2018

URI's NetFlow Traffic Logs' Behavioral Analysis and Monitoring Visualization Tool

Semhar Kessete Gebregiorgis
University of Rhode Island, th.co.gt@gmail.com

URI'S NETFLOW TRAFFIC LOGS' BEHAVIORAL ANALYSIS AND MONITORING VISUALIZATION TOOL

BY

SEMHAR KESSETE GEBREGIORGIS

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE

REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

IN

COMPUTER SCIENCE

UNIVERSITY OF RHODE ISLAND

2018

MASTER OF SCIENCE THESIS

OF

SEMHAR KESSETE GEBREGIORGIS

APPROVED:

Thesis Committee:

Major Professor Lisa DiPippo

Natallia Katenka

Noah M. Daniels

Yan Sun

Nasser H. Zawia
DEAN OF THE GRADUATE SCHOOL

UNIVERSITY OF RHODE ISLAND
2018

ABSTRACT

As long as the Internet users and the dependency of human on IT are evolving, the detailed inspection of NetFlow data will be useful, especially for the detection of cyber anomalies and outbreaks. To date, numerous researchers have examined NetFlow with respect to numerical fields including, for example, Packets, IPs, Bytes, and Bandwidth consumption. But only a handful of projects have paid attention to the analysis of NetFlow activity using categorical fields including Internet application and computer location, especially concerning a particular academic institution.

The primary focus of this project is on the development of a tool for analyzing NetFlow activity at the University of Rhode Island (URI) computer network. This tool helps to monitor the NetFlow activity over time stratified first by the Primary and then by the Secondary fields selected by the user. NetFlow activity is evaluated and visualized with; frequency of traffic flow – if user only selects filter option 'Primary Log Field', and relative frequency of traffic flow – after selecting Field value of interest from 'Primary Log Field' if user continues and select filter option 'Secondary Log Field'. Automatically, the drill-down of data through those log fields along timestamp of interest will trigger the generation of an advanced log table grid view.

Additionally, the proposed tool takes advantage of the network theory and provides visualization of the bipartite graph representation of NetFlow data subset with selected fields and time period with pre-specified sets of node degrees. This representation helps to monitor and characterize communication behavior of individual nodes in the selected time period.

Overall, the tool created for this project can be regarded as the first step in the development of the comprehensive cyber security system for monitoring and analysis of the URI NetFlow activity.

ACKNOWLEDGEMENTS

First and foremost, I would like to thank God for giving me family with such virtuous quality that words can't describe. I owe my success to my parents (Mr. Kessete Gebregiorgis and Mrs. Abrehet Mehari); the most hardworking people I have ever known in my life and who have lovingly nurtured the person I have become and their urge on the pursuit of my desire, as well as my big-hearted and altruistic brother, Medhanie.

It is a genuine pleasure to express my deep sense of gratitude to Dr. Lisa DiPippo and Dr. Natallia Katenka without whom this project wouldn't be a realization. I thank both of them for the dedication, and constructive and insightful feedback they have provided me throughout my journey in my thesis project. Their timely and scholarly response has helped be both develop and craft my work better.

I owe a deep sense of gratitude to Mrs. Elizabeth Larimer, Mrs. Lorraine Berube and Dr. Joan Peckham for their carrying support, kindness, enthusiasm and dynamism throughout my stay in the University of Rhode Island.

I thank Profusely all staffs of the department of Computer Science and Statistics for their kind help and cooperation during my entire study period at URI.

It is my Privilege to thank my noble beloved husband as well as my rock, Mr. Ftsum Asfaha for his encouragement throughout my research period and life itself.

I am also extremely thankful to my friends and the University of Rhode Island.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

FIGURES PAGE

FIGURES PAGE

FIGURES PAGE

CHAPTER 1

1. INTRODUCTION

After the advent of internet, it has been quite a while since cyber-attacks (threats) became the most infamously threatening phenomenon (notorious virtual combats) to both large and small corporations whether private (individuals and international) or government owned, which transparency of their activities and information depends on the greater internet. Evolving at rapid pace, many attacks have been triggered viciously throughout the past couple of decades. Malicious pieces of software have compromised and tormented computer systems [3]. DDoS attacks have put many services of internet-based resource request in question [34], phishing emails have tricked a lot of ordinary users and vulnerable applications have been exploited to access internet resources [3] – not to mention that all the attacks that are generated by unintentional misconfiguration or entities that don't use their resources wisely. So far these are some of the threats that are recognized. The next question is, therefore, what kind of threats could be expected after P2P and IOT (Internet of Things) botnet? In these networks data no more resides in only one key servers and nor the access in the network is restricted [14].

So far, numerous mechanisms have been introduced and developed to make a network environment secure. Even though different types of firewalls including NIDS (Network Intrusion Detection System) and NIPS (Network Intrusion Prevention System), proxy servers, all-in-one security appliances, and other mechanisms are put to place intended to make a network environment safe, it is never enough to challenge every new

threat outbreak. The main challenge by far is creating and implementing a defense mechanism ahead of an outbreak as most are thought of after blast wreak havoc on a system. So ideally, to combat cyber-attacks, mechanisms have to be dug out to estimate when and what type of attack to expect next.

The era we are living in is all about the power of big data science and its impact in most governmental and non-governmental business applications and research developments [44]. ISPs (Internet Service Providers) and many other companies have situated scrutinizers at the edge of their Internet switches/routers to persistently collect and store network traffic flow records, which are ready for retrieval and information display within short time of intervals. At the core of all this is NetFlow analysis, which enables IT-teams to identify threatening communication patterns by maintaining baselines of end system behaviors. In order to bring a significant impact in NetFlow characterization performances, detection and forensics analysis have to be shifted from the edge to the core of the Internet NetFlow by being engaged with the visual pattern of the flow.

Historically, IT organizations focused heavily on perimeter network security to protect their networks from cyber-attacks. Protection at the perimeter edge works well for data moving in and out towards the protected system. Recent breaches, however, show that the perimeter security alone is not sufficient to combat the advanced persistent threats [14]. To protect the network against the emerging threats, IT organizations need intelligent solutions that are pervasive, behavior-based and complementary to the current zone-based security solutions. One such solution is to leverage the network infrastructure itself to function as a sensor by activating the network to collect IP traffic flows and

deploy anomaly detection system based on network behavior analysis (i.e. *NBA-Network behavior analysis*). This is done in order to detect suspicious traffic flows, policy violations and compromised endpoints by analyzing network traffic for unusual behavior, events or trends [38].

First-generation network security tools do not provide adequate visibility to factor in dynamic network topologies**.** The increasing level of sophistication and customization of malicious attacks are forcing organizations to implement stringent security measures in their networks to alert them of any potential security breach incident before it occurs. As such, *Network behavior analysis (NBA) or Network traffic monitoring is fast becoming a necessary part of any network along with anomaly detection systems to ensure comprehensive network security*. Security analysts recommend that IT organizations should deploy ***Network behavior analysis*** solution in addition to the ***perimeter-based security*** solutions such as ***firewalls and intrusion prevention system (IPS)*** in their networks as part of security and threat protection strategy. **Behavior-analysis-based systems** are often able to detect security threats such as malware, viruses and botnets against which other security tools often were ineffective. Such systems (Behavior-analysis-based**)** boosts security through monitoring Network traffic flow and keeps track of unusual actions, events or trends [38].

In line with the recommendations of security analysts, the **University of Rhode Island (URI)** has deployed a system that captures ***network traffic flows*** in order to perform ***behavior-based analysis***. In addition to the main attributes available through NetFlow protocol that includes applications, source IP/Port, destination IP/Port and

others, the URI NetFlow Log has names of departments. This paper will look through the URI network behavior-based analysis of the logs recorded on a daily basis.

University of Rhode Island is one of the institutes that uses NetFlow or IPFIX features and functions [1, 38] to cache, export and collect (to present data) network traffic flows for analysis. In the metadata of a NetFlow, the firewall and other network security probes can block signature-based or learned threats, but with the technology going fast, the mind behind malicious intentions has sprinted in a way that cannot be understood easily. Network traffic activity's convolutedness and its nature of being very hard to understand made the level of carrying out accurate characterization of network traffic flows more difficult.

Subsequently, this project's aim is to address the significance of categorical Log Fields in a NetFlow to identify and analyze behavioral difference using a tool. Log Field in a NetFlow analysis would refer to standard field names of any traffic system logs. The tool can help a user to filter out any preferred subset of traffic Log with the help of **Primary** and **Secondary** fields' values and be able to see the activity of the prevalent flow over a selected period of flow time in a form of relative or raw frequency to begin with. Another feature in the tool will generate a traffic network graph along with its activity line graph depending on the degree of communication a user demands from a list of available degrees. This presentation then provides an IT personnel with a visualized network graph that enables them to see and investigate all involved nodes (hosts) in detail. The network graph is set to show the communication between IP/Port of the nodes. The tool also has an additional feature of displaying table grid view of any preferred subset of data or filtered NetFlow data.

Therefore, this thesis paper is arranged as; Chapter 2 – will review literatures on tools related to my tool and will discuss some future enhancement of tool limitations, Chapter 3 – will discuss the methodologies and procedures toward the tool implementation and pre-processing of Netflow log for data refinement, Chapter 4 – will discuss on how developed tool operates along with result discussion, and Chapter 5 – is conclusion.

CHAPTER 2

2. RELATED WORKS AND FUTURE ENHANCING APPROACHES

Great number of researches has been conducted on monitoring flow activity, detection and identification of anomalous behaviors in a NetFlow. Some number of these researches has been conducted on the analysis of NetFlow traffic with a support of a visualization tools. These visual monitoring tools of enormous raw sequence flow data have significantly aids the aim to enhance the awareness of malicious NetFlow patterns. Excerpts of some of these studies and tools are highlighted for this thesis in a form of written and summary tables (See **Table 2A**, for Tools summary).

2.1. NETFLOW ANALYSIS TOOLS

NetFlow logs analysis is time-consuming and undeniably not an easy job to perform without the application of analysis-tools particularly implemented for the task. Therefore, to ease the task of detection of interesting information, mechanisms (tools) with visualization effects were designed. [5, 7, 20, 22, 26, 29, 41, 42] are some developed related tools, which are looked into for this thesis paper. They generally flow from highest-level[1] to lowest-level[2] semantic constructs visualization. At the very least, most of the tools require IP address information with or without Port information and some might also require byte and/or packet counts and/or Protocols. All those related tools

[1] Overall network visualization

[2] On-demand drill-down help users find more detail information they are interested in

incorporate the use of IP address and/or Port/packet details as well as Protocols in use but failed to include internet-sites/utilities/applications to monitor flow activities.

NVisionIP [26] is a tool that visually presents Class-B[3] IP addresses' state of traffic flow on a single screen for situational awareness. The display is based upon either number of bytes transmitted or the flows to and from hosts in a network. The tool has three stages; *Galaxy view* (highest-level construct) - hosts filtered based upon useful attributes in classification of security incidents, *Small-Multiple view* - statistics view of small subnet hosts (particular area from *Galaxy view*), and *Machine View (*Lowest-level construct*)* - view detailed information about single host from the small-multiple view. With the stages, NVisionIP tool can suggest various attacks such

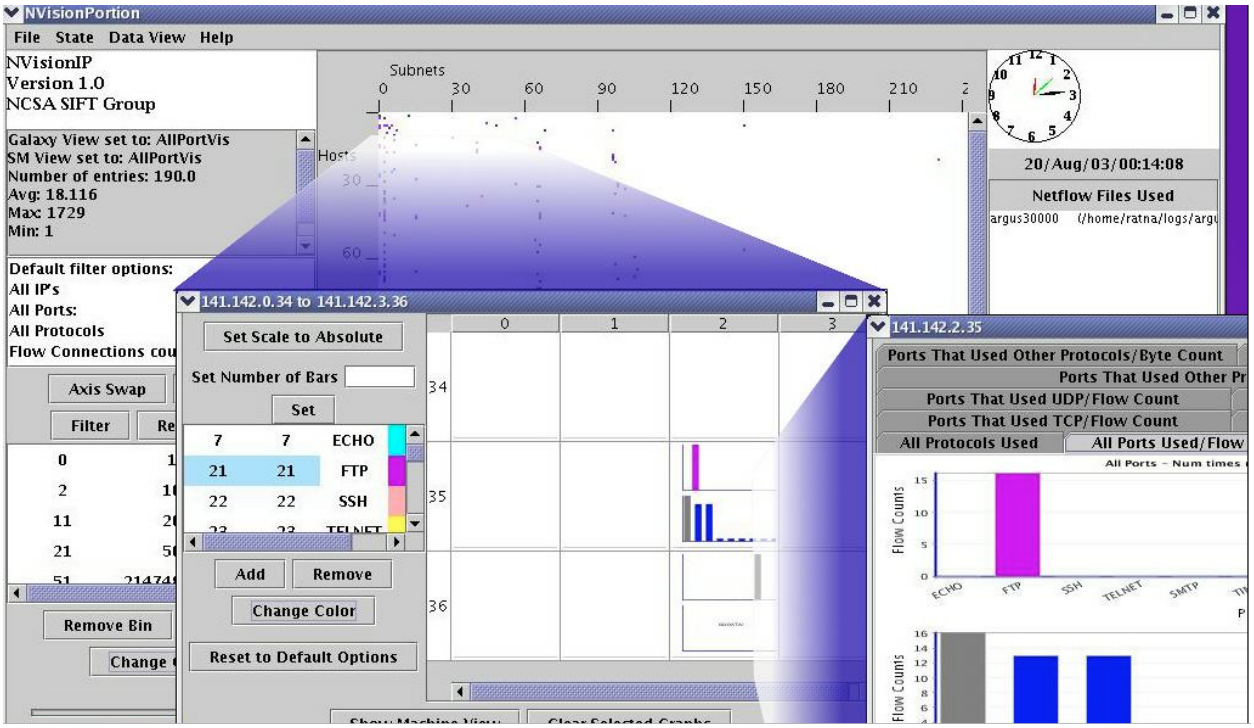


Figure 2A – ***Galaxy, Small-Multiple and Machine Views*** of NVisionIP (Courtesy of NVisionIP development team)

[3] **Class B** - IP Addresses range from 128.0.x.x to 191.255.x.x. Its default subnet mask is 255.255.x.x. It has 16384 (2^{14}) Network addresses and 65534 (2^{16}-2) Host addresses.

as Worm Infection, Compromised Systems, Port Scans, Denial of Services and Misuses of computer network.

