	45.146.165.171	129,178	0.31
16	122.228.19.79	173,835	0.57	**141.98.10.138**	183,492	0.52	93.174.93.123	123,090	0.29
17	**64.95.96.217**	166,537	0.55	122.228.19.79	173,714	0.49	103.145.13.43	115,955	0.28
18	45.146.165.171	133,920	0.44	**64.95.96.217**	166,761	0.47	205.220.231.26	112,965	0.27
19	89.248.160.178	133,555	0.44	**103.145.13.58**	155,482	0.44	161.189.114.127	111,699	0.27
20	194.26.25.13	129,986	0.43	205.220.231.25	153,812	0.43	38.130.221.107	110,597	0.26

Table A.2: Top 20 SRCIP Breakdown Based on Volume of UDP Traffic [Jan 2021]

Rank	146/8	146_count	%	155/8	155_count	%	196-A/8	196_count	%
1	**146.88.240.4**	287,622	8.01	**146.88.240.4**	287,309	7.63	196.216.37.82	1,021,138	17.74
2	**95.214.52.175**	155,741	4.34	**95.214.52.175**	156,895	4.17	77.247.108.45	445,672	7.74
3	**95.214.53.145**	119,274	3.32	**95.214.53.145**	119,436	3.17	77.247.108.35	429,821	7.47
4	**69.162.117.142**	86,511	2.41	**69.162.117.142**	84,113	2.24	**146.88.240.4**	287,207	4.99
5	**95.214.54.95**	78,373	2.18	**95.214.54.95**	78,645	29	**95.214.52.175**	168,297	2.92
6	**193.29.14.109**	67,999	1.89	**104.243.40.37**	58,112	1.54	**95.214.53.145**	119,263	2.07
7	**104.243.40.37**	58,098	1.62	**185.94.111.1**	53,176	1.41	**95.214.54.95**	82,819	1.44
8	80.94.93.24	54,782	1.53	109.248.203.69	51,310	1.36	**69.162.117.142**	81,977	1.42
9	**185.94.111.1**	53,165	1.48	**95.214.54.161**	47,942	1.27	80.94.93.24	60,416	1.05
10	**95.214.54.161**	48,395	1.35	**45.125.65.52**	45,926	1.22	**104.243.40.37**	58,110	1.01
11	**45.125.65.52**	45,821	1.28	**193.29.14.109**	41,305	1.10	**185.94.111.1**	53,176	0.92
12	**80.94.93.16**	30,800	0.86	**80.94.93.16**	30,595	0.81	**45.125.65.52**	51,557	0.90
13	**80.82.65.90**	30,344	0.85	**80.82.65.90**	30,359	0.81	109.248.203.69	51,426	0.89
14	213.59.4.26	30,000	0.84	**80.94.93.10**	29,966	0.80	**95.214.54.161**	49,292	0.86
15	**80.94.93.10**	29,671	0.83	72.251.228.101	26,112	0.69	23.148.145.30	40,077	0.70
16	83.97.20.25	29,211	0.81	104.152.52.31	25,600	0.68	**193.29.14.109**	39,873	0.69
17	193.29.14.125	28,074	0.78	104.152.52.23	25,600	0.68	**80.94.93.10**	32,577	0.57
18	72.251.228.101	26,368	0.73	122.228.19.79	25,126	0.67	196.192.178.26	31,733	0.55
19	122.228.19.79	26,105	0.73	147.203.255.20	21,666	0.58	**80.82.65.90**	30,370	0.53
20	104.152.52.26	25,600	0.71	83.97.20.25	19,307	0.51	**80.94.93.16**	30,063	0.52

Table A.3: Top 20 SRCIP Breakdown Based on Volume of TCP Traffic [Feb 2021]

Rank	146/8	146_count	%	155/8	155_count	%	196-A/8	196_count	%
1	**89.248.165.101**	5,668,657	17.50	**89.248.165.101**	5,652,894	15.72	**89.248.165.101**	5,650,015	15.71
2	**79.124.62.74**	1,854,415	5.73	**79.124.62.74**	1,855,064	5.16	**79.124.62.74**	1,855,040	5.16
3	**79.124.62.234**	1,338,862	4.13	**79.124.62.234**	1,340,499	3.73	**79.124.62.234**	1,340,525	3.73
4	194.147.140.41	401,659	1.24	194.147.140.41	404,738	1.13	**89.190.156.53**	511,199	1.42
5	**89.190.156.53**	303,974	0.94	**89.190.156.53**	302,676	0.84	45.79.121.175	334,725	0.93
6	194.147.140.42	244,421	0.75	194.147.140.42	248,443	0.69	**74.106.249.155**	181,419	0.50
7	194.147.140.68	214,189	0.66	205.220.231.26	203,076	0.56	**45.146.164.211**	175,815	0.49
8	194.147.140.66	198,991	0.61	178.33.221.97	203,002	0.56	**89.190.156.52**	159,963	0.44
9	194.147.140.70	182,379	0.56	194.147.140.68	198,191	0.55	89.248.160.178	125,636	0.35
10	**74.106.249.155**	178,717	0.55	194.147.140.66	192,859	0.54	94.232.46.244	124,431	0.35
11	194.147.140.40	176,606	0.55	**74.106.249.155**	181,064	0.50	89.248.165.104	113,042	0.31
12	**45.146.164.211**	174,881	0.54	194.147.140.70	176,733	0.49	93.174.93.123	110,019	0.31
13	194.147.140.69	165,107	0.51	194.147.140.40	176,536	0.49	103.145.13.58	104,420	0.29
14	194.147.140.26	161,244	0.50	**45.146.164.211**	175,720	0.49	89.248.165.53	103,412	0.29
15	122.228.19.79	159,826	0.49	194.147.140.69	174,925	0.49	205.220.231.26	101,526	0.28
16	**89.190.156.52**	159,702	0.49	194.147.140.26	161,799	0.45	89.248.165.51	101,079	0.28
17	194.147.140.67	150,879	0.47	194.147.140.96	161,660	0.45	194.61.25.194	98,610	0.27
18	194.147.140.96	143,411	0.44	122.228.19.79	160,018	0.45	103.145.13.43	96,439	0.27
19	89.248.160.178	124,997	0.39	**89.190.156.52**	160,000	0.45	89.248.165.93	93,688	0.26
20	94.232.46.244	123,950	0.38	194.147.140.67	142,746	0.40	45.125.65.105	89,003	0.25

Table A.4: Top 20 SRCIP Breakdown Based on Volume of UDP Traffic [Feb 2021]

Rank	146/8	146_count	%	155/8	155_count	%	196-A/8	196_count	%
1	**146.88.240.4**	263,347	7.38	**146.88.240.4**	263,228	79	196.216.37.82	1,916,962	33.32
2	**77.247.108.175**	255,629	7.16	**77.247.108.175**	253,870	6.84	**146.88.240.4**	263,130	4.57
3	**77.247.108.74**	239,625	6.72	**77.247.108.74**	240,801	6.49	**77.247.108.74**	248,225	4.32
4	**77.247.108.58**	120,845	3.39	**77.247.108.58**	122,097	3.29	**77.247.108.175**	243,376	4.23
5	103.145.13.60	60,156	1.69	213.59.4.26	58,500	1.58	**77.247.108.58**	118,007	2.05
6	**185.94.111.1**	48,629	1.36	**185.94.111.1**	48,667	1.31	103.145.13.60	60,120	1.05
7	**103.145.13.55**	45,308	1.27	**103.145.13.55**	45,305	1.22	**185.94.111.1**	48,677	0.85
8	**193.29.14.109**	43,335	1.21	**156.96.156.138**	42,496	1.14	**103.145.13.55**	45,311	0.79
9	**156.96.156.138**	42,495	1.19	**45.125.65.52**	33,783	0.91	**156.96.156.138**	42,496	0.74
10	103.145.13.59	34,047	0.95	**80.82.65.90**	28,072	0.76	**45.125.65.52**	36,174	0.63
11	**45.125.65.52**	33,823	0.95	**103.145.13.18**	27,643	0.74	38.91.100.237	34,676	0.60
12	**80.82.65.90**	28,570	0.80	**193.29.14.109**	26,729	0.72	103.145.13.59	34,040	0.59
13	**103.145.13.18**	27,644	0.77	**72.251.228.101**	26,110	0.70	217.182.199.129	32,362	0.56
14	**72.251.228.101**	26,112	0.73	104.152.52.32	25,600	0.69	193.46.255.20	29,952	0.52
15	104.152.52.28	25,600	0.72	104.152.52.24	25,599	0.69	95.214.53.145	29,806	0.52
16	104.152.52.18	25,600	0.72	122.228.19.79	24,627	0.66	**80.82.65.90**	28,574	0.50
17	122.228.19.79	25,104	0.70	193.29.14.112	23,432	0.63	**103.145.13.18**	27,635	0.48
18	193.29.14.127	23,565	0.66	193.107.216.17	23,065	0.62	**72.251.228.101**	25,853	0.45
19	193.29.14.112	23,435	0.66	89.40.70.237	22,586	0.61	**193.29.14.109**	25,614	0.45
20	89.40.70.237	22,589	0.63	217.182.199.129	22,515	0.61	104.152.52.34	25,600	0.45