VisFlowConnect-IP [42, 43] is a NetFlow visualization tool also implemented to enhance security awareness of administrator by detecting aberrant traffic between an internal (local) network and external domains. This tool goes through three stages to generate network flow *animation* - (i) Extraction of NetFlow logs, (ii) Processing and storing of important statistics being carried out via analysis, and finally (iii) Display visualizations of the derived statistics in *Parallel axes Views* [42]. The tool display consists of some variety *Interface views;* (1) *Global View* (Figure 2B) is the initial (high-level) overview that has three vertical parallel lines showing - left [right] axis line that represents external domains that are sourcing [receiving] flows to [from] the internal network (center axis line), (2) *Domain View* is on demand (drill down) view of selected external domain and the internal network with inherited features as *Global view,* (3) *Internal view* is represented by two axes showing traffic flows that are entirely within the scope of internal (local) network, and (4) *Host Statistics View* (Figure 2C) is a statistical table view of number of total bytes being transferred to and from individual machines.

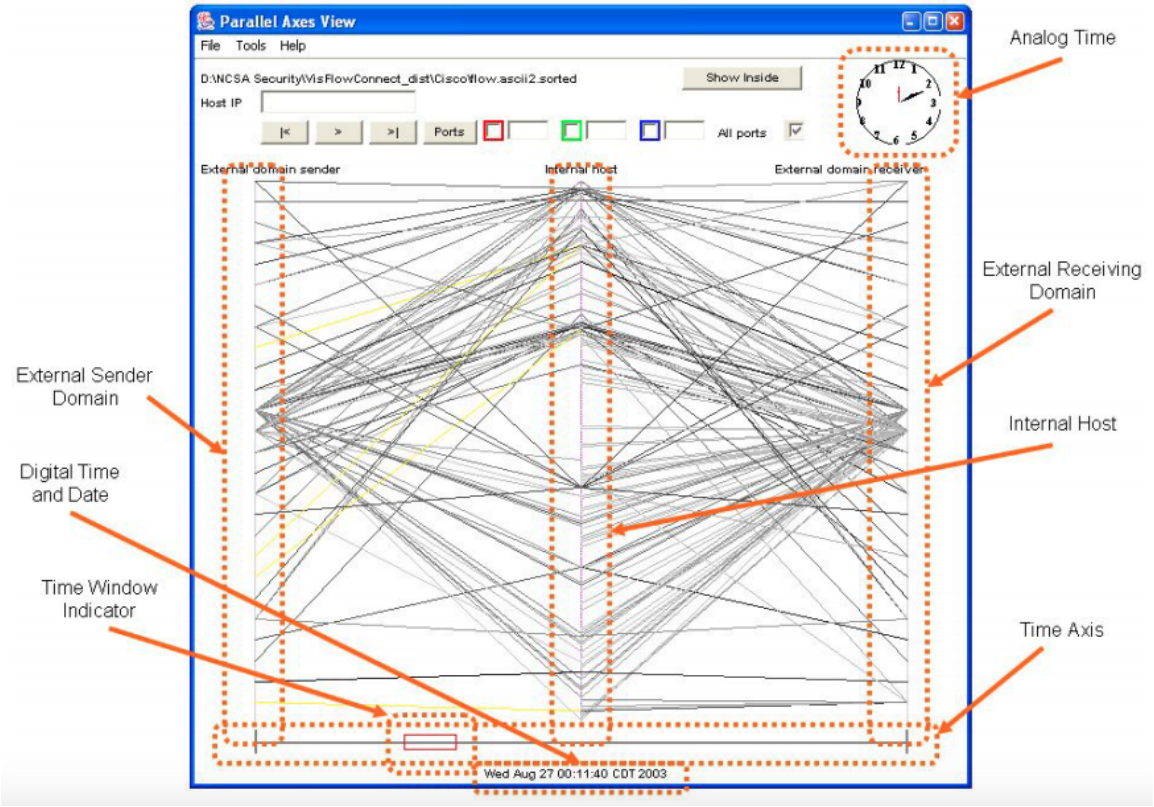
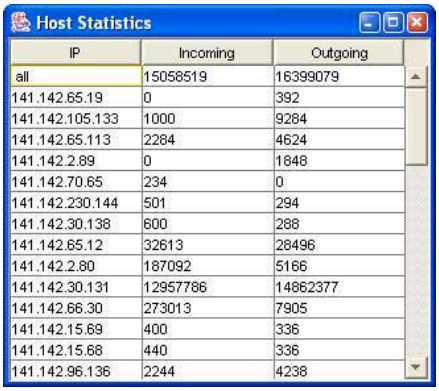


Figure 2B – ***Global View*** *of VisFlowConnect* (Courtesy of VisFlowConnect development team)



Host Statistics

IP	Incoming	Outgoing
all	15058519	16399079
141.142.65.19	0	392
141.142.105.133	1000	9284
141.142.65.113	2284	4624
141.142.2.89	0	1848
141.142.70.65	234	0
141.142.230.144	501	294
141.142.30.138	600	288
141.142.65.12	32613	28496
141.142.2.80	187092	5166
141.142.30.131	12957786	14862377
141.142.66.30	273013	7905
141.142.15.69	400	336
141.142.15.68	440	336
141.142.96.136	2244	4238

Figure 2C – ***Host Statistics View*** *of VisFlowConnect* (Courtesy of development team)

Both NVisionIP [26] and VisFlowConnect-IP [42] tools have similar filter settings (options). NVisionIP has {IPs, Ports and Protocols} and VisFlowConnect has {Ports, Protocols, Transfer rates and Packet sizes} as their default filter options. Whereas

my tool has advanced filtering options over these two tools; (1) a user can filter Netflow data with as many fields as the Netflow log record includes, and (2) as well as, to help user analyze traffic flow with granular filtering with some additional aggregated category fields (See **Section 3.2**) also embedded to the Netflow data before processing. My tool also provides user with some statistical-summaries and brand-new on-demand relative-frequency (see **Section 3.3.1**) presentation of traffic usage distribution among certain Netflow log fields values, which NVisionIP tool insufficiently presents in its '*Machine View'* as a bar chart of only Ports usage with respect to flow [byte] counts and VisFlowConnect presents in table form of only total number of bytes flowing [in and out] out of individual machines.

FloVis [41] is another related tool designed to provide security analysts with interactive flow-level analytic visualization of network traffic. It integrates three visualization approaches toward its analytic presentation. (1) *Activity View* (Figure 2D) is one segment of the visualization tool presented with a two-dimensional grid of host activity per unit of time with colored squares indicating activities and distinction in behaviors, (2) *FlowBundle View* (Figure 2E) is another segment of the tool presented with '*B-spline curves connecting entities in a circle'* that exhibit connections between hosts or subnets in a Netflow record via colored lines indicating flow direction between entities, and (3) *NetBytes View* (Figure 2F) is the third segment of the tool that is triggered to focus on the volume of flow data transacted in to [out of] a individual entities over a period of time.

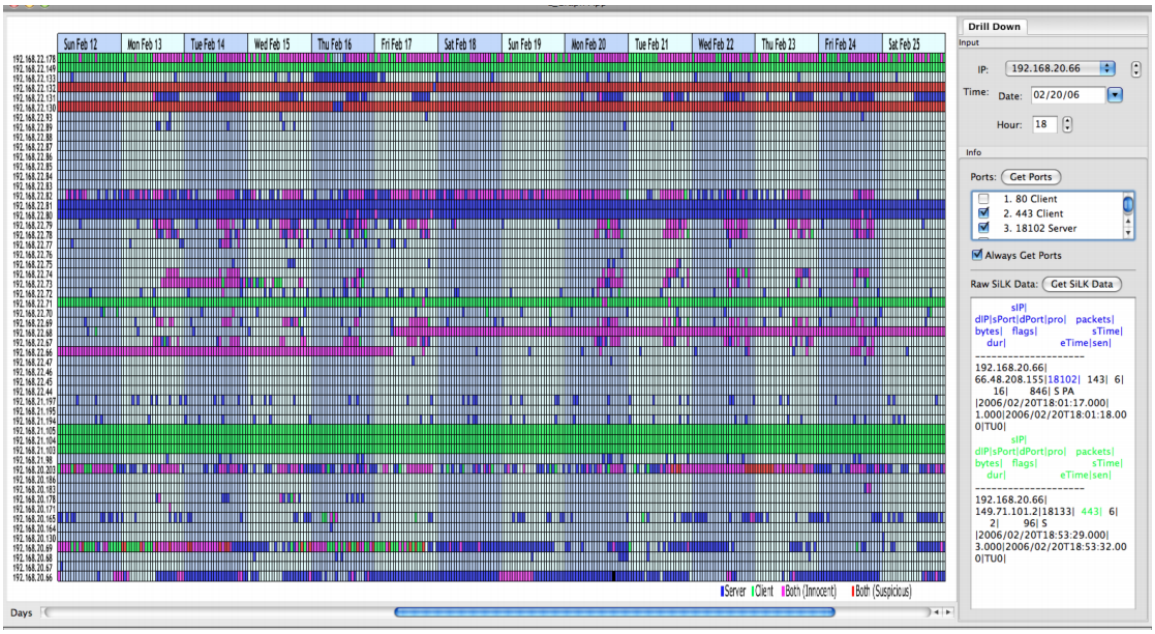


Figure 2D – ***Activity View*** *of FloVis* (Courtesy of development team)

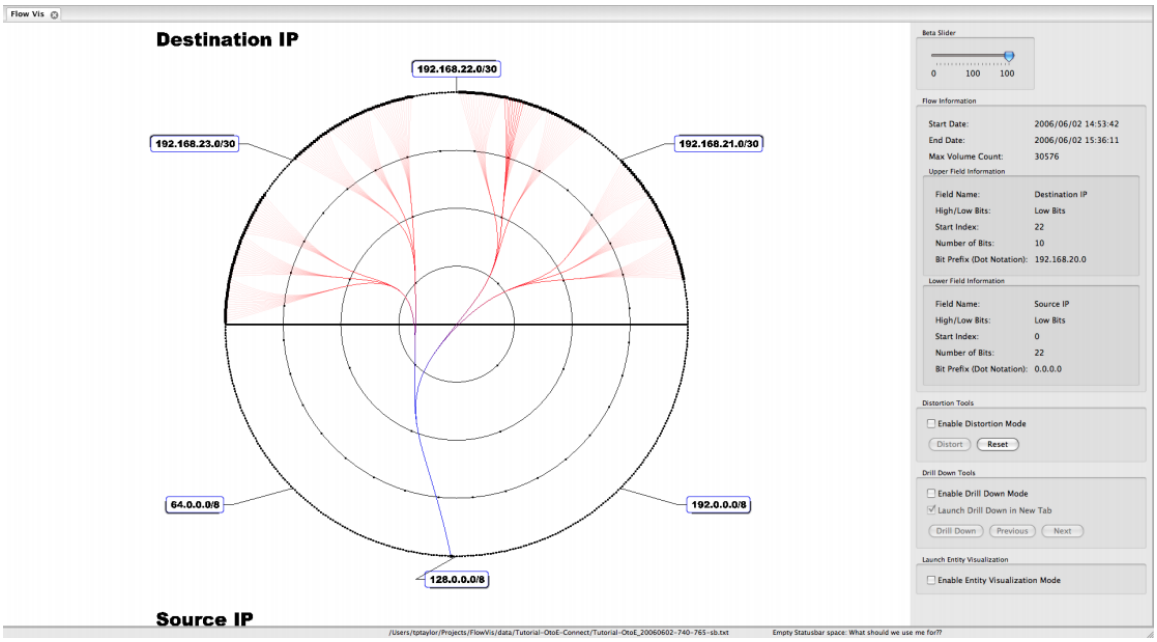


Figure 2E – ***FlowBundle View*** *of FloVis* (Courtesy of development team)

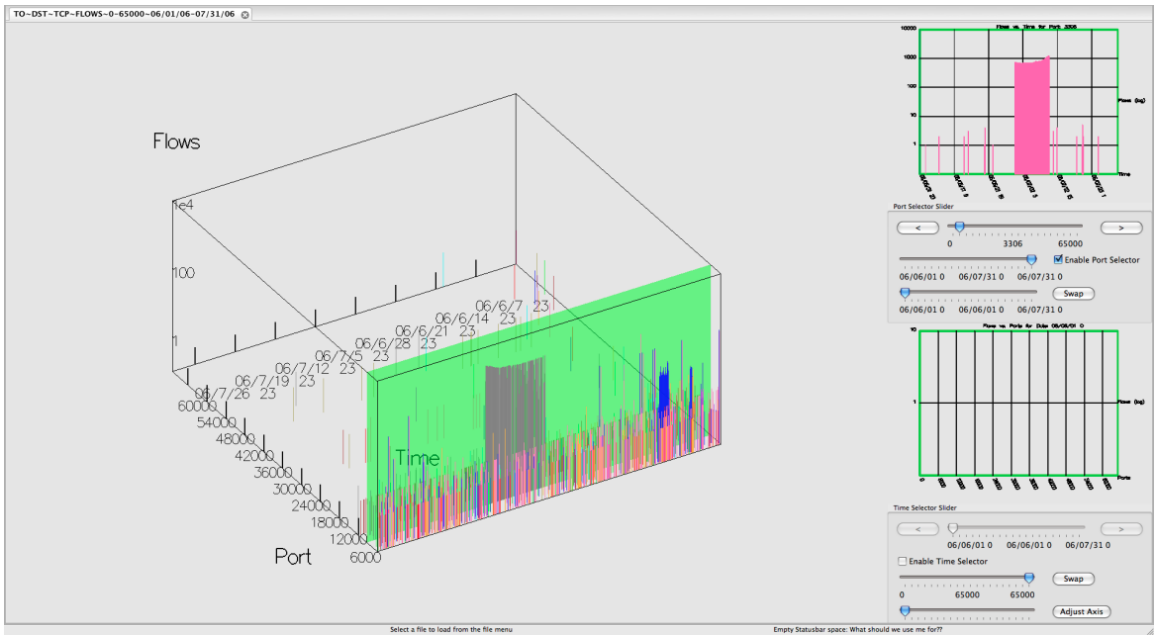


Figure 2F – ***NetBytes View*** *of FloVis* (Courtesy of development team)

Nfsight [7] is a tool related to FloVis [41] that is designed with a proposition of high practicality in presenting Netflow status awareness over other existing tools by then. Its technical and visual approaches are compartmentalized into three components. (1) A web interfaced front-end that provides administrators with a capability to visualize large-scale network traffic activities as well as on-demand drill-downs, (2) A *service detector* that is implemented to correctly distinguish Servers from Clients via set of heuristics and Bayesian inference, and (3) Once *service detector* sets clients and servers apart in bidirectional manner *Intrusion detection* will take over to identify malicious activities with '*graphlet'* detection approaches.