Table A.5: Top 20 SRCIP Breakdown Based on Volume of TCP Traffic [Mar 2021]

Rank	146/8	146_count	%	155/8	155_count	%	196-A/8	196_count	%
1	194.147.140.122	1,302,516	3.87	194.147.140.122	1,302,603	3.46	**45.93.201.188**	1,137,997	3.39
2	194.147.140.126	1,297,916	3.86	194.147.140.126	1,299,723	3.45	**82.102.137.130**	748,049	2.23
3	**45.93.201.188**	1,143,559	3.40	**45.93.201.188**	1,139,971	33	**193.27.229.207**	368,295	1.10
4	**82.102.137.130**	762,880	2.27	**82.102.137.130**	764,487	23	**193.27.229.47**	346,319	1.03
5	194.147.140.41	469,816	1.40	194.147.140.41	471,380	1.25	**89.190.156.52**	278,016	0.83
6	**193.27.229.207**	317,639	0.94	**193.27.229.207**	366,137	0.97	**69.25.114.212**	272,886	0.81
7	**193.27.229.47**	303,293	0.90	**193.27.229.47**	347,506	0.92	**45.155.205.155**	233,120	0.69
8	**89.190.156.52**	280,221	0.83	**89.190.156.52**	280,064	0.74	**89.190.156.53**	204,800	0.61
9	**69.25.114.212**	272,994	0.81	**69.25.114.212**	273,095	0.73	72.251.228.103	194,984	0.58
10	194.147.140.42	222,687	0.66	**45.155.205.155**	232,819	0.62	**89.248.165.101**	194,247	0.58
11	194.147.140.26	213,325	0.63	194.147.140.42	223,010	0.59	**45.146.164.211**	191,935	0.57
12	**45.155.205.155**	204,586	0.61	194.147.140.26	212,690	0.57	89.248.165.203	189,977	0.57
13	**89.248.165.101**	203,308	0.60	**89.248.165.101**	196,924	0.52	**45.146.165.24**	177,905	0.53
14	**89.190.156.53**	175,821	0.52	**45.146.164.211**	191,929	0.51	**103.99.2.190**	173,859	0.52
15	**103.99.2.190**	174,535	0.52	**103.99.2.190**	176,573	0.47	185.188.182.105	153,588	0.46
16	**45.146.165.24**	172,099	0.51	**45.146.165.24**	176,431	0.47	94.232.46.244	147,106	0.44
17	**45.146.164.211**	171,197	0.51	**89.190.156.53**	176,128	0.47	41.57.124.37	144,648	0.43
18	185.156.73.67	161,825	0.48	194.147.140.29	167,958	0.45	45.146.164.170	129,416	0.39
19	194.147.140.29	158,771	0.47	185.156.73.67	163,000	0.43	45.125.65.105	129,380	0.39
20	194.147.140.40	158,654	0.47	194.147.140.40	158,658	0.42	89.248.165.104	127,727	0.38

Table A.6: Top 20 SRCIP Breakdown Based on Volume of UDP Traffic [Mar 2021]

Rank	146/8	146_count	%	155/8	155_count	%	196-A/8	196_count	%
1	**146.88.240.4**	295,019	7.90	**103.145.13.75**	324,931	6.40	107.148.161.86	4,291,755	27.15
2	103.145.13.131	126,697	3.39	**103.145.13.74**	299,979	5.90	196.216.37.82	2,347,917	14.85
3	**103.145.13.74**	94,934	2.54	**146.88.240.4**	294,991	5.81	103.248.20.30	1,112,395	7.04
4	**193.46.255.40**	92,160	2.47	103.145.13.131	126,680	2.49	23.27.103.158	882,595	5.58
5	**103.145.13.75**	79,983	2.14	**193.46.255.40**	97,660	1.92	103.248.20.21	428,508	2.71
6	**103.145.13.167**	75,768	2.03	**193.107.216.17**	74,321	1.46	23.27.103.157	419,184	2.65
7	**89.40.70.51**	68,790	1.84	**103.145.13.167**	73,307	1.44	77.247.108.45	378,103	2.39
8	**193.107.216.17**	68,267	1.83	**89.40.70.51**	68,789	1.35	**103.145.13.75**	338,377	2.14
9	**103.145.13.78**	66,042	1.77	103.145.13.147	67,327	1.33	77.247.108.35	323,305	2.05
10	**185.94.111.1**	50,310	1.35	45.143.221.110	66,300	1.30	**103.145.13.74**	314,255	1.99
11	193.29.14.125	43,008	1.15	**185.94.111.1**	50,380	0.99	**146.88.240.4**	294,725	1.86
12	**89.190.156.53**	41,688	1.12	193.29.14.125	43,263	0.85	45.121.107.128	223,170	1.41
13	103.145.13.69	40,190	1.08	**89.190.156.53**	40,960	0.81	103.145.13.130	126,676	0.80
14	92.204.135.183	34,816	0.93	92.204.135.183	34,816	0.69	**193.46.255.40**	97,659	0.62
15	209.222.98.168	30,966	0.83	**103.145.13.78**	31,742	0.62	45.143.221.110	87,520	0.55
16	81.177.143.31	30,000	0.80	209.222.98.168	30,966	0.61	**193.107.216.17**	79,927	0.51
17	89.248.165.164	28,871	0.77	89.248.165.164	28,546	0.56	**103.145.13.167**	75,714	0.48
18	103.145.13.77	28,672	0.77	72.251.228.101	26,112	0.51	**89.40.70.51**	68,896	0.44
19	193.46.254.182	26,368	0.71	104.152.52.30	25,600	0.50	103.145.13.147	67,277	0.43
20	72.251.228.101	26,112	0.70	104.152.52.26	25,599	0.50	**103.145.13.78**	66,045	0.42

B

Top 20 DPORT address

This appendix details the top 20 DPORTs that were observed in this study. Note that all the tables presented in this appendix have been presented in **Section 4.2.2**, however, in this appendix, the study has added the actual count of each of the unique DPORT. as in **Section 4.2.2**, DPORTs in bold shows that these DPORTs were present in all three Datasets for the month that they are presented in. The tables in this appendix used data that was collected between January - March 2021. The percentage computed is based in the total amount of traffic for each sensor. Thus for TCP traffic, the total was based on the total TCP traffic and for UDP traffic it used the total UDP traffic for the sensor being evaluated. The total count of packets contributed by each network telescope are also displayed in tables found in **Section 4.2.2**

Table B.1: Top 20 DPORT Breakdown Based on Volume of TCP Traffic [Jan 2021]