In a similar trend, TNV (Time-Based Network Traffic Visualizer) [20] is a visualization tool that is designed to enhance and aid navigational scope of analysis task of Network traffic. It embraces procedures for filtering links and selecting area of significance to assess packet detail. Matrix visualization is the main visual component of

TNV where matrix displays of time (across x-axis) with respect to available IP addresses hosts (across y-axis) in a data are shown. To warrant further investigation, Packet numbers within each communication of hosts are colored. Those colors are user-defined colors that correspond number-of-packets. To display all involved source ports of a specific IP, the tool enables users to highlight a specific port. And analysts can query a filter from emphasized links and propagate packet details on demand to learn behaviors of their network and evaluate intrusion activities such as port scanning.

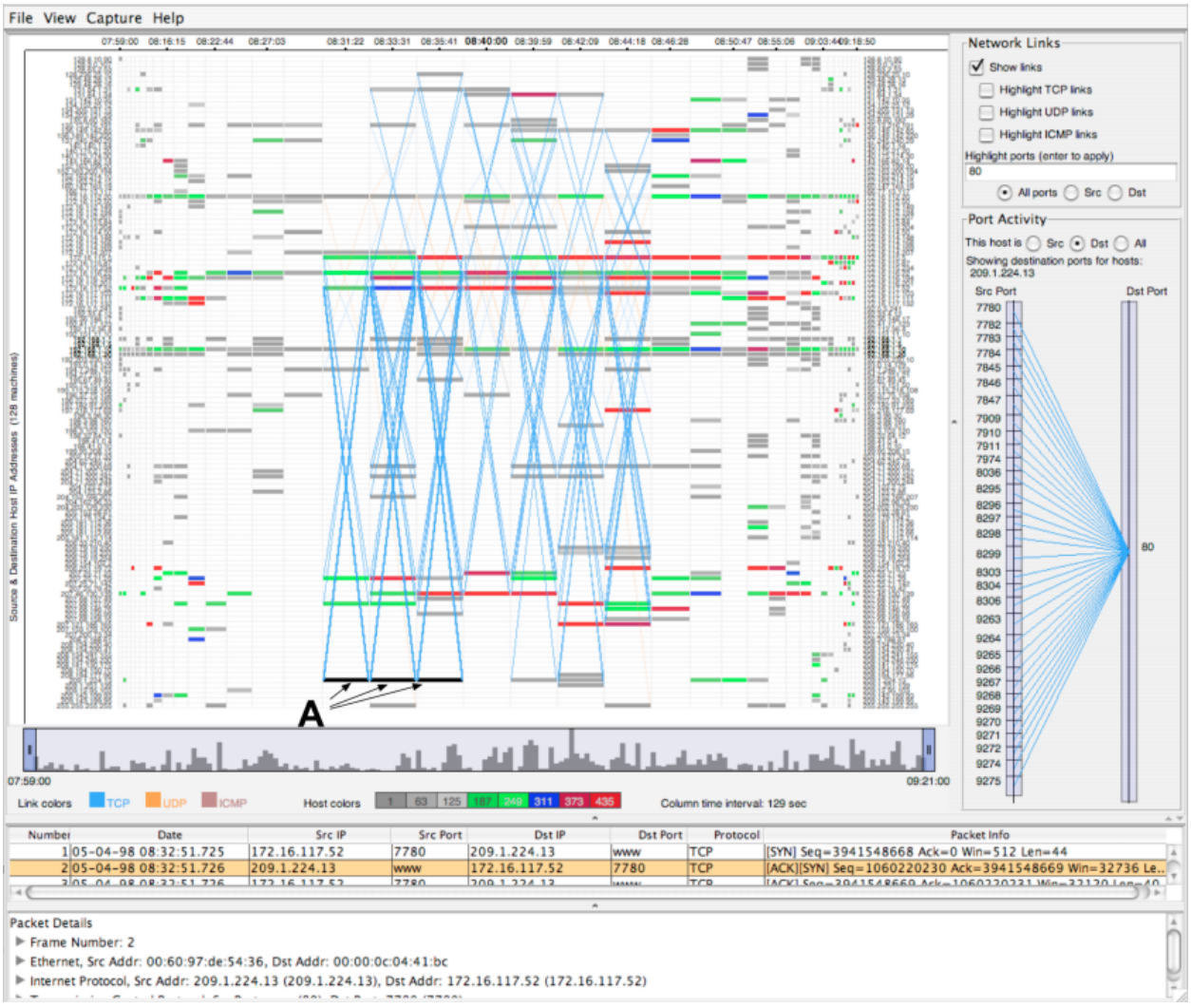


Figure 2J – **TNV** tool (Courtesy of TNV development team)

Even though FloVis [41], Nfsight [7] and TNV [20] clearly implemented useful overviews of particular areas of interest and provide security analysts with needed situational awareness, they do have some shared shortcoming that advanced by my tool. They clearly lagged behind from being able to see prevalently occurring situations that my tool triggers through flow comparison among aggregated categories of Netflow field values. In general, they lack any additional technique of granularity overview of Network flow other than IPs and Ports.

PortVis [29] is a tool that produces visualization of network traffic in single screen with axes those correspond some important features of flow data such as time and Port numbers. Similar to earlier mentioned tools, PortVis also support both high-level and low-level semantic construct visualization. Its operation is to alert analysts for any security event that causes eloquent changes while monitoring network activity with respect to Ports activity frequencies. The drawback of this tool is its failure to identify the traffic causing suspicious patterns because PortVis works with a summarized flow data that only has counts of the most significant Netflow fields such as Source and Destination IPs for Ports.

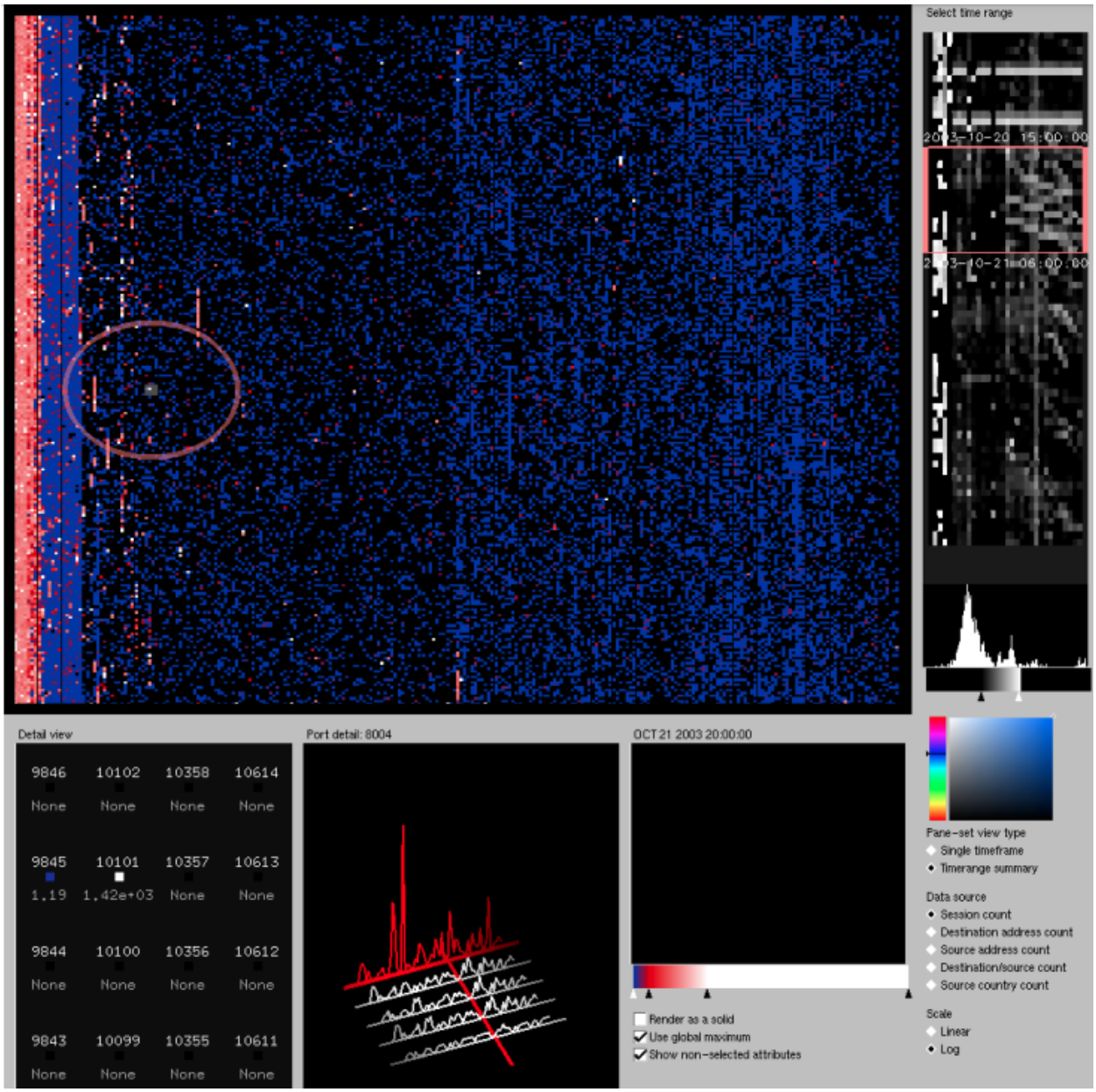


Figure 2G – **PortVis** tool (Courtesy of PortVis development team)

Field	Example
Protocol	TCP
Port	80
Hour	2003-10-20 3:00am
Session count	1,443
Unique source addresses	342
Unique destination addresses	544
Unique source/destination address combinations	411
Unique source countries	20

Figure 2H – **PortVis** Summarized Input Data (Courtesy of PortVis development team)

VISUAL (Visual Information Security Utility for Administration Live) [5], is a related tool that requires; (i) [IP, Ports, Protocols and time of observation] of network traffic, and (ii) text file of list of internal IP addresses [which is a requisite in order for the tool to map IPs in grid] to operate. It is designed to present a small to mid-size network data. In its layout, small squares forming larger grid view and markers outside the grid indicates home (internal) and external hosts respectively connected with communication edge [where edge color indicates replay status]. Basically, the overview display incorporates internal vs. external hosts concept of a network. Functionally the tool maneuvers chronologically as, "overview first, zoom and filter, then detail on demand" of network traffic. This results in bringing insights to network analysts about the home hosts that are actively interacting with larger number of external hosts and external hosts that receives communications from large number of internal hosts. The shortcoming of this tool is its failure to work with large-scale Netflow records.

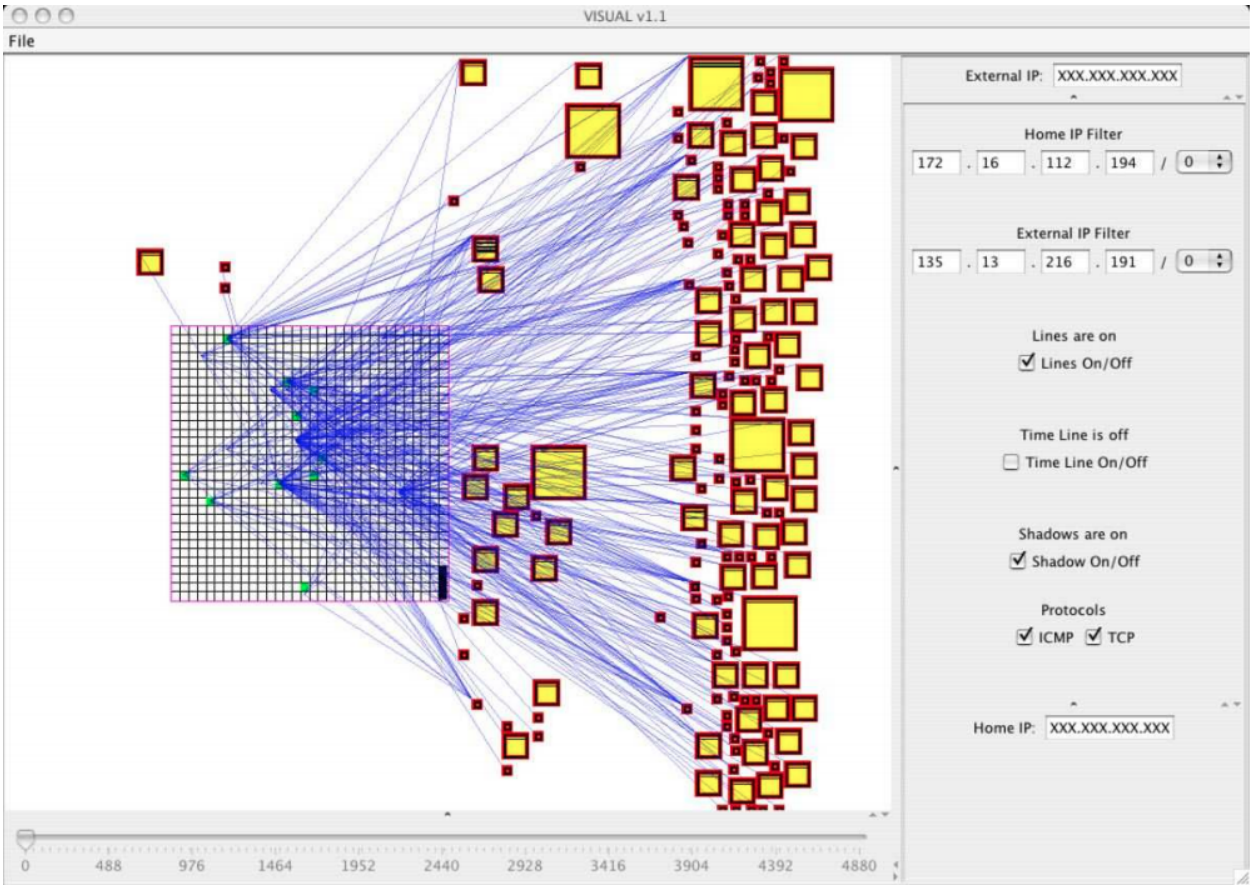


Figure 2I – **VISUAL** tool (Courtesy of VISUAL development team)

Tools	Research Year	Open Source Status / Availability	Platform / Functionality	Input Data	Output Result (Section **2.1**)	Real-Time/ Offline/
NVisionIP [26]	2004	http://security.ncsa.uiuc.edu/distribution/NVisionIPDownLoad.html → Provided Source's page can't be found	Not Specified	CISCO NetFlows[4] & Argus[5] flows	Galaxy, Small-Multiple & Machine Views	Offline
VisFlowConnect-IP [42, 43]	2006	http://security.ncsa.uiuc.edu/distribution/VisFlowConnectDownLoad.html/ → Provided Source's page can't be found	Not Specified (work in progress	NetFlow log file (Argus or NFdump or Cisco format version 5 or 7)	Interface (Global, Domain & Internal) Views	Offline
FloVis [41]	2009	http://www.flovis.net/ → Page can't be found	Built in OpenGL (runs in most Operating Systems)	SiLK[6] data	Activity, FlowBundle and NetBytes Views	Real-time Scalable
Nfsight [7]	2010, 2014	http://nfsight.research.att.com/ (Latest release → 2014)	Built in Nfsen[7] (Perl) & PHP (runs in *NIX systems)	NetFlow by Nfdump/Nfsen	Web based Network Activities View, Service and Intrusion detections	Real-time Scalable

[4] Record contains: - (1) source/destination IP addresses and ports, (2) number of bytes and packets, (3) start and end timestamps, and (4) protocol type

[5] http://www.qosient.com/argus/

[6] https://tools.netsa.cert.org/silk/ - A SiLK packaging system collects IPFIX, NetFlow v9, or NetFlow v5 and converts the data into a more space efficient format, recording the packed records into service-specific binary flat files.

[7] Nfsen. http://nfsen.sourceforge.net, 2010.

PortVis [29]	2004	Not Provided	Not Specified	Summary[8] of large data of IP traffic (flows to/from Internet)	Port-based statistics view, & network and Port scan detection	Not Specified
VISUAL [5]	2004	Not Provided	Not Specified	TCPDump[9] or Ethereal	Graphical overview of communication (internal vs. external perspective) pattern	Real-time Scalable
TNV [20]	2005	Not Provided	Built in Java using the Sourceforge jpcap[10] library	Standard TCPDump data	View of temporal traffic flow, inspection of network link and state, detail of individual packets	Real-time or offline

Table 2A – Related Tools Summary Table

[8] The data is reduced to a set of counts of entities. For instance, instead of a list of each TCP session, there is a field that specifies how many TCP sessions are present; instead of a list of source IP addresses, a field specifies how many different source IP addresses were present. (See **Figure 2H)**

[9] Tcpdump public repository, June 2004. Communication packets sent between computers. Packet data includes: Source and destination IP address, source and destination port (for TCP/UDP), protocol, and time of observation.

[10] http://jpcap.sourceforge.net/ - utilizes libpcap, a widely deployed standard for network packet capture.

2.2. ANOMALY DETECTION APPROACHES FOR FUTURE WORK

Since this tool is only a preliminary step towards URI (University of Rhode Island) Netflow analysis, feature such as traffic flow anomaly identifiers and alert triggers needs to be included in close future. And instead of only practical for URI, this tool could be off working for other organizations with trivial improvements. Such as, for instance, the home IP address that is already predefined in code as of only URI's should be coded to be dynamically interchangeable for the corresponding organization using the tool. One way or the other, the following research projects addresses some principal points on Netflow anomaly detection, that could be considered towards the future enhancement of this tool.

Even though general method for detecting anomalies in network traffic is an important yet unresolved problem [30], nevertheless, using Network flows it should be possible to observe most anomaly behaviors by inspecting traffic flow [30]. As well as use of subspace method to detect anomalies by using different types of traffic flows is in [9].

SnortView [22] is a tool with slightly different objective than the previous mentioned related works. This visualization system is designed to reinforce the correctness of Network-based Intrusion Detection Systems (NIDS) in customizing their signature DB (Database) as well as minimize false detections. The system takes in NIDS logs and creates alarms criteria of false detections heuristically from analyzing every alert. The visual presentation of the tool is a Matrix frame of the interaction between the hosts that are causing the generation of the alert/s. The most essential information that is placed in consideration to substantiate the alarms is Time, Type (services and protocols),

Source and Destination of access. The tool magnifies alerts as symbols (shapes) within the grid of the display frame constructed with axes (IP address and Time).

Paul Barford and David Plonka [6] study on the characteristics of network traffic flow anomalies pays particular attention to a precise description of anomalous behavior to assist tools that help manage a network. In their study [6], they used flow level data to generate much smaller data sets that are sufficient to determine aberrant behaviors in the network traffic. This research is similar to [28] in that a detailed discussion of source packet and flow level feature datasets are set forth. The most crucial step in [6] is the clustering of the anomalies into three groups based on the similarities and differences observed in the data flow. The three groups that they identified are: network operation anomalies, flash crowd anomalies and network abuse anomalies. While Network Operation anomalies are significant differences in network behavior caused by configuration changes, Flash Crowd anomalies are caused by software release or a sudden interest after publicity that causes a rapid increase in traffic flow. Network abuse anomalies, as categorized by the researchers, are ordinary denial of services and port scan attacks. In [6], the tools they used to rigorously cluster and identify the anomalous behaviors include simple statistics, time series analysis and wavelet analysis. The research only focuses on three major categories of anomalies and overlooks other forms of anomalies that can be identified in thorough analysis of data flows. My paper will complement this research by providing a detailed analysis of the data flow on the basis of application. Their packet-level based measurement also further scrutinized on application-accessed level analysis. Likewise, a detailed classification of various anomaly

detection methods, systems and tools and characteristics of attacks have also been given in [18].

Lakhina et al. [25] addresses the problems involved in diagnosing traffic anomalies that "may span multiple links in a network, using link-based statistics" [25]. Their diagnostics process involves three steps: the use of a general method to identify anomalies in a network, incorporation of distinct methods to identify each type of anomaly and finally quantify them. The paper emphasizes on the correct detection, identification and quantification of anomalies. For this purpose, the 'Principal Component Analysis' system is used to separate a NetFlow into normal and anomalous subspaces. Here the researchers introduce a specific variant of traffic anomalies called "volume anomalies" that refers to a sudden change in the Origin-Destination flow traffic. Traffic anomalies, according to [25] are those that are prevalent when traffic is viewed in ten minutes interval. The paper establishes applicable terms that define the curves between anomalous and normal network traffic.

Francois, et al. [19] were able to detect peer-to-peer based Botnet in a NetFlow using graph dependency model and adapted Google PageRank algorithm. In [19], the computation was performed with open-source based Hadoop cluster. The Project aims at detecting new generation of botnets from large dataset of NetFlow data. A Botnet, according to the paper is a network of compromised hosts (bots) that are controlled by an attacker also called the Botmaster that sends commands via a C&C (Command and Control) channel [19]. The research project involved the usage of 720 million NetFlow records (77GB) covering only 23 hours of a major Internet operator in Luxembourg. The first step in their project is the gathering of NetFlow records via exports from routers to a

collector. This then produces an interaction between hosts that are ready for analysis. The data is then used to produce a dependency graph that can be analyzed in a PageRank [19]. The PageRank is then executed on Hadoop by distributing the adjacency matrix of the dependency graph among all data nodes. Though the research is based on synthetic dataset and primarily focuses on Botnet, it still is substantial approach toward identifying Botnets.

Exhaustive surveys on network anomaly detection tools methods and systems have also been conducted in [8] and [4]. In [8], structured overview of different forms of network anomaly detection is drawn. The research basically presents different categories of anomalies based on computational techniques. The research also discusses tools that can be used as network defenders and directions that future researchers need to follow in identifying and detecting anomalies. Unlike prior surveys conducted before it, [8] presents a wide variety attacks and up-to-date methods, systems, and tools of analysis. Comprehensive survey of a graph-based anomaly detection and description are constructed in [4]. The research emphasizes on the categorization of outliers and anomalies in unstructured collection data points.

More specifically, Coull, et al. [13] tried to evaluate the strength of current 'Anonymization Methodology' in achieving goal by focusing on providing a realistic assessment of the feasibility of identifying individual web pages within anonymized NetFlow logs [13]. Their work distinguishes itself from prior work by operating on flow-level data rather than packet traces and examines many of the practical concerns associated with implementing such identification within real network data. They concluded that there are certain web pages whose behaviors are so variable that they may

be difficult to detect in practice. The research underlines that thorough analysis is to be made on individual website visited indiscriminately.

In [11, 35, 39], it is suggested that, with proper analytic methods being employed can provide greater detail in the identification of aberrant behaviors in a network. Though these studies deal with network security in different manners, they share common factors; such as putting in consideration a serious study and investigation of network flow for proper analysis of abnormal network behaviors and suggestive ways in which their analysis can be done. These studies assert that despite the difference in the methods incorporated to identify anomaly behavior, the major task in identifying these behaviors is 'learning the network in which all activities take place,' which according to them harbors the channel with which these behaviors can be spotted.

CHAPTER 3

3. METHODOLOGY AND PROCEDURE

3.1. OVERVIEW OF NETFLOW LOG

Netflow [37] is an industry-standard network protocol technology developed by CISCO that answered questions such as; (1) what end-systems are using the most bandwidth, and (2) is traffic traversed your Internet routers related to your company operations. It is a technology that the firewall can use to export information (statistics) about IP network traffic as flow enters or exits an interface. The Firewall exports the information to Netflow-collector[11] in a form of NetFlow Fields. Like a phone bill, Netflow provides actual metadata from the traffic flow traversing through a device – but not the actual detail of data packet. By metadata[12], it implies that collected Netflow Log records at least have attributes (Log Fields) such as source and destination IP addresses, source and destination Ports, Protocol type, Class of Service, and Router/switch interface of data packet (illustrated in Figure 3A).

[11] A server that analyst use to analyze network traffic for security, administration, accounting and troubleshooting purposes.
[12] Data that provides descriptive information about other data.

Figure 3A – NetFlow Attributes (Log Fields) - NetFlow Record Provides a Significant Amount of Information (Courtesy of CISCO)

Depending on their configuration, NetFlow enabled Network routing devices can have certain standard Syslog types such as Traffic-Logs[13], Threats-Logs[14], Config-Logs[15] and more [2] - that would be forwarded to an external server for the purpose of both [either] storage and [or] application of any substantial analysis. Those Netflow Logs comprise a number of Fields that describes the details of each network communication. Brief list of standard Syslog types along with their Field descriptions can be found in [2]. Without the knowledge of Log Field values description, it is hard to analyze and imply a hypothesis on NetFlow data. As such it is based on the description provided in [2] that this thesis paper distinguished and removed irrelevant Log Fields. All traffic-log Field names, which are observed in URI's traffic records, along with their description, and possible values some Log Fields can reflect, are presented in Table 3A.

[13] Indiscriminately display an entry for the start and end of each traffic flow session.
[14] Display entries when traffic matches one of the Security Profiles attached to a security rule on the firewall.
[15] Display entries for changes to the firewall configuration

Log Field Name	Description
Receive Time (receive_time)	Time the log was received at the management plane
Serial Number (serial)	Serial number of the firewall that generated the log
Type (type)	Specifies type of log; values are traffic, threat, config, system and hip-match
Subtype (subtype)	Subtype of traffic log; values are start, end, drop, and deny Start—session started End—session ended Drop—session dropped before the application is identified and there is no rule that allows the session. Deny—session dropped after the application is identified and there is a rule to block or no rule that allows the session.
Generated Time (time_generated)	Time the log was generated on the data plane
Source IP (source_address)	Original session source IP address
Destination IP (destination_address)	Original session destination IP address
NAT Source IP (natsrc)	If Source NAT performed, the post-NAT Source IP address
NAT Destination IP (natdst)	If Destination NAT performed, the post-NAT Destination IP address
Rule Name (rule)	Name of the rule that the session matched
Application (application)	Application associated with the session
Source Zone (from)	Zone the session was sourced from
Destination Zone (to)	Zone the session was destined to
Session ID (sessionid)	An internal numerical identifier applied to each session
Repeat Count (repeatcnt)	Number of sessions with same Source IP, Destination IP, Application, and Subtype seen within 5 seconds; used for ICMP only
Source Port (source_port)	Source port utilized by the session
Destination Port (destination_port)	Destination port utilized by the session
NAT Source Port (nat_source_port)	Post-NAT source port
NAT Destination Port (nat_destination_port)	Post-NAT destination port
Flags (flags)	32-bit field that provides details on session; this field can be decoded by AND-ing the values with the logged value: 0x80000000 —session has a packet capture (PCAP) 0x00400000 —session has a NAT translation performed (NAT) And more.