Rank	146_dport	146_count	%	155_dport	155_count	%	196_dport	196_count	%
1	**23**	2,103,448	6.95	**445**	3,547,173	10.01	37215	8,010,179	19.09
2	**22**	796,501	2.63	**23**	2,140,939	6.04	**445**	4,037,807	9.62
3	**80**	633,181	2.09	**1433**	908,346	2.56	**23**	2,171,084	5.17
4	**445**	539,204	1.78	**22**	728,336	2.06	**22**	1,031,770	2.46
5	**443**	399,802	1.32	**80**	609,937	1.72	**1433**	998,307	2.38
6	**8080**	375,804	1.24	**8080**	378,385	1.07	**80**	609,896	1.45
7	**3389**	344,989	1.14	**443**	367,386	1.04	**8080**	470,206	1.12
8	**81**	318,837	1.05	**3389**	352,536	1.00	**443**	384,988	0.92
9	**6379**	239,438	0.79	**81**	323,190	0.91	**3389**	339,200	0.81
10	**5555**	209,123	0.69	**6379**	263,730	0.74	**81**	319,810	0.76
11	**5038**	161,858	0.53	10530	211,076	0.60	**6379**	241,491	0.58
12	**8545**	151,148	0.50	33529	210,868	0.60	**5555**	204,819	0.49
13	**1433**	142,819	0.47	12111	210,775	0.60	34694	198,565	0.47
14	50802	126,700	0.42	61380	210,682	0.59	**5038**	166,840	0.40
15	**8081**	120,998	0.40	**5555**	206,421	0.58	**8545**	150,837	0.36
16	8443	110,419	0.36	16979	199,625	0.56	50802	129,653	0.31
17	**11211**	109,500	0.36	**8545**	151,147	0.43	8728	125,954	0.30
18	2323	106,157	0.35	**5038**	140,816	0.40	**8081**	123,881	0.30
19	3306	105,181	0.35	**8081**	123,004	0.35	8443	114,720	0.27
20	139	103,111	0.34	**11211**	110,200	0.31	**11211**	109,414	0.26

Table B.2: Top 20 DPORT Breakdown Based on Volume of UDP Traffic [Jan 2021]

Rank	146_dport	146_count	%	155_dport	155_count	%	196_dport	196_count	%
1	**5060**	515110	14.35	**5060**	473251	12.58	**53**	1,155,912	20.08
2	**123**	217,903	6.07	**123**	208,960	5.55	**5060**	597,160	10.37
3	**53**	136,016	3.79	**53**	137,180	3.65	**123**	273,423	4.75
4	**1900**	116,912	3.26	**1900**	118,501	3.15	**161**	133,982	2.33
5	**161**	103,117	2.87	**161**	103,130	2.74	**1900**	116,837	2.03
6	**389**	77,299	2.15	**389**	76,867	2.04	**389**	90,367	1.57
7	**1434**	58,155	1.62	**11211**	62,037	1.65	**1434**	58,181	1.01
8	**11211**	53,967	1.50	**1434**	57,953	1.54	**11211**	53,644	0.93
9	**5353**	49,921	1.39	**5353**	50,248	1.34	**137**	51,009	0.89
10	**137**	48,121	1.34	**137**	47,873	1.27	**5353**	49,645	0.86
11	**5683**	43,720	1.22	**5683**	43,895	1.17	**5683**	44,299	0.77
12	**111**	39,892	1.11	**111**	39,622	1.05	**111**	40,522	0.70
13	**1194**	38,549	1.07	**1194**	38,901	1.03	**1194**	38,577	0.67
14	**6881**	36,866	1.03	**6881**	36,504	0.97	**6881**	36,799	0.64
15	**3283**	31,539	0.88	33434	34,709	0.92	19	34,615	0.60
16	19	31,417	0.88	**3283**	32,575	0.87	**3283**	31,544	0.55
17	6536	30,000	0.84	33435	32,382	0.86	5070	31,032	0.54
18	5070	29,536	0.82	33441	31,338	0.83	5632	28,061	0.49
19	5632	28,355	0.79	19	31,127	0.83	5351	27,366	0.48
20	5351	27,667	0.77	33440	30,823	0.82	1027	25,523	0.44

Table B.3: Top 20 DPORT Breakdown Based on Volume of TCP Traffic [Feb 2021]

Rank	146_dport	146_count	%	155_dport	155_count	%	196_dport	196_count	%
1	**23**	1,820,860	5.62	**445**	3,057,231	8.50	**445**	3,495,767	9.72
2	**22**	590,156	1.82	**23**	1,852,139	5.15	**23**	1,814,881	5.05
3	**80**	507,086	1.57	**1433**	779,613	2.17	37215	1,685,544	4.69
4	**445**	432,801	1.34	**22**	547,601	1.52	**1433**	812,691	2.26
5	**8080**	317,590	0.98	**80**	501,459	1.39	**22**	763,259	2.12
6	**3389**	312,189	0.96	**8080**	322,299	0.90	**80**	506,570	1.41
7	**5555**	310,462	0.96	**3389**	312,504	0.87	**8080**	401,767	1.12
8	**443**	292,483	0.90	**443**	290,860	0.81	**3389**	310,219	0.86
9	**6379**	210,472	0.65	**6379**	240,808	0.67	**443**	292,249	0.81
10	**81**	193,098	0.60	**5555**	233,314	0.65	**5555**	250,890	0.70
11	5038	151,179	0.47	**81**	193,893	0.54	**6379**	214,772	0.60
12	**8081**	124,847	0.39	**8081**	124,585	0.35	**81**	195,131	0.54
13	**1433**	114,567	0.35	**8888**	109,340	0.30	5038	154,125	0.43
14	**3306**	107,009	0.33	**3306**	106,648	0.30	8291	136,011	0.38
15	**8888**	106,170	0.33	**11211**	103,204	0.29	8728	129,896	0.36
16	**11211**	102,804	0.32	12111	102,325	0.28	**8081**	129,721	0.36
17	26	97,586	0.30	61380	101,766	0.28	**8888**	107,545	0.30
18	2323	92,595	0.29	10530	101,757	0.28	**3306**	105,943	0.29
19	8443	91,648	0.28	16979	101,755	0.28	**11211**	103,726	0.29
20	50802	87,537	0.27	33529	101,540	0.28	34694	101,538	0.28

Table B.4: Top 20 DPORT Breakdown Based on Volume of UDP Traffic [Feb 2021]

Rank	146_dport	146_count	%	155_dport	155_count	%	196_dport	196_count	%
1	**5060**	529,974	14.85	**5060**	486,822	13.12	**53**	2,050,077	35.64
2	**123**	266,334	7.46	**123**	244,452	6.59	**5060**	622,502	10.82
3	**53**	132,695	3.72	**53**	132,761	3.58	**123**	263,103	4.57
4	**1900**	112,406	3.15	**1900**	112,048	3.02	**1900**	113,604	1.97
5	**389**	96,894	2.72	**389**	97,018	2.61	**161**	104,983	1.83
6	**161**	87,724	2.46	**161**	87,694	2.36	**389**	89,719	1.56
7	**1434**	44,578	1.25	**11211**	46,786	1.26	**137**	47,715	0.83
8	**137**	42,799	1.20	**1434**	44,413	1.20	**11211**	47,231	0.82
9	**11211**	41,000	1.15	**137**	42,672	1.15	**1434**	45,143	0.78
10	**5683**	40,329	1.13	54047	40,408	1.09	**5683**	40,049	0.70
11	**5353**	38,542	1.08	**5683**	40,311	1.09	**5353**	38,388	0.67
12	**5070**	36,187	1.01	**5353**	39,131	1.05	**3283**	37,676	0.65
13	69	35,732	1.00	**5070**	35,575	0.96	**5070**	37,388	0.65
14	**111**	35,288	0.99	69	35,455	0.96	69	35,224	0.61
15	3702	34,264	0.96	**111**	35,315	0.95	**111**	34,875	0.61
16	**1194**	33,753	0.95	**1194**	33,625	0.91	**1194**	33,474	0.58
17	**19**	28,094	0.79	6576	29,999	0.81	**19**	28,200	0.49
18	**3283**	27,821	0.78	**19**	29,108	0.78	6881	26,992	0.47
19	6881	27,564	0.77	**3283**	28,833	0.78	3702	26,380	0.46
20	5632	25,824	0.72	6532	28,503	0.77	5632	25,120	0.44

Table B.5: Top 20 DPORT Breakdown Based on Volume of TCP Traffic [Mar 2021]