Protocol (ip_protocol)	IP protocol associated with the session
Action (action)	Action taken for the session; possible values are: allow—session was allowed by policy deny—session was denied by policy drop—session was dropped silently And more.
Bytes (bytes)	Number of total bytes (transmit and receive) for the session
Bytes Sent (bytes_sent)	Number of bytes in the client-to-server direction of the session
Bytes Received (bytes_received)	Number of bytes in the server-to-client direction of the session
Packets (packets)	Number of total packets (transmit and receive) for the session
Start Time (start_time)	Time of session start
Elapsed Time (elapsed_seconds)	Elapsed time of the session
Category (category)	URL category associated with the session (if applicable)
Sequence Number (seqno)	A 64-bit log entry identifier incremented sequentially; each log type has a unique number space.
Action Flags (actionflags)	A bit field indicating if the log was forwarded to Panorama
Source Location (source_country)	Source country or Internal region for private addresses; maximum length is 32 bytes
Destination Location (destination_country)	Destination country or Internal region for private addresses. Maximum length is 32 bytes
Packets Sent (pkts_sent)	Number of client-to-server packets for the session
Packets Received (pkts_received)	Number of server-to-client packets for the session
Action Source (action)	Specifies whether the action taken to allow or block an application was defined in the application or in policy. The actions can be allow, deny, drop, reset- server, reset-client or reset-both for the session.

Table 3A – University of Rhode Island Netflow (Traffic Log) Fields (attributes) and their description (Description Resource - Palo Alto Networks)

As it is this tool's main objectives to develop and conduct a system that serves URI (University of Rhode Island) in the visual analysis and monitoring of its NetFlow logs, the analysis of this work will be based on the NetFlow logs of URI. The focus of this analysis has precisely taken in consideration only Traffic Logs. From over 40 Log

Fields that are known to network analysts, only the indispensable and significant log Fields along with some miscellaneously added Fields (see Section 3.2) are selected towards the analysis process.

The University of Rhode Island (URI) is an institute which has Class-B IP network architecture, which accommodates 65,536 (2^{16}) unique IP addresses of suffix [131.128.*.*]. That being the case, to monitor its massive network traffic activities, NetFlow logs are collected from an edge router, that routes flow data to and from the URI-Campus network and the greater Internet. It captures network traffic activity as Logs per each day at collection sampling rate based on 1:100 packets [10]. The captured Log data size of wired (Ethernet) network traffic of a day might extend from about 100 MB to 160MB (each flow data containing about half a million or more records that is only Wired without considering Wireless communications). This figure only represents the Log records of Spring 2014. Table 3B below shows typically grouped Netflow Log information each flow Cache entry contains at URI.

General		**Source**	**Destination**	**Time**	**Miscellaneous**
Type - drop Action - deny Application Rule - Src_NAT Category - any Padding IP Protocol - tcp Log Action	Bytes Received Bytes Sent Repeat Count Packets Packets Received Packets Sent Seq Number Flags	Source Address Source Port Source Zone Source Country Nat Source Port	Destination Address Destination Port Destination Zone Destination Country Nat Destination Port	Receive Time Start Time General Time Elapsed Time (Sec)	Source Department Source Description

Table 3B – University of Rhode Island NetFlow Log Fields Group-Categories; shaded terms are some of the possible Field values.

3.2. LOG FIELDS REFINEMENT AND UPGRADE

In its existing format, it might be possible to observe some characteristics of the URI Netflow data through a spreadsheet application such as Microsoft Excel, but only in a limited fashion. As such graphical and statistical analysis of the data characteristics requires more advanced tool. A far more powerful approach is to parse the data with a preferable language to remove parts of it that are irrelevant and include additional Log Fields in advance to the process of flow analysis. Therefore, to reformat the data into a form that is conducive for analysis and visualization, a statistical and other computing software environment comes to application.

Palo Alto Networks as a network and enterprise security company suggests that firewall policy (Next generation Firewall) should be built based on Applications, users and content [31]. And most network security companies deploy application as one of business initiative at the firewall policy level. But questions like, 'what rule network security companies does (as CISCO) follow to identify applications' can arise. The process starts with a so-called process App-ID. App-ID classifies all traffic across all ports first and foremost (if application exists in existing application database). As soon as traffic hits the firewall the mechanism begins to see what the application is. It starts looking across all ports using heuristics, application decoders, protocol decoders, SSL decryptions[16], and signatures and once it identifies the application and that becomes the basis for the policy. So instead of allowing port 80, now you are allowing specific application such as ORACLE, Gmail, or box.net, or any other of that matter [31]. When the firewall is unable to identify an application using the App-ID, the traffic is classified

[16] Standard security technology for establishing an **encrypted** link between a web server and a browser.

Application Classifications	
Category	The application category will be one of the followings: business-systems collaboration General-internet Media Networking unknown
Subcategory	The subcategory in which the application is classified. Different categories have different subcategories associated with them. For example, subcategories in the collaboration category include: email, file-sharing, instant-messaging, Internet-conferencing, social-business, social-networking, voip-video, and web-posting. Whereas, subcategories in the business-systems category include auth-service, database, erp-crm, general-business, management, office-programs, software-update, and storage-backup.
Technology	The application technology will be one of the followings: **client-server**—An application that uses a client-server model where one or more clients communicate with a server in the network. **network-protocol**—An application that is generally used for system-to-system communication that facilitates network operation. This includes most of the IP protocols. **peer-to-peer**—An application that communicates directly with other clients to transfer information instead of relying on a central server to facilitate the communication. **browser-based**—An application that relies on a web browser to function.

Table 3C – Classifications of application from Palo Alto Networks [application - Netflow Log Fields]

as unknown (unknown-tcp or unknown-udp). This behavior applies to all unknown applications except those that fully emulate HTTP [31]. This App-ID initiative in a

Netflow is the main reason why my thesis paper tries to emphasize the application Log Field and its categories (Table 3C) in the use of Netflow monitoring and analysis.

No doubt that a well-categorized data provides unambiguous information. Hence, the applications under the URI Netflow Field - 'application' are categorized into standard categories for a better picture (see Table 3D). Fortunately, Palo Alto Networks [32] provides a neat open source categorization of applications (Table 3C). This order of classifications groups each application to which Category, Subcategory and Technology it belongs. Except for some applications that fail to have the standardized names, all the applications in the URI NetFlow logs have matched the applications in [32] and those categories are incorporated into the URI's NetFlow logs accordingly (see Table 3D). The need for the addition of application categories to the Netflow Log is for two purposes: 1) since Network traffic flows at high volume and complexity is very hard to have a visualization of entire flow at once; therefore, blending of those classification can assist in the reduction of analysis granularity, and 2) helps to draw an insight on the communication technology utilized by each application.

The URI NetFlow logs have a log Field that shows the URI departments from where a communication is sourced/destined. In this tool an additional field that corresponds to building is also incorporated in the NetFlow logs. This additional field contains the corresponding building for each department. The field is obtained from the information provided under 'Directory for Buildings & Departments' in the URI map [15]. The first step taken towards including the 'Building' field is to alter the department names provided under the field 'iso_dept' to meet the standardized name format as it is in the directory [15]. The next step was to append new table fields to the NetFlow dataset

(shown in ***Table 3D -*** Field name 'Department') so that the dataset has standardized names that can be associated. Since a single building can hold multiple departments, this modification of fields can reduce the granularity in the analysis of the data. Through these steps it will be much easier to identify the physical location in which apparent network communications take place. The following table shows part of the NetFlow Log dataset that is used in the project. The fields marked with asterisks (*) are the newly added ones and the Fields without asterisks (*) are the benchmarks for the new Fields added ones.

...	application	App_Category*	App_Subcategory*	Technology*	iso_dept	Department*	Building*
	100bao	general-internet	file-sharing	peer-to-peer	Computer Science	Computer Science and Statistics	Tyler Hall

Table 3D – An Example of URI NetFlow Log with the **newly incorporated Fields**

In addition to this, there were some mismatches of log Field names in the URI NetFlow dataset that needed to be fixed before further proceeding with the analysis. At times, the Field name does not match the Field value. There were, for instance, two fields names represented as one. There were also some application names that were not standardized according to the list of applications provided in [32]. For instance, in the URI NetFlow dataset Instagram was recorded as 'instagram' while it should have been recorded as 'instagram-base' the way it appeared in [32]. Hence this resulted in the collapse of the field information detail in that there was a mismatch when generating analysis.

Due to some abnormality detected by the security appliances (such as firewall) deployed at the edge of URI routers, some of the records are observed to have 'deny'

under the ‘action’ field name within the Netflow logs. Such records are excluded from the NetFlow Log to be considered for analysis. The reason for the exclusion of those records is because they lack Log information such as having “not-applicable” under the ‘application’ attribute which means their signature is already in the database of IDS (Intrusion Detection Systems).

All communications that are sourced from hosts outside URI are represented with ‘DNE’ label under the field ‘iso_dept’. DNE is a standard label applied to indicate data that does not exist. Whether malicious or benign, we need to know the flows target (destination) Departments/Buildings. In order to accommodate this, the ‘DNE’ labels need to be replaced with the appropriate Departments/Buildings. So, an aggregated dataset (IP-Department) that contains all unique URI IP addresses with their associate Departments/Buildings has to be created from 90-days NetFlow datasets. Then with the aid of the IP-Department dataset the replacement of ‘DNEs’ (found under ‘iso_dept’ field of NetFlow logs) with corresponding Department was performed. This process enhanced the parameter with which the tool analyzes the incoming NetFlows.

3.3. ANALYTICAL PROCEDURE

3.3.1. DESCRIPTIVE STATISTICS AND IMPLEMENTATIO

Statistical summary (Descriptive statistics) is a method that quantitatively describes (summarizes) features of a collection of information or set of observations as simplified as possible in order to communicate a largest amount of information. Some of the commonly used Statistical descriptions that present compressed format of set of observations are measurement of central tendency (mean ($\bar{X} = \frac{1}{n}\sum_{i=1}^{n}(f_i)$; where n is the

number of observations and f_i is i^{th} observation), median ($Q_2 - 2^{nd} quartile\ or\ 50^{th}\ percentile$) and mode), minimum ($min\{f_{i\ldots n}\}$) and maximum($max\{f_{i\ldots n}\}$) values, as well as quartiles ($Q_1 - 25^{th}\ percentile$ & $Q_3 - 75^{th}\ percentile$) and outliers (Lower-bound[17] > Outliers > Upper-bound[18]) of an ordered observations [27]. In general, these statistical summaries can provide sufficient information as to what a set of quantitative data is composed of. Without even having a thorough view of the data set, one can have preliminary assumptions based on these statistical summaries. Therefore, the extraction of those significant statistical summaries will bring forth the founding details of the analysis.

Prior to the tool's visualization of the analysis, the NetFlow Log dataset is sliced into sections of unique timestamps. This is done in order to generate activity frequency per each time stamp for a magnified observation. The frequency is then presented graphically with a Line Graph that can assist in identifying the pattern of flow. This derived frequency can help users (NetFlow administrators) to compare previous patterns and draw hypothetical inferences and in turn establish a foundation to a baseline for future assessment of NetFlow activity. With this, users can evaluate significant peaks and lows and pinpoint at the aberrant activities based on their current observation.

The assessment of frequencies that are generated per each timestamp requires the application of statistical summary described above such as mean, median, minimum and maximum frequencies, quartiles, and outliers (if any) per distinct subset of Log Field value/s of interest. The application of the statistical summary toward this analysis is

[17] Q1 - 1.5 (Q3 - Q1)

[18] Q3 + 1.5 (Q3 - Q1)

computed in two ways: (i) the statistical summary that is applied to the frequencies obtained from **single Log Field value** in a given range of time and (ii) the statistical summary that is computed from the mean line obtained from each slice of timestamp of **multiple Log Field values**. Let, the set of unique timestamps from selected time range be $= \{t_1, t_2, \ldots, t_i\}$, and the set of unique selected Log Field values be $= \{V_1, V_2, \ldots, V_k\}$.

I. Then, if user selects **single Log Field value (V_1)**, then the statistical summary that would be computed will be from the set of frequencies obtained from each unique timestamp per that individual Log Field value. That is - the statistical summary derived for the set of frequencies denoted as $\left\{V_{1f_{t_1}}, V_{1f_{t_2}}, \ldots, V_{1f_{t_i}}\right\}$. For instance, the mean of the set of frequencies would be denoted as, mean $(\bar{X}) = \frac{1}{n}\sum_{i=1}^{n}\left[V_{1f_{t_i}}\right]$; where $V_{1f_{t_i}}$ is the frequency of V_1 at the i^{th} timestamp.

Demonstration with instances: Let the following Table 3E be a summary table of frequencies. Assuming user prefers to see traffic flow frequencies for some URI departments in a time range.

		Frequency of flow Per Departments				Mean-Line
		Tyler Hall (V_1)	Taft Hall (V_2)	Graduate School (V_3)	Pharmacy (V_4)	
Unique Timestamps (hh:mm)	t_1 =08:30	$V_{1f_{t_1}}$= 9	$V_{2f_{t_1}}$= 0	$V_{3f_{t_1}}$= 5	$V_{4f_{t_1}}$= 2	→
	t_2 =08:31	$V_{1f_{t_2}}$=4	$V_{2f_{t_2}}$=5	$V_{3f_{t_2}}$=0	$V_{4f_{t_2}}$=8	→
	t_3 =08:32	$V_{1f_{t_3}}$=5	$V_{2f_{t_3}}$=6	$V_{3f_{t_3}}$=2	$V_{4f_{t_3}}$=0	→

	t_4=08:33	$V_{1f_{t_4}}$=7	$V_{2f_{t_4}}$=9	$V_{3f_{t_4}}$=0	$V_{4f_{t_4}}$=1	→
Mean ($\bar{X}$)		↓	↓	↓	↓	↓ Mean of Mean-Line

Table 3E – A Sample summary table of traffic flow frequencies per some departments in URI.

II. If m unique multiple Log Field values (in Table 3E, 4 departments) are selected, then the statistical summaries that would be computed will be, let us for instance calculate the mean ($\bar{X}$) of the frequencies obtained from each Log Field values:

A. From the flow frequencies obtained from each timestamp per each Log Field values mean ($\bar{X}$) would be computed as $\frac{1}{n}\sum_{i=1}^{n}\left[V_{1f_{t_i}}, V_{2f_{t_i}}, V_{3f_{t_i}}, \ldots, V_{mf_{t_i}}\right]$. In Table 3E, it is at the most-bottom row denoted as arrows vertically pointing downward (↓) for each Department, to be Mean ($\bar{X}$) values.

B. The statistical summary of the last column denoted as 'Mean-Line' in Table 3E. The Mean-Line generated by; (i) first computing mean values of frequencies from m number of Log Field values in each timestamp (in Table 3E denoted as horizontal arrows pointing to right – each arrow indicates mean value computed from its corresponding row – that is, $\bar{X}_{each\ row} = \frac{1}{m}\sum_{i=1}^{m}\left[V_{if_{t_1}}, V_{if_{t_2}}, \ldots, V_{if_{t_n}}\right]$) will result in giving the 'Mean-Line' values as $\rightarrow [V_{f_{t_1}}, V_{f_{t_2}}, \ldots., V_{f_{t_n}}]$, and then (ii) draw the statistical summary of the 'Mean-Line' itself, for instance in Table 3E the mean ($\bar{X}$) of the 'Mean-Line' is denoted as '(↓) Mean of Mean-Line', formulated as $(\bar{X}_V) = \frac{1}{n}\sum_{j=1}^{n}\left[V_{f_{t_j}}\right]$.

Table 3E can also be generally formulated as matrix → $\begin{bmatrix} V_{1f_{t_1}}, V_{2f_{t_1}}, V_{3f_{t_1}}, \ldots, V_{mf_{t_1}} \\ V_{1f_{t_2}}, V_{2f_{t_2}}, V_{3f_{t_2}}, \ldots, V_{mf_{t_2}} \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ V_{1f_{t_n}}, V_{2f_{t_n}}, V_{3f_{t_n}}, \ldots, V_{mf_{t_n}} \end{bmatrix}$; where each column (m number of columns) represents Log Field values and the corresponding line graph generated with this matrix would have m plot traces, and each row would be the frequencies obtained in n timestamp of m traces. Figure 3B below shows sample Line-graph with two frequency traces (Field values V1 and V2) of two Log Field values and the generated corresponding 'Mean-Line' trace (Mean-Line value V).

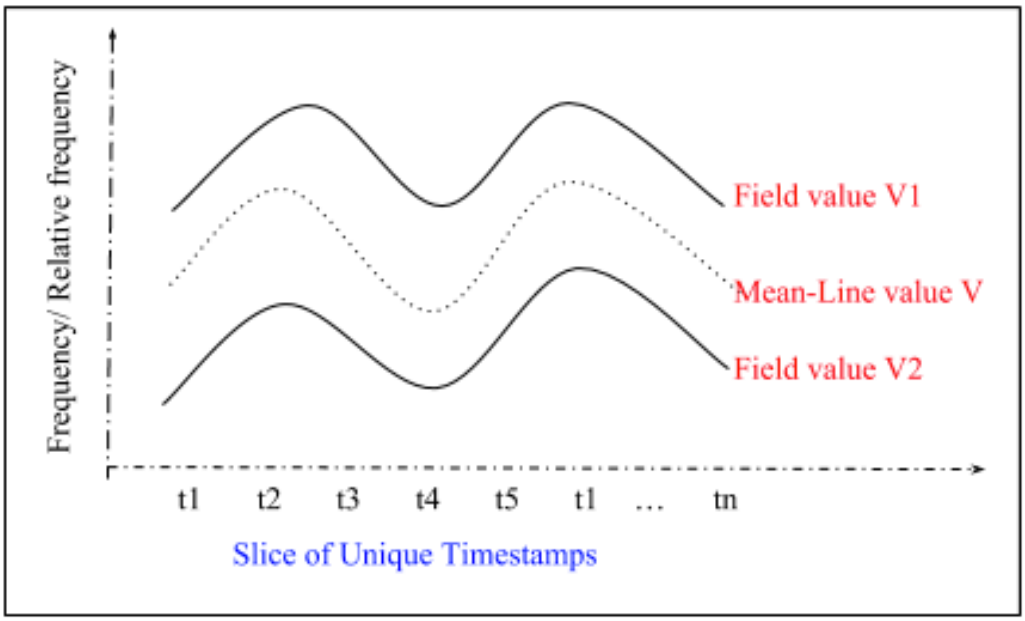


Figure 3B – A Sample frequency Line-Graph that the Tool would generate along unique slices of timestamp.

3.3.2. GRAPH-BASED DESCRIPTION AND IMPLEMENTATION

In Graph theory, graphs are mathematical structures used to pair wisely model relationships between objects. Formally, a model of network can be represented as a graph by G (V, E), with V denoting the set of nodes and E the set of edges in graph G. Any graph G can be classified as directed or undirected based on the condition whether the edges have sense of direction. Likewise, graphs can also be divided into weighted and unweighted graphs depending on the strength of edge weight they are assigned, where a graph that has an equal weight is unweighted and weighted graph if its edges are assigned with different strengths values. In this tool's graph presentation, we will only focus on undirected and unweighted graphs. For an undirected and unweighted, the connectivity pattern can be characterized as an N × M matrix named adjacency matrix A whose value a_{ij} $(i = 1, \ldots, N, j = 1, \ldots, M)$ is 1 if there exists an edge between node i and j or 0 if none. In graph G (V, E), 'Degree' is the number of edges incident to a vertex. The Degree of any node i is the number of edges linked to it and is computed as $K_i = \Sigma_{j \in G} a_{ij}$, where a_{ij} is the i^{th} row and j^{th} column element of an adjacency matrix A. Degree is a simple measurement for the connectivity of a node with the rest of the nodes in a network [17].

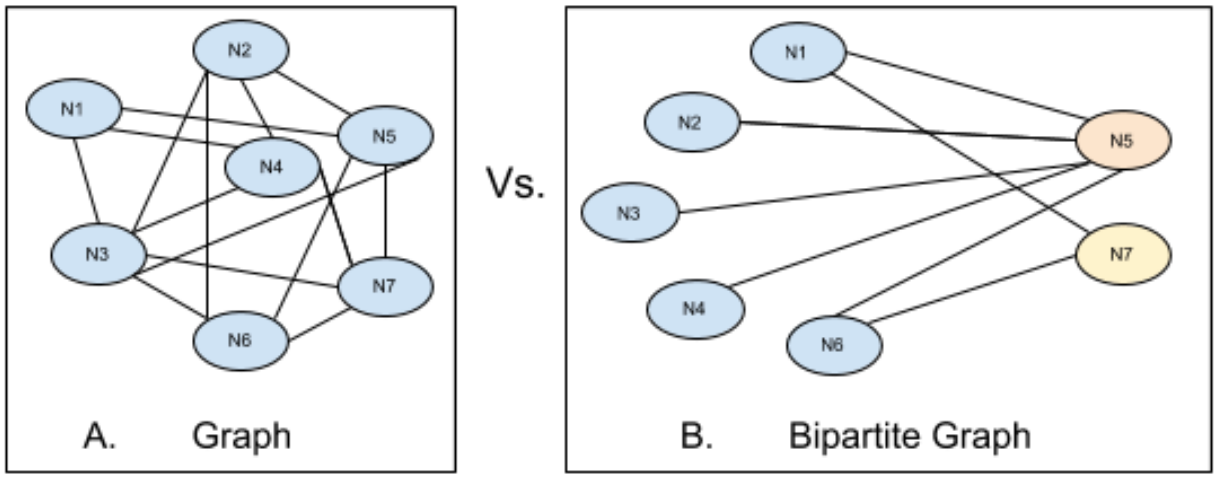


Figure 3C – Visual illustration of Graph (Left) and Bipartite Graph (Right)

NetFlow Log data can be represented by a bipartite graph [17] consisting of edges and nodes/vertices. In this section, there is an introduction of some relevant network graph models [17] and brief description analysis methods that can help in the flow characterization of Source-Destination IP/Port. A bipartite network is a graph model G = (V, E), such that the vertex set V may be partitioned into two disjoint sets, and each edge in E has one endpoint in V1 and V2. In a Netflow record, a host that initiated (sourced) IP traffic could also appear in the set of reception (destination) hosts or vise-versa. So, with this bidirectional behavior of communication, bipartite graph presentation of network flow would not be applicable. But my tool will present an occasion of either outgoing [from URI] traffic flow or incoming [to URI] traffic flow, that is basing URI-hosts (internal) as one set (V1) and all external hosts being as another set (V2) of nodes. So, at this point the need for a bipartite graph comes to application.

Once URI traffic flows are partitioned into egress/ingress traffics through advanced tool filtering options, the bipartite graph are used to represent the Source

IP/Port (V1) and Destination IP/Port (V2) in the URI network. With the help of the visualization of the constructed bipartite graph, one could be able to view inbound or outbound communications between Source IP and Source Port, Source IP and Destination IP/Port, Source Ports and Destination Port/IP, and Destination Port and Destination IP (see Figure 3D). User can also observe on how Ports are actively used and their usage conformity. For instance, once a tool user filters (drills-down) Netflow data to a Field value of interest through accessed Applications' - Category/Subcategory/Technology, the bipartite graph with the tool filtering features (Chapter 4) can help analysts to view what standard ports (as well as IPs) were in use between the interacting internal (URI's) and external hosts.

Through the bipartite graph, one can practically observe the communication frequency denoted as node degree associated with each vertices (hosts). If there is a matching edge between two hosts then there is at least a communication or a NetFlow tie between two nodes. The graph also provides on-hover communication count info, edge thickness and color representing communication density, and vertices' communication densities are distinguishable through their node radius and colors. This bipartite graph then will bring us to the next step of the analysis methods in which a line (plot) graph of communication frequency per unique timestamp will be generated. This frequency graph will incorporate the methods described in (section 3.3.1) where the flow frequency of a selected node/s (IP/Port) with designated (selected) degree in a given range of timestamp is displayed (see details of Tool in Chapter 4).

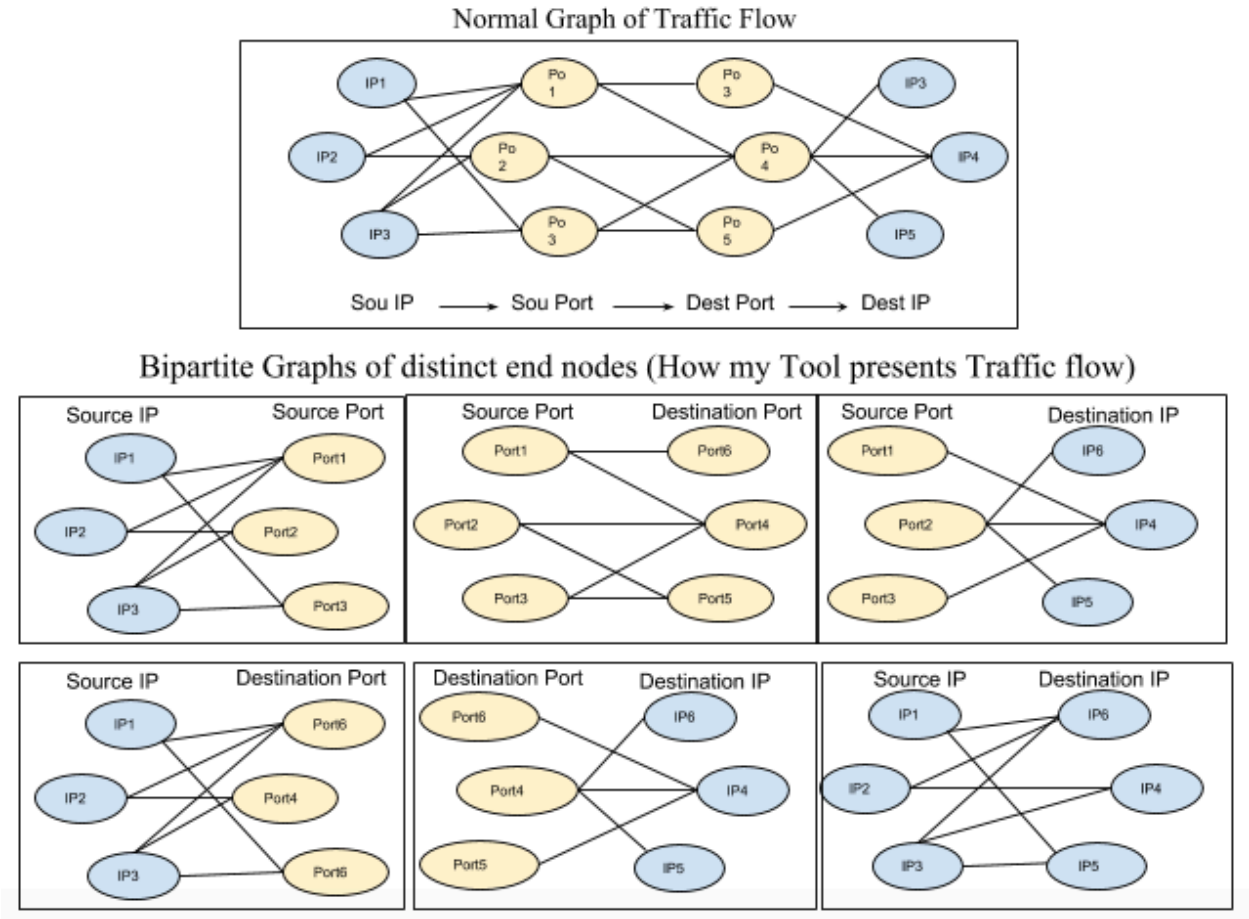


Figure 3D – IP/Port Traffic flow presented in six Bipartite Network Graphs (Six Ways of Bipartite (pair of two sided nodes) graph presentation of traffic flow options are available for visualization in the tool)

CHAPTER 4

4. FEATURES, FUNCTIONS AND OPERATION OF THE TOOL

As its name indicates, the 'University of Rhode Island NetFlow activity monitoring and analysis visualization tool' is a traffic Netflow monitoring and analysis tool developed particularly for the University of Rhode Island. But with some trivial change, it can also be used for other organizations that retain Netflow-Log that is structurally like URI's. It is a tool developed and implemented in a computing software environment called **RStudio** and its interactive approach and HTML/CSS supporting environment called **Shiny.** It is a scalable and an open-source tool. The tool can be accessible through – https://semhar.shinyapps.io/MainUI/, and user guide documentation of tool is accessible through – https://semhar.shinyapps.io/MainUI/_w_1e10b9d8/READMEinstruction.html or it can be found on the top of tool's web page. When it comes to limitation, the tool obviously has some limitations – when a multiple Netflow datasets (approximately about more than two million records) is uploaded, the generation of the bipartite graph would fail to generate corresponding graph. Which would be as one of the future enhancement of the tool. As an input file specification matter, the tool accepts as input of format CSV (Comma Separated Value) file of any standard Netflow Log (as of URI's – Traffic-Log or threat-Log). As it is described in detail, in the following sections of this chapter, the output results would be generated in three tabs – Plot Graph View, Traffic Network Graph View, and Logs Grid View within a single screen of a web page.