Rank	146_dport	146_count	%	155_dport	155_count	%	196_dport	196_count	%
1	**23**	2,385,546	7.10	**445**	3,897,416	10.35	**445**	4,454,415	13.27
2	**22**	745,548	2.22	**23**	2,380,361	6.32	**23**	2,313,902	6.89
3	**80**	595,523	1.77	**1433**	966,434	2.57	**1433**	989,313	2.95
4	**445**	534,708	1.59	**22**	702,522	1.87	**22**	904,012	2.69
5	**6379**	483,091	1.44	**6379**	601,606	1.60	37215	696,852	2.08
6	**5555**	469,066	1.40	**80**	586,196	1.56	**80**	601,699	1.79
7	**443**	400,252	1.19	**443**	394,699	1.05	**6379**	504,017	1.50
8	**3389**	367,592	1.09	**3389**	367,050	0.98	**8080**	425,178	1.27
9	**8080**	343,352	1.02	**8080**	340,652	0.90	**443**	405,559	1.21
10	**81**	254,384	0.76	**5555**	281,651	0.75	**3389**	353,370	1.05
11	**26**	164,865	0.49	**81**	251,768	0.67	**5555**	271,174	0.81
12	**1433**	141,313	0.42	**26**	162,756	0.43	**81**	251,579	0.75
13	**8291**	140,492	0.42	**8081**	133,197	0.35	**8291**	212,772	0.63
14	**8081**	132,507	0.39	**8443**	122,669	0.33	8728	175,859	0.52
15	**8443**	124,734	0.37	**8291**	122,295	0.32	**26**	162,095	0.48
16	5038	120,929	0.36	5900	118,650	0.32	2375	139,709	0.42
17	5900	116,078	0.35	2323	115,354	0.31	**8081**	133,585	0.40
18	2323	114,257	0.34	8545	114,217	0.30	9090	124,670	0.37
19	8545	113,921	0.34	8000	106,338	0.28	**8443**	121,927	0.36
20	8000	103,650	0.31	9999	97,969	0.26	5038	121,446	0.36

Table B.6: Top 20 DPORT Breakdown Based on Volume of UDP Traffic [Mar 2021]

Rank	146_dport	146_count	%	155_dport	155_count	%	196_dport	196_count	%
1	**5060**	626,574	16.77	**5060**	594,975	11.71	**123**	8,253,368	52.21
2	**123**	327,728	8.77	**123**	293,839	5.78	**53**	2,503,148	15.83
3	**53**	153,350	4.11	**53**	154,426	3.04	**5060**	804,339	5.09
4	**389**	139,354	3.73	**389**	142,216	2.80	**389**	145,954	0.92
5	**1900**	99,837	2.67	**1900**	106,217	2.09	**161**	144,321	0.91
6	**161**	91,394	2.45	**161**	92,053	1.81	**1900**	99,558	0.63
7	3702	52,340	1.40	49693	55,479	1.09	3283	94,775	0.60
8	**1434**	49,791	1.33	**11211**	49,713	0.98	**137**	51,816	0.33
9	**137**	47,094	1.26	**1434**	48,701	0.96	**11211**	50,955	0.32
10	**5683**	45,240	1.21	**137**	46,384	0.91	**1434**	49,460	0.31
11	**5353**	43,986	1.18	**5683**	45,734	0.90	3702	47,523	0.30
12	**11211**	40,485	1.08	25631	44,495	0.88	**5683**	45,393	0.29
13	**1194**	39,513	1.06	**5353**	44,240	0.87	**5353**	44,023	0.28
14	111	31,089	0.83	11551	42,696	0.84	**1194**	39,861	0.25
15	6881	30,695	0.82	44060	39,858	0.78	5070	31,756	0.20
16	6572	30,000	0.80	44830	39,850	0.78	111	30,818	0.19
17	19	29,255	0.78	**1194**	39,816	0.78	6881	30,444	0.19
18	5070	28,529	0.76	9757	39,594	0.78	19	29,257	0.19
19	3283	28,140	0.75	28447	39,440	0.78	5080	25,750	0.16
20	17	24,907	0.67	62495	39,022	0.77	5632	25,104	0.16

C

Ports and Services

This appendix contains two tables, one for TCP ports ans another for UDP ports. The services were categorised based on these two protocols. The top 20 ports that have been includes in each table represent ports that appeared the most in all the datasets under study. The tables containing these ports and their proportion in each network telescope are presented in **Section 4.2.2**. More explanation is found in that section with the focus primarily on those that displayed interesting pattern or unexpected ranking in their respective tables. In this appendix, all services are listed for all top performing ports. All the details regarding the port services were taken from Speed Guide[1] and IANA[2].

[1]https://www.speedguide.net
[2]https://www.iana.org/assignments/service-names-port-numbers

Table C.1: Top 20 TCP DPORTs and Services Run on them

Rank	DPORT	Service
1	23	Telnet
2	22	Secure Shell (SSH)
3	80	Hyper Text Transfer Protocol (HTTP)
4	445	Microsoft Directory Services for Active Directory (AD) and for the Server Message Block (SMB)
5	8080	Common alternative HTTP
6	3389	Microsoft Remote Desktop Protocol (RDP)
7	5555	Default for Microsoft Dynamics CRM
8	443	Hypertext Transfer Protocol Secure (HTTPS)
9	6379	Remote Dictionary Server (Redis)
10	81	Hyper Text Transfer Protocol (HTTP)
11	5038	Asterisk Manager Interface (AMI)
12	8081	Hyper Text Transfer Protocol (HTTP)
13	1433	Microsoft SQL Server
14	3306	MySQL database server connections
15	8888	NewsEDGE server
16	11211	Memory cache service
17	26	Used by Secure File Transfer Protocol (SFTP) - a simple FTP-like protocol
18	2323	3d-nfsd
19	8443	PCSync HTTPS (SSL)
20	5900	Virtual Network Computing (VNC)

Table C.2: Top 20 UDP DPORTs and Services Run on them

Rank	DPORT	Service
1	5060	Session Initiation Protocol (SIP)
2	123	Network Time Protocol (NTP)
3	53	Domain Name Service (DNS)
4	389	Lightweight Directory Access Protocol (LDAP)
5	1900	Simple Service Discovery Protocol (SSDP)
6	161	Simple network management protocol (SNMP)
7	3702	Web Services Discovery (WSD)
8	1434	Microsoft SQL Server
9	137	NetBIOS
11	5353	Multicast DNS (MDNS)
12	11211	Memory cache service
13	1194	OpenVPN (Virtual Private Networking)
14	111	SUN Remote Procedure Call
15	3283	Apple Remote Desktop Net Assistant reporting feature
16	6572	Unassigned
17	19	Character Generator
18	5070	VersaTrans Server Agent Service,
19	3283	Apple Remote Desktop Net Assistant reporting feature
20	69	Trivial File Transfer Protocol (TFTP)

D

Project Online Repository

This appendix details electronic resources relating to this research project. The scripts repository contains the analysis scripts used during the data analysis and quantification. Access to the online repository is not public and as such should be requested from the author of this document. To contact the author to request access please send an email to daltiso@gmail.com. The repository can be found in the link found below:

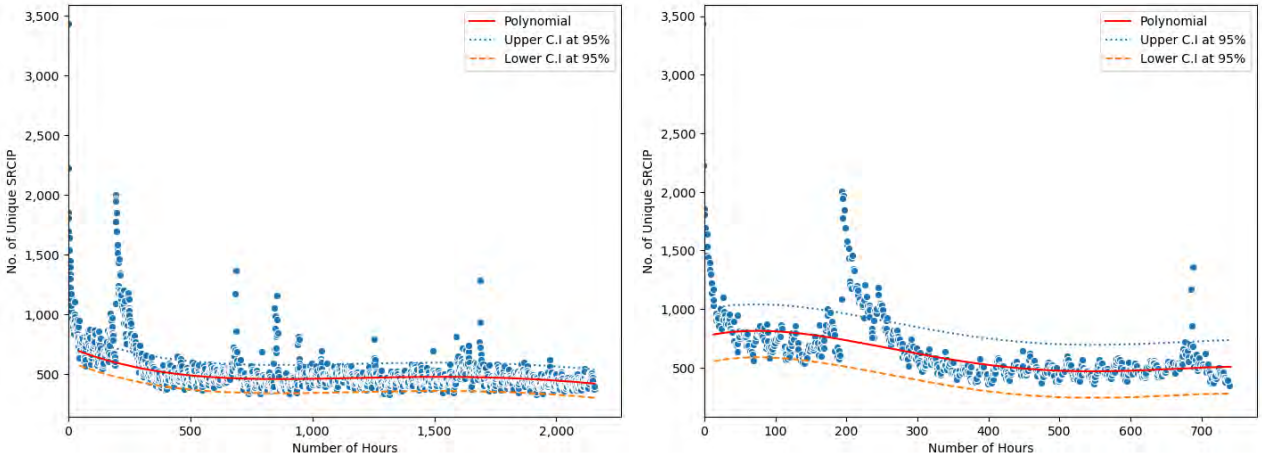
https://bitbucket.org/daltiso/phd-project/src/master/