4.1. TOOL OVERVIEW

The University of Rhode Island NetFlow activity monitoring and analysis visualization tool is composed of a ***Navigation panel*** and ***Visualization panel*** as a whole. The Navigation panel (left side of tool) controls the general execution and generation of graphs and grid tables. Whereas the Visualization panel (right side of tool) presents graphs in three tabs of plot graph views, traffic network graph view and Logs grid view. The Plot Graph view tab presents a line graph of frequencies as well as relative frequencies with respect to a range of timestamp. The traffic Network graph view tab on the other hand presents the network graph in terms of IP/Port along with a Line graph of selected node's (IP/Port) activity over the selected period of time. The Logs grid view tab presents a grid view of the dataset according to selections made in the Navigation panel. The user interface as shown in Figure 1 is created in such manner that it is user friendly in that the tool has panels that are easy to understand and follow. In addition to this the tool has also incorporated short functionality notes that explain the features in the tool controls.

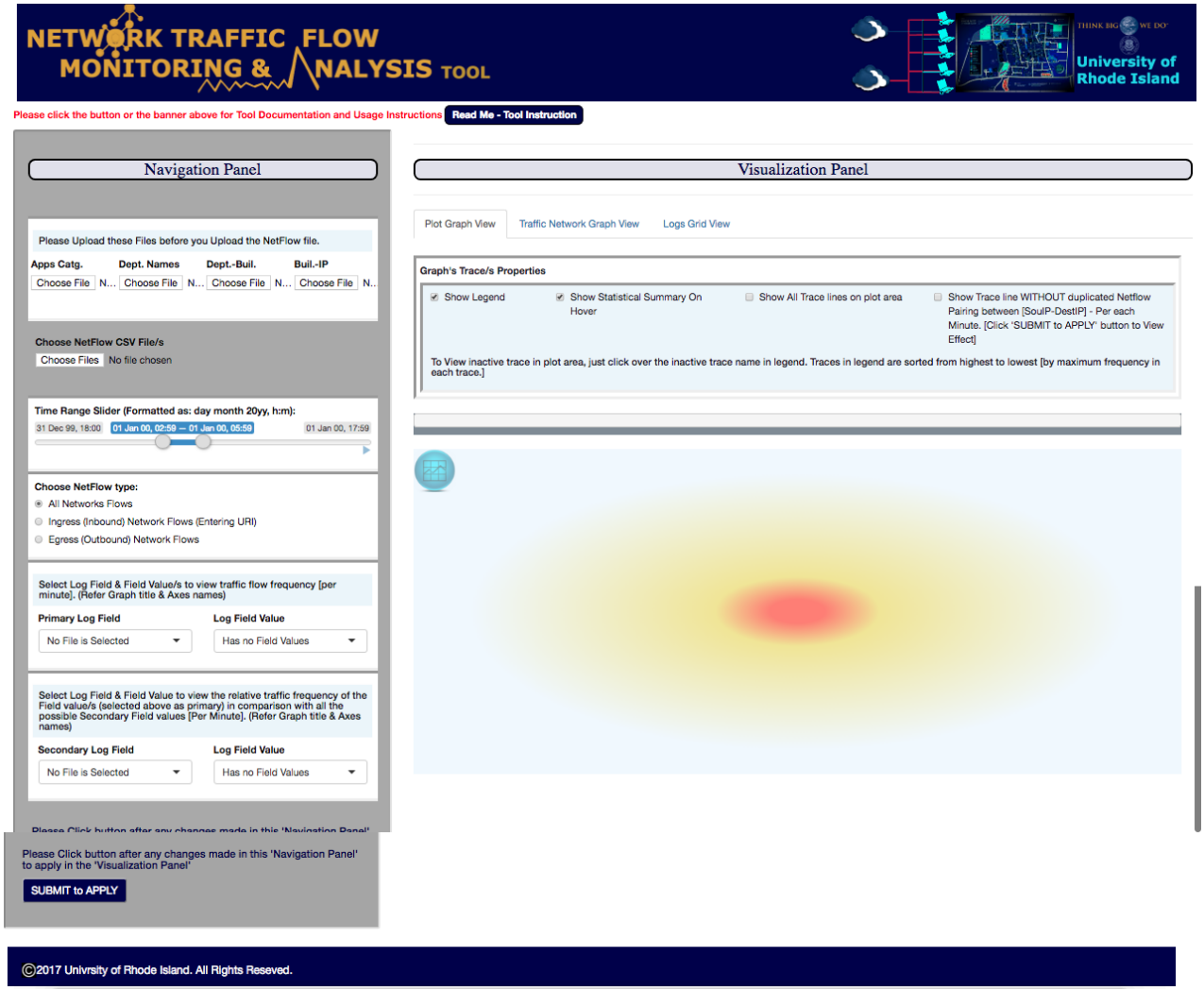


Figure 4A – The Blank Feature of the tool

4.2. NAVIGATION (CONTROL) PANEL

Before tool begins analysis, there are prior datasets (Table 4A) user needs to upload in advance in advance to uploading Netflow traffic. These datasets are part of the pre-processing of the Netflow data mentioned in Chapter 3. In Figure 4B.1; 'Apps Catg.' (shown as File-upload button) corresponds the dataset that is of applications' – category/subcategory/technology, 'Dept. Names' (shown as File-upload button)

corresponds the dataset that is of standardized URI department names (described in Section 3.2), 'Dept.-Buil.' (shown as File-upload button) corresponds the dataset that is of URI departments and their respective Building, and 'Buil.-IP' (shown as File-upload button) corresponds the dataset that is of URI buildings with the range of IP address they represent.

'Apps Catg.' Dataset

application	App_Category	App_Subcategory	Technology
100bao	general-internet	file-sharing	peer-to-peer
1c-enterprise	business-systems	erp-crm	client-server
1und1-mail	collaboration	email	browser-based

'Dept. Names' Dataset

iso_dept	Department
Adams House	Adams Residence Hall
Admissions	Admissions - Main Office
Alton Jones	W. Alton Jones Campus

'Dept.-Buil.' Dataset

Department	Building
Budget - Main Office	Adams House
Undergraduate Housing Assignments	Adams Residence Hall
Adams Residence Hall	Adams Residence Hall

'Buil.-IP' Dataset

Building	IP
Robert L. Carothers Library and Learning Commons	131.128.1.1
Robert L. Carothers Library and Learning Commons	131.128.1.126

Table 4A – Format of the Datasets (shown in Figure 4B.1 – 'Apps Catg.', 'Dept. Names', 'Dept.-Buil.', and 'Buil.-IP') to be uploaded prior to Netflow Dataset

The 'Navigation panel' of this NetFlow activity monitoring and analysis tool that is used in the initial control of the execution and generation of graphs and grid tables – embraces six sections (Figure 4B). The first section (Figure 4B - section 1) of the panel provides user with a button to upload NetFlow Log dataset/s (single/multiple) in CSV

(Comma Separated Values) format. This is the section that marks the first step in the usage of this tool. The second section (Figure 4B - Section 2) is a time slider that initially displays some default date to hold space. By the time file uploading completes, this section updates itself to the flow timestamp range available in the uploaded dataset. Updating process might take a while depending on the size of the uploaded data. So, an indicator of running processes (

) will be displayed at the top-right corner of the tools' page. The time slider in this section will enable users to specifically slide between time ranges that they preferred to focus on. As we move one step down, there are three radio buttons in Figure 4B - Section 3 of the Navigation panel that provide users with options to subset their data according to the source IP address. In Figure 4B - Section 4 of the Navigation panel there are two drop-down menus. The one on the left labeled as "Primary Log Field" contains all the possible Log fields of the uploaded dataset excluding IP, Port and other numerical Log fields. The drop-down menu labeled "Log Field value" to the right contains the field values associated with the field selected in the "Primary Log Field". Figure 4B - Section 4, in general helps user to select and navigate through possible Log fields on the uploaded NetFlow data and provides user the capability to filter Netflow log to the log field value/s of interest to be analyzed. Similar to Section 4, Section 5 of the Navigation panel contains two drop-down menus. The drop-down menu on the left side of this section is labeled "Secondary Log Field." The drop-down menu on the right side of this section is labeled "Log Field Values." The "Secondary Log Field" contains possible Log Fields from the uploaded NetFlow dataset with an exception of the Log field that is already selected in the "Primary Log Field". The data that is to be considered in the "Secondary Log Field" is the data resulted from

the selection (filter) made by user in the "Primary Log field" (if user chooses to make selection as in Figure 4B - section 4). Thus, it is significant to note that the Secondary Log Field is dependent on the selections made in the Primary Log Field and its values.

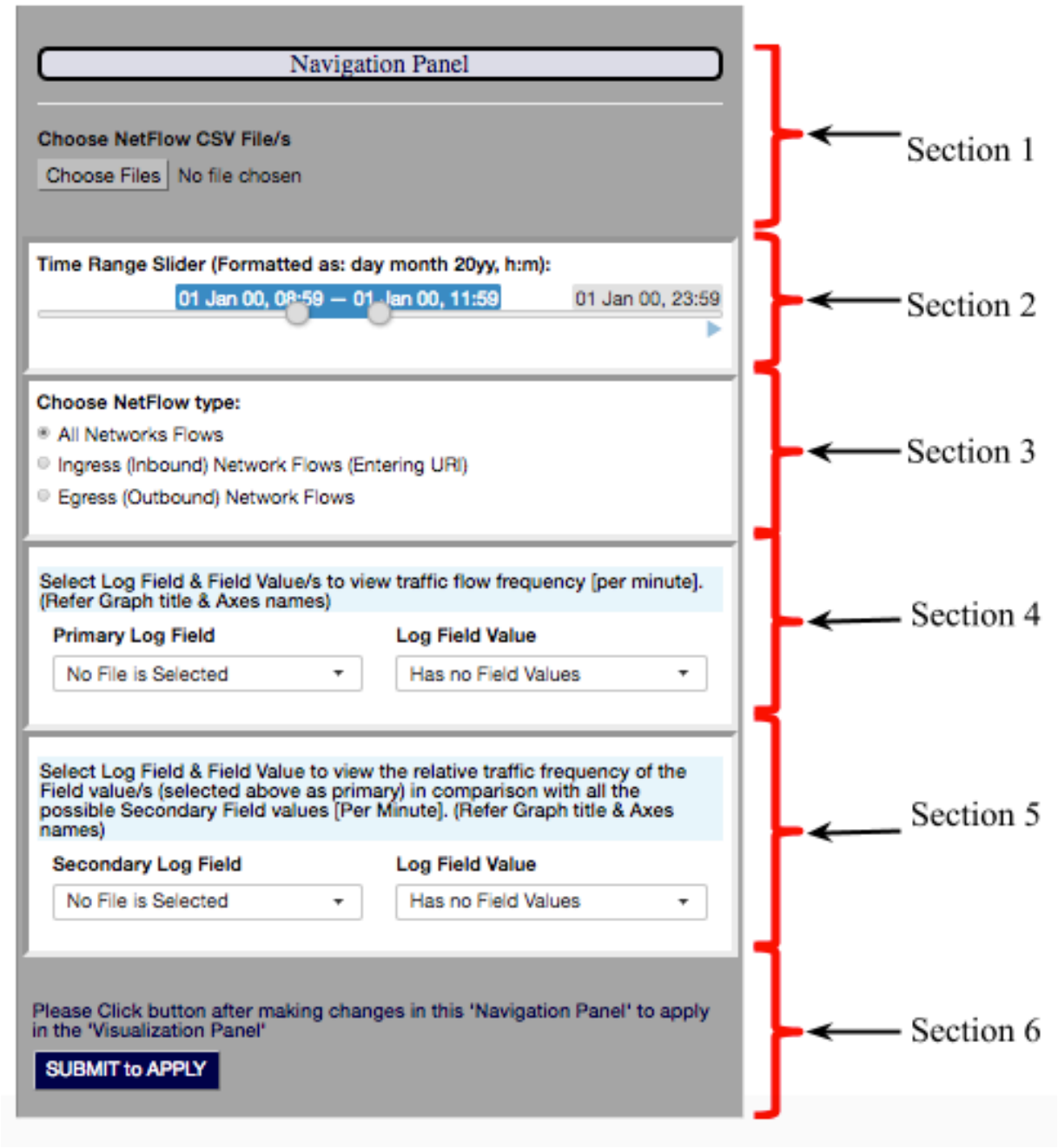


Figure 4B – Navigation panel of the tool

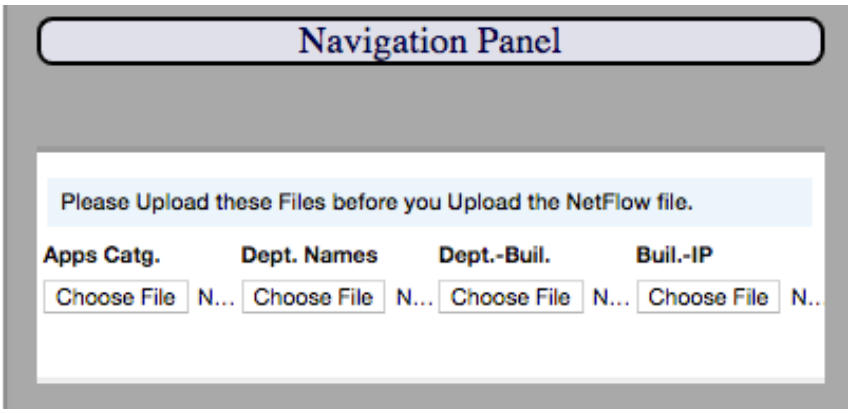


Figure 4B.1 – Datasets that Needs to be Uploaded in advance of the tool analysis

Once the user made all the preferred selections, it is necessary for users to submit or apply their selections to render results. And this can be executed in the button 

that is specified in Figure 4B - section 6 of the Navigation panel. Most execution of the tool takes some time to show results and as an indicator to user a process indicator (

) will be shown at the top-right corner of the page of the tool.