E

Regression Plots

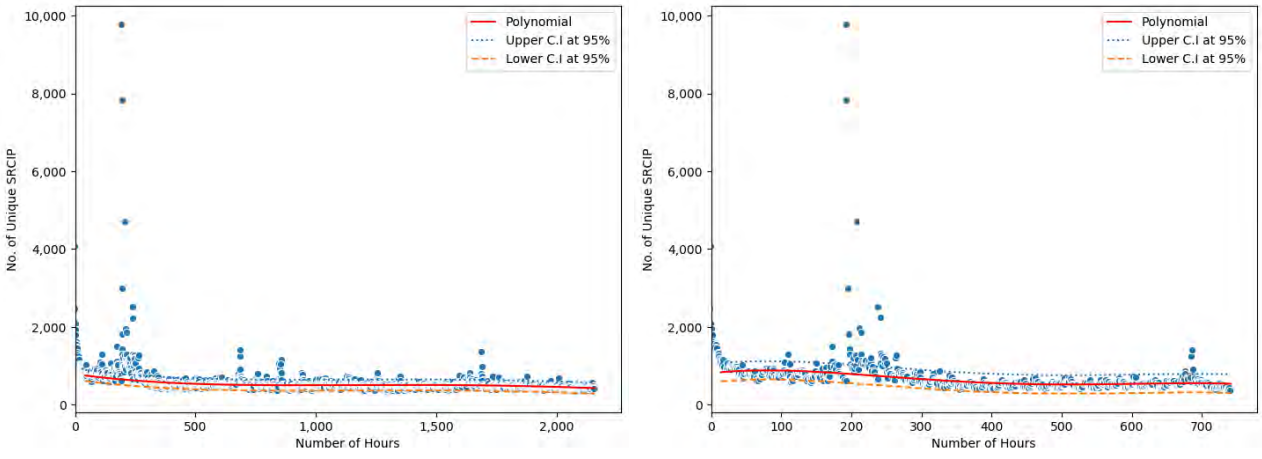
This appendix contain regression analysis plots that show the relationship between the duration of observation and the number of unique SRCIPs observed per hour. Unlike in **Section 5.4**, this appendix includes regression that contained outliers which were not account for in **Section 5.4**. The regression plots with outliers are found in **Section E.1**. The reason for the exclusion of these plots have also been explained in that section (**Section 5.4**). This appendix also contains additional plots that were not included in the same section (**Section 5.4**) to help the reader have a better understanding from all network telescopes. The additional plots without outliers can be found in **Section E.2**.

E.1 Regression Plots With Outliers



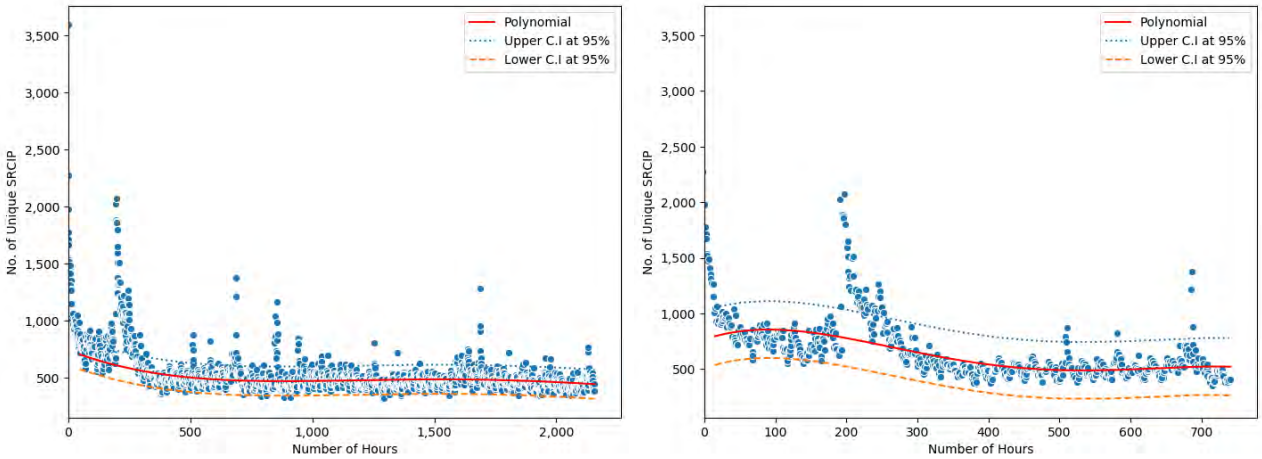
(a) January - March (b) January

Figure E.1: 146/8: Number of Unique SRCIP observed/hour [with outliers]



(a) January - March (b) January

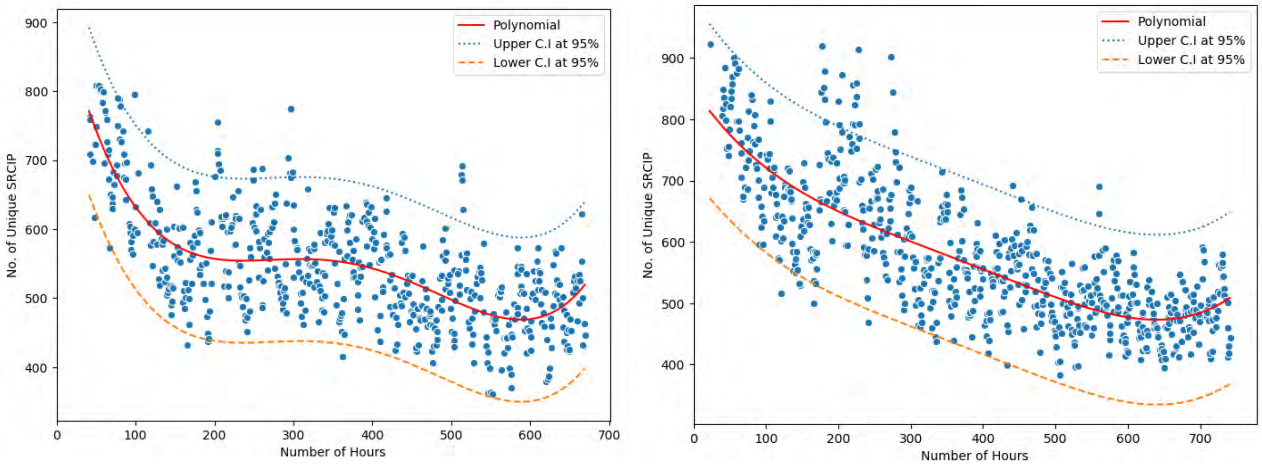
Figure E.2: 196-A/8: Number of Unique SRCIP observed/hour [with outliers]



(a) January - March (b) January

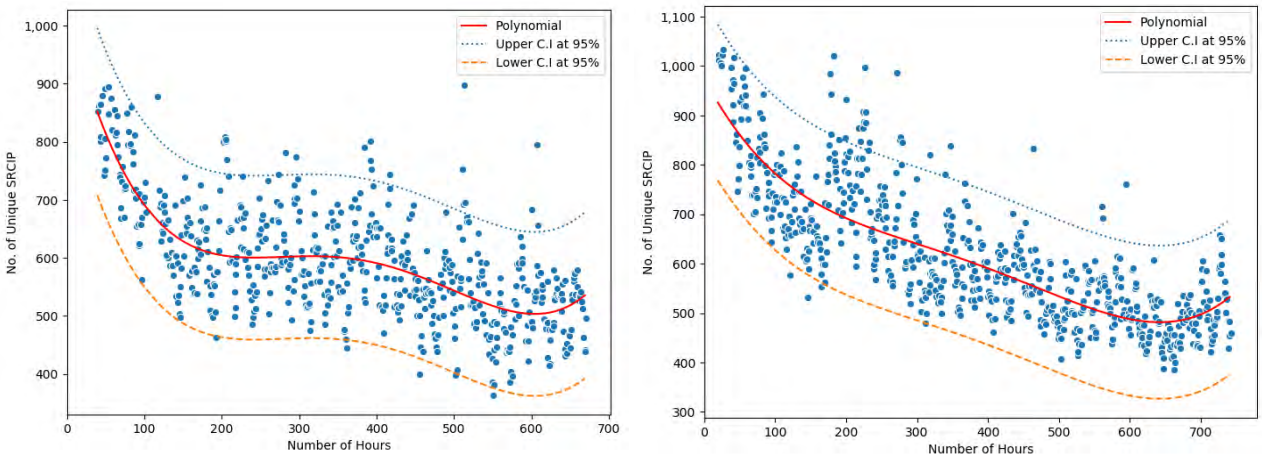
Figure E.3: 155/8: Number of Unique SRCIP observed/hour [with outliers]

E.2 Regression Plots Without Outliers



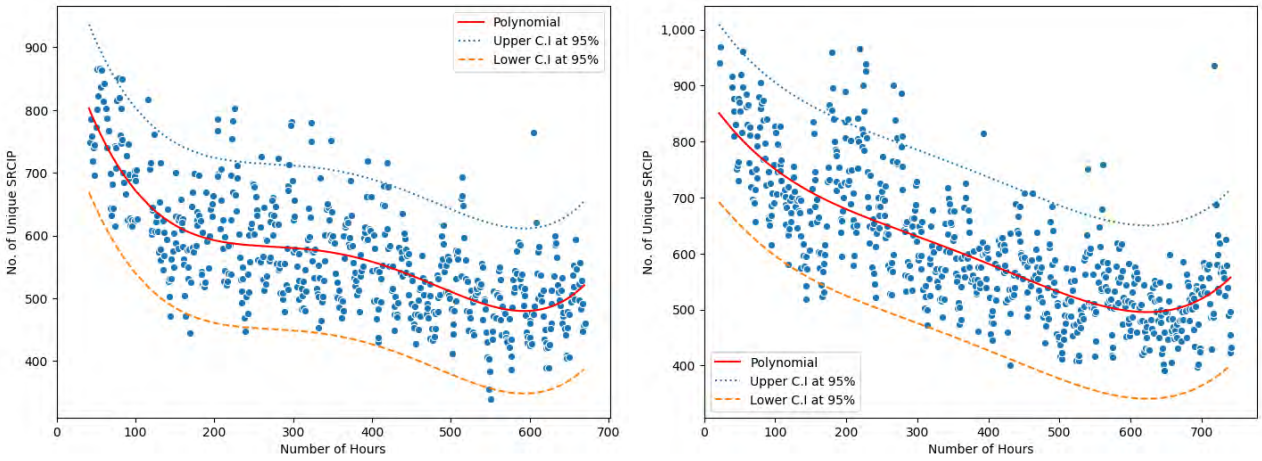
(a) February (b) March

Figure E.4: 146/8 -[Feb - Mar]: Number of Unique SRCIP observed/hour



(a) February (b) March

Figure E.5: 196-A/8 -[Feb - Mar]: Number of Unique SRCIP observed/hour



(a) February (b) March

Figure E.6: 155/8 -[Feb - Mar]: Number of Unique SRCIP observed/hour

F

CI findings for January

This appendix contain additional confidence interval findings that were not included in **Section 5.5**. In **Section 5.5**, the findings that were included were those of **146/8** and **155/8** network telescopes . However in this appendix, the **196-A/8** network telescopes CI findings that were not included in **Section 5.5** can be found. Note that the findings that were not included in the main body were those of the month of January only.

Table F.1: 196-A/8-012021: CI for No. of Unique SRCIP/hour [Non-Parametric]

	CI Level			
Bootstrap sample	80%	90%	95%	99%
196-A/8 - /e24	[793 - 806]	[791 - 807]	[790 - 809]	[787 - 812]
196/8 - /e25	[399 - 406]	[398 - 406]	[397 - 407]	[396 - 409]
196/8 - /e26	[199 - 202]	[199 - 203]	198 - 203]	[198 - 204]
196/8 - /e27	[99 - 101]	[99 - 101]	[98 - 101]	[98 - 102]

Table F.2: 196-A/8-012021: CI for No. of Unique SRCIP/hour [Parametric]

	CI Level			
Bootstrap sample	80%	90%	95%	99%
196/8 - /e24	[931 - 951]	[927 - 954]	[924 - 957]	[920 - 961]
196/8 - /e25	[468 - 479]	[467 - 481]	[466 - 482]	[462 - 486]
196/8 - /e26	[234 - 240]	[233 - 241]	[233 - 241]	[232 - 243]
196/8 - /e27	[116 - 120]	[116 - 120]	[115 - 121]	115 - 121]

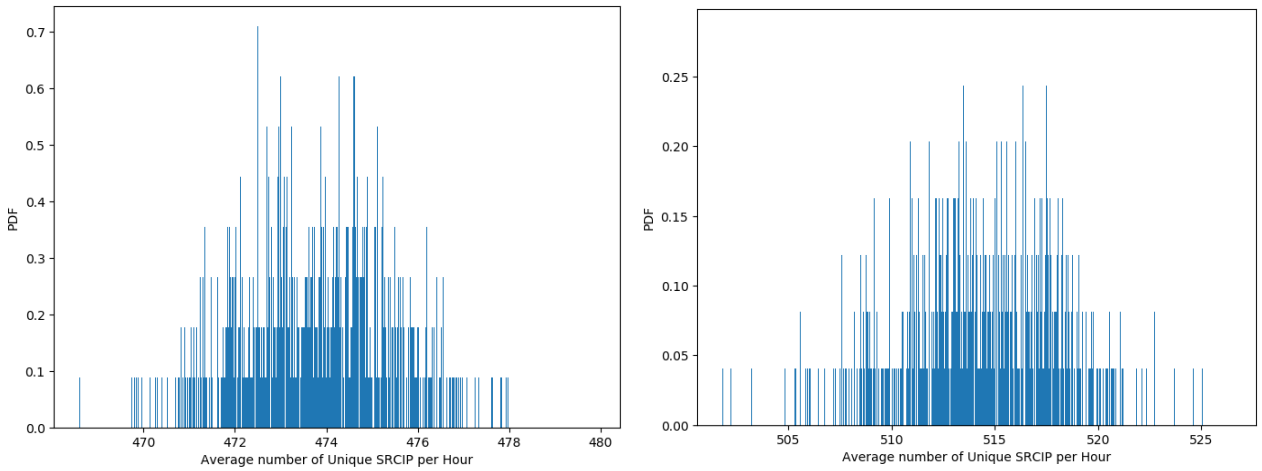
Table F.3: Summary statistics for 196-A/8-012021 [Non-Parametric]

Bootstrap Sample	Baseline Mean	Bootstrap Mean	Baseline SEM	Bootstrap SEM
196/8 - /e24	799	800	4.843	4.84
196/8 - /e25	403	402	2.52	2.52
196/8 - /e26	201	201	1.32	1.32
196/8 - /e27	100	100	0.70	0.70

G

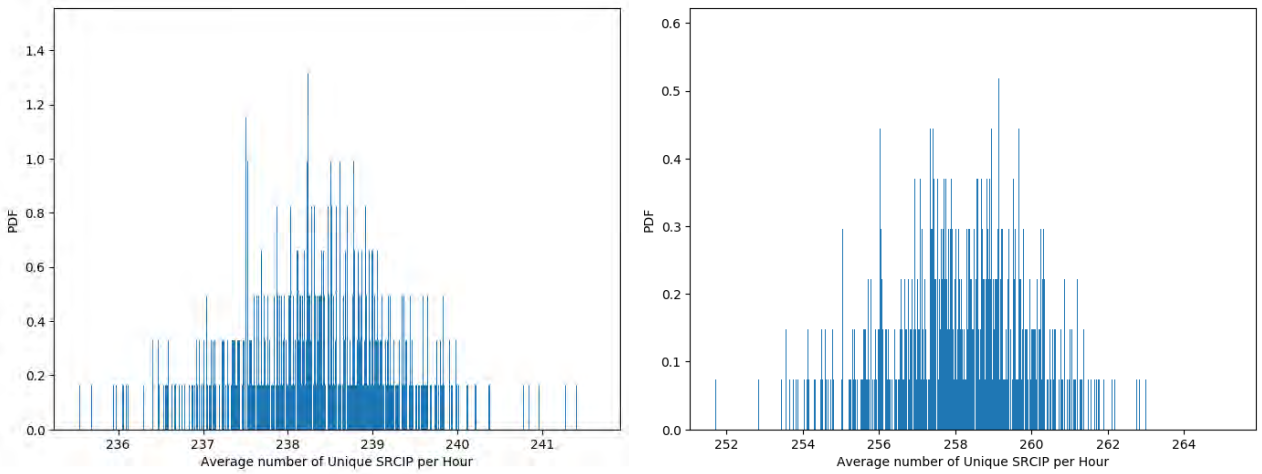
Plots for CI

This appendix contains additional findings of confidence interval plots. These plots are graphical representation of the bootstrap samples. Similar results in the main body can be found in **Section 5.6** where their interpretation is given.



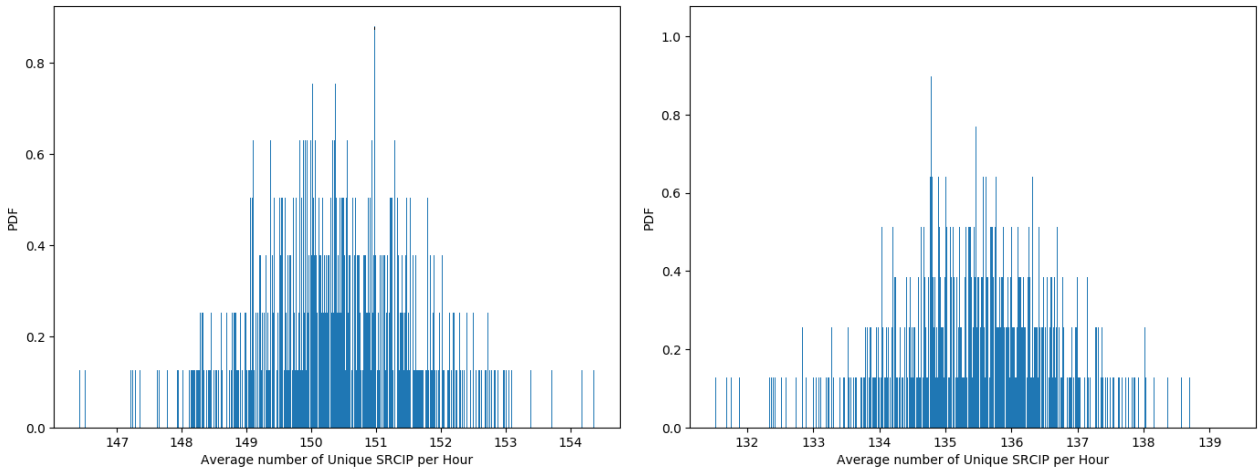
(a) Non - Parametric Bootstrap Sample (b) Parametric Bootstrap Sample

Figure G.1: 146/8-012021: /24 Subnet equivalent Bootstrap Sample at 95% CI



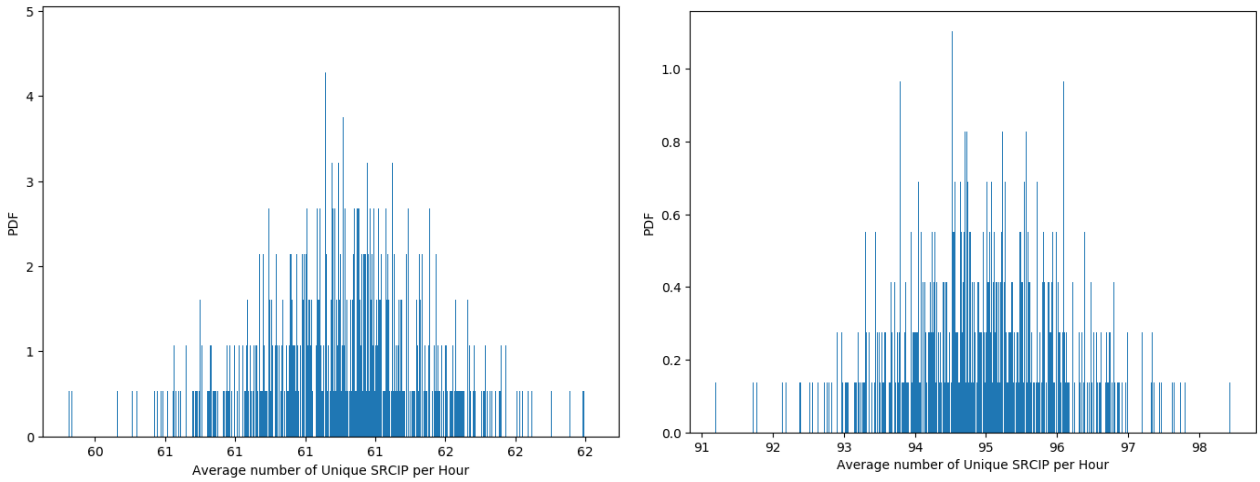
(a) Non - Parametric Bootstrap Sample (b) Parametric Bootstrap Sample

Figure G.2: 146/8-022021: /$_e$25 Subnet equivalent Bootstrap Sample at 95% CI



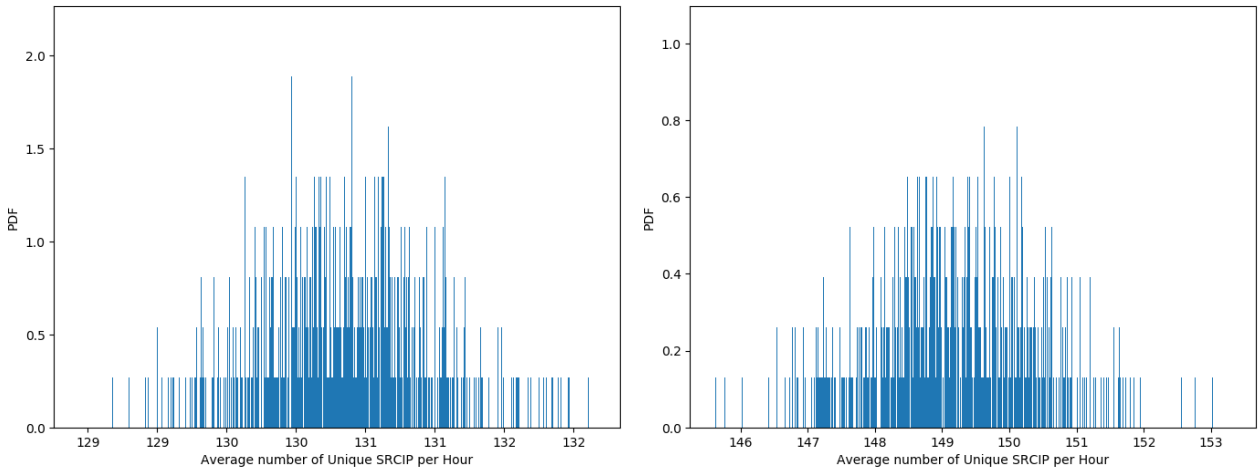
(a) Non - Parametric Bootstrap Sample (b) Parametric Bootstrap Sample

Figure G.3: 155/8-0220211: /$_e$26 Subnet equivalent Bootstrap Sample at 95% CI



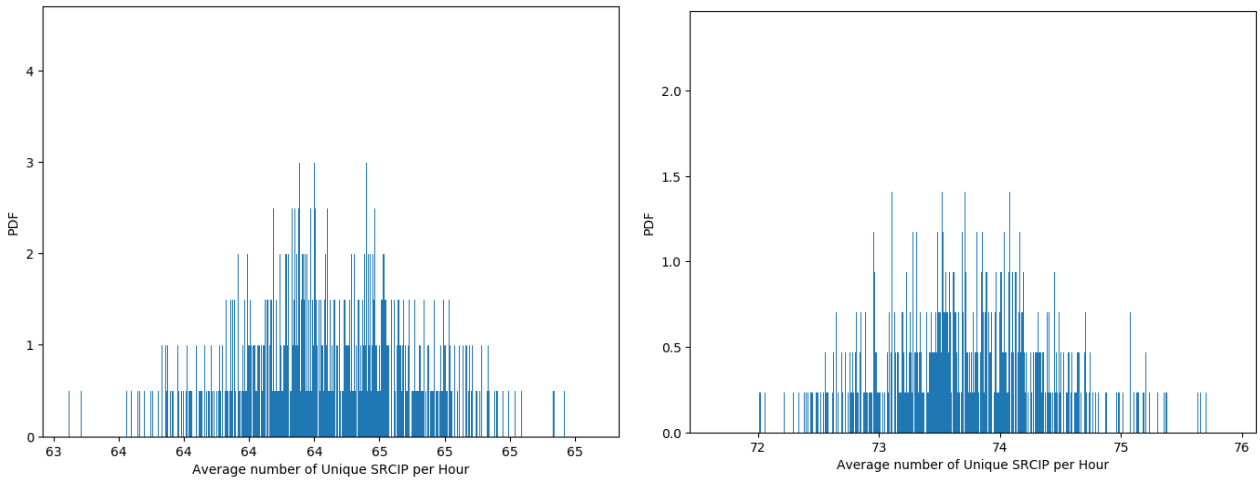
(a) Non - Parametric Bootstrap Sample (b) Parametric Bootstrap Sample

Figure G.4: 155/8-022021: /$_e$27 Subnet equivalent Bootstrap Sample at 95% CI



(a) Non - Parametric Bootstrap Sample (b) Parametric Bootstrap Sample

Figure G.5: 196-A/8-032021: /$_e$26 Subnet equivalent Bootstrap Sample at 95% CI



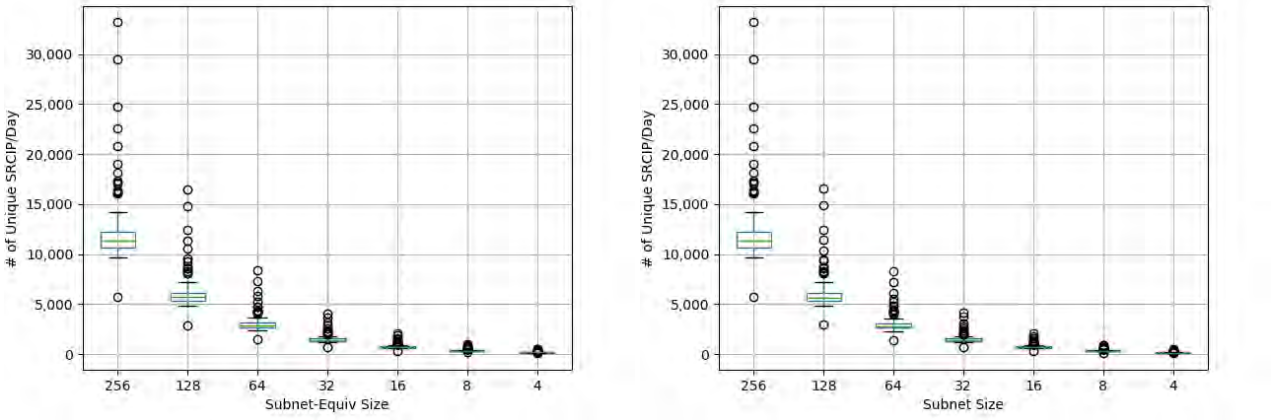
(a) Non - Parametric Bootstrap Sample (b) Parametric Bootstrap Sample

Figure G.6: 196-A/8-022021: /$_e$27 Subnet equivalent Bootstrap Sample at 95% CI

H

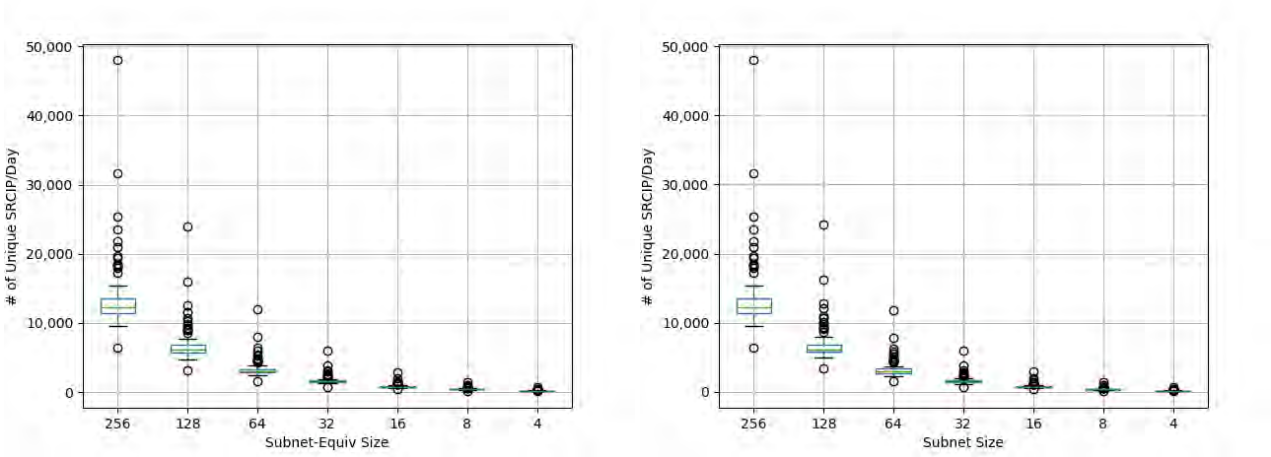
Raw Data Summary of Unique SRCIP/Day

This appendix contains additional box-plots that summarise the distribution of unique SRCIPs in each data sample using the box and whisker plots. This is for both random and sequential sampling. The box plots that have been added in the main body and their interpretation can be found in **Section 4.3**



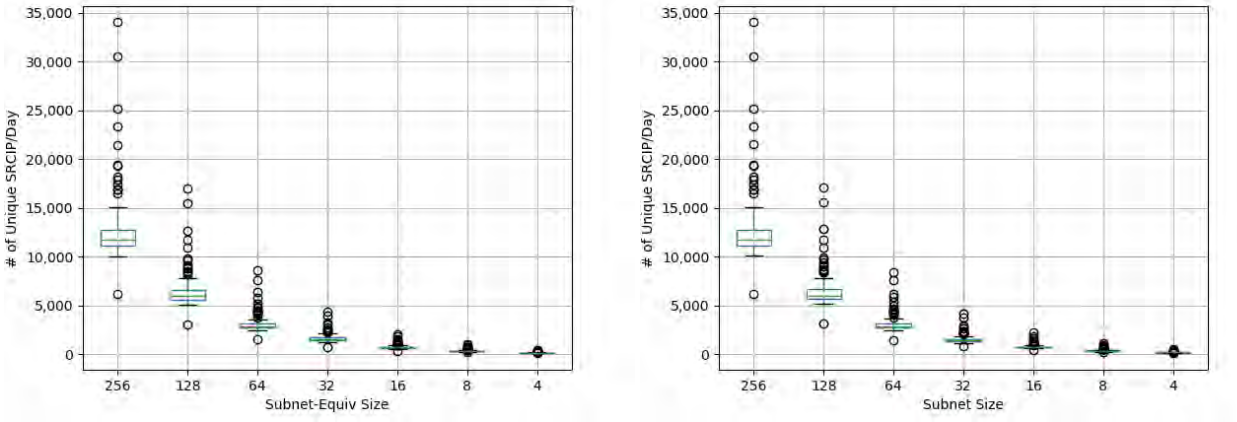
(a) Random Samples (b) Sequential Samples

Figure H.1: 146/8: Data Summary of No. Unique SRCIP/Day



(a) Random Samples (b) Sequential Samples

Figure H.2: 196-A/8: Data Summary of No. Unique SRCIP/Day



(a) Random Samples (b) Sequential Samples

Figure H.3: 155/8 - [Jan - Mar]: Data Summary of Unique SRCIP/Day

I

Sequential Sampling Subnet Hierarchy

This appendix contains a table that shows sequential sampling subnet hierarchy that was used in this research study. The left column indicates how many DSTIP each subnet is expected to have while the right hand column shows how many subnets each level of level has. The work related to this table is explained in **Section 4.1.2**.

IP Count	Subnet Hierarchy (Sequential Sampling Net-mask)																																
256	/24																																
128	/25	/25																															
64	/26	/26	/26	/26																													
32	/27	/27	/27	/27	/27	/27	/27	/27																									
16	/27	/27	/27	/27	/27	/27	/27	/27	/27	/27	/27	/27	/27	/27	/27	/27																	
8	/28	/28	/28	/28	/28	/28	/28	/28	/28	/28	/28	/28	/28	/28	/28	/28	/28	/28	/28	/28	/28	/28	/28	/28	/28	/28	/28	/28	/28	/28	/28	/28	
4	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	⟶
	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	/29	
2	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	⟶
	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	⟶
	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	⟶
	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	/30	

Figure I.1: Sequential Sampling Subnet Hierarchy