4.3. VISUALIZATION PANEL

The visualization panel is the section of the tool that has three tabs that display the resulted visualization. These tabs present NetFlow dataset in Plot graph view, Traffic network graph view and Logs grid view. These visualizations are sectioned in tabs named after the respective view they present which helps users to move from one tab to another. Each tab possesses individual control features to manipulate the view they depict. User can navigate from one form of graphic view to another form while still maintaining the different form of graphic presentations. Users will initially see Plot Graph View tab as a default view. All visualization tabs are explained in the following sections.

4.3.1. PLOT GRAPH VIEW

The Plot Graph View tab as indicated by its very name, presents a line graph of NetFlow activity in the form of Flow frequencies or relative frequencies (in percentile) with respect to unique timestamp after a submission is carried out in the Navigation panel. This graphic presentation will be displayed in Section 9 of Figure 4C. In Figure 4C, section 8 displays the verbatim to indicate users the time range of which the timestamp of the line graph covers.

The line graph presented in this figure will have feature properties like legend, trace/s and a display of statistical properties on hover. Those features can be controlled with the Checkboxes (☑) provided in Section 7.

- ☑ **Show Legend** - will control the appearance of Legend in the plot view.
- ☑ **Show Statistical Summary on Hover** - controls the appearance and disappearance of the on-hover statistical summary of each trace line in the line graph.
- ☑ **Show All Trace Lines on Plot Area** - this check box will be functional if there is a need to display multiple trace lines in the plot area. If plot rendered is composed of multiple traces, then by default 'mean line' of multiple traces will appear in the plot area where all traces of lines will be shown in legend inactively. To view each trace individually, user has to click on the inactive name that appears in the Legend. The tool is designed to color multiple traces with only ten distinct colors and fill opacity. By that, multiple trace names on Legend will appear in a rank of descending order depending on the maximum frequency or relative frequency each trace reflects and are colored and filled with an opacity accordingly. For instance, in a Legend of group ten or less, traces will have distinct colors and decreasing fill opacity as ranking value decreases. Whereas

in a group of more than 10 traces, color and fill opacity will repeat itself in every ten traces but ambiguity of coloring could be avoided by the name order of traces in Legend. Any predecessor trace name in the Legend has a maximum value greater or equal to its successor and will always appear behind its successor in the plot area.

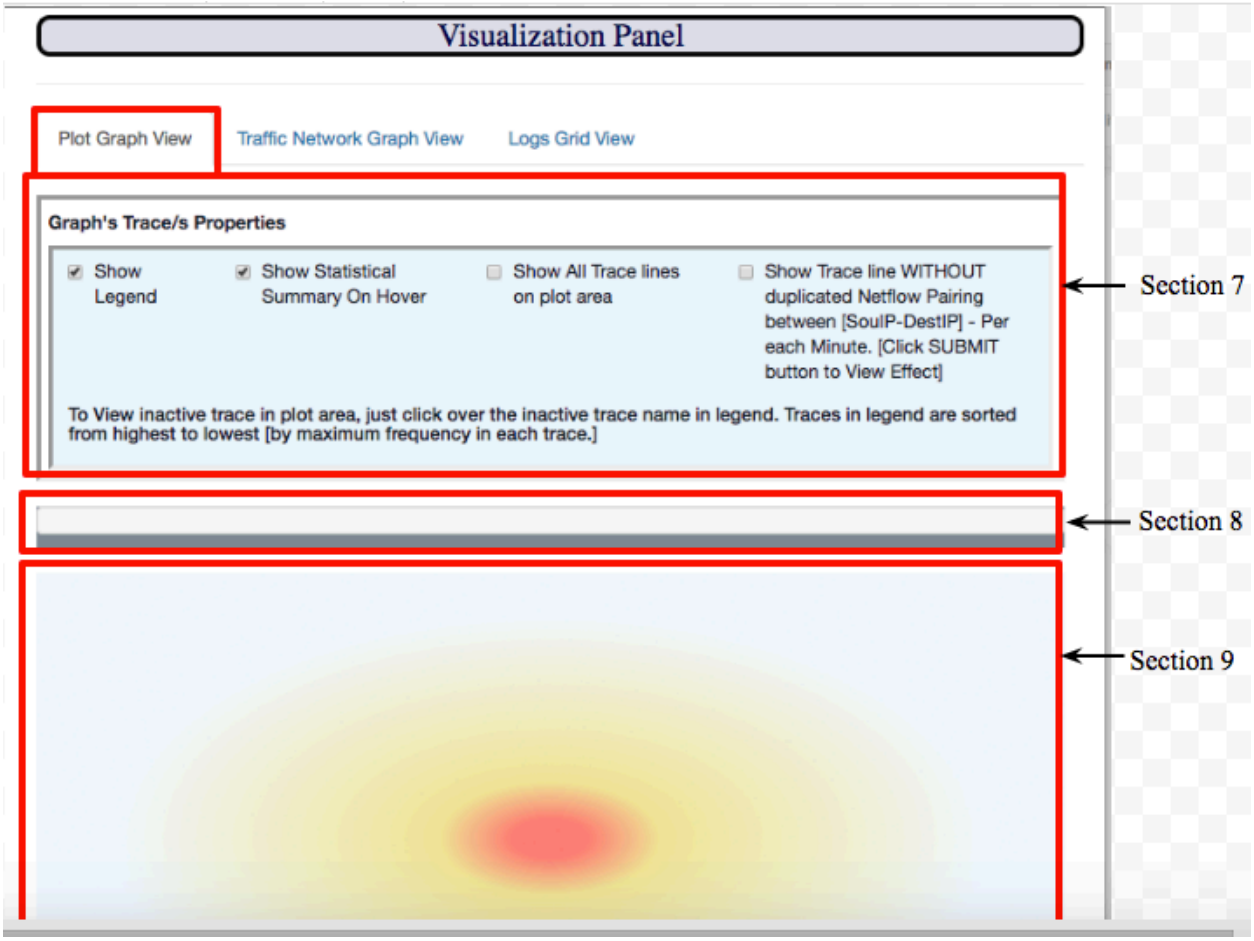


Figure 4C – Visualization panel of the tool

- ☑ **Show Trace Line WITHOUT duplicated** …. [Click Submit button ….] - as it indicates in the squared parentheses, in order for the effect of this check box to take place user needs to click the Submit button SUBMIT to APPLY located in the 'Navigation panel'.

4.3.2. TRAFFIC NETWORK GRAPH VIEW

Traffic Network Graph View is the second tab located next to the Plot Graph View. It generates a visual network graph of only IP↔Port combination in Figure 4D - Section 14. It also generates a line graph of the same format in Figure 4D - Section 15 as the graph rendered by the 'Plot Graph view' which in this case the trace in the plot area indicates the flow activity of IP address or Port number with respect to a timestamp. These two graphical presentations are controlled using the controlling features provided in Section 10 through 13 as shown in Figure 4D.

In section 10 of Figure 4D there are a set of radio-buttons () that are useful to manipulate the properties of the graphs in Section 14 and 15. Once data is submitted through the 'Navigation panel', this visualization tab will work with the IP/Ports fields of the subset of data. Therefore, features in Section 10 enable users to choose preferred IP⇔Port interaction, NetFlow type according to the source IP address and focus of node side (whether user wants to focus on Source or Destination of the Network graph to be rendered). The selection made in Section 10 will then determine the availability of node degree in the drop-down menu in Section 12 only when the update button 

in Section 11 is authorized (clicked). The 'apply' and 'draw network' button 

in Section 13 finally marks the end of the process as it enables user to generate and draw both Network and line graph in Section 14 and Section 15 respectively. This is better illustrated in Figure 4D.

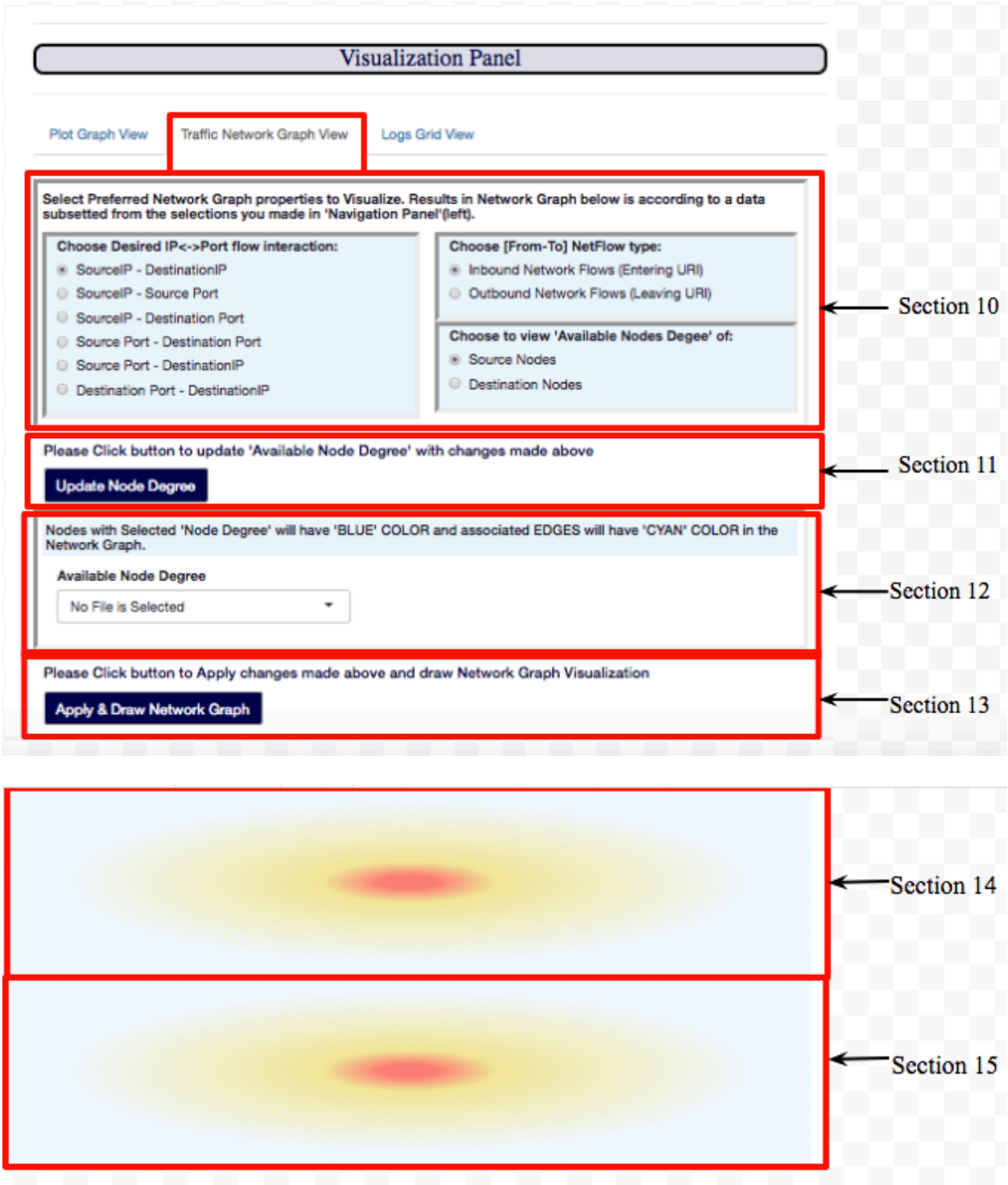


Figure 4D – Traffic Network Graph view of the tool

4.3.3. LOGS GRID VIEW

The Grid View tab renders table grid view of the NetFlow data (cleaned with the process mentioned in Chapter 3) subsetted according to the selection made in the navigation panel. Once authorization is given in the Navigation panel of the tool, Section 16 as illustrated in Figure 4E will bring up all the available Log Fields of the NetFlow dataset with an option of Checkboxes that can be accessed for further full view of table in Section 17. The image shown in Figure 4E is the view of the tab before any action is taken for data visualization.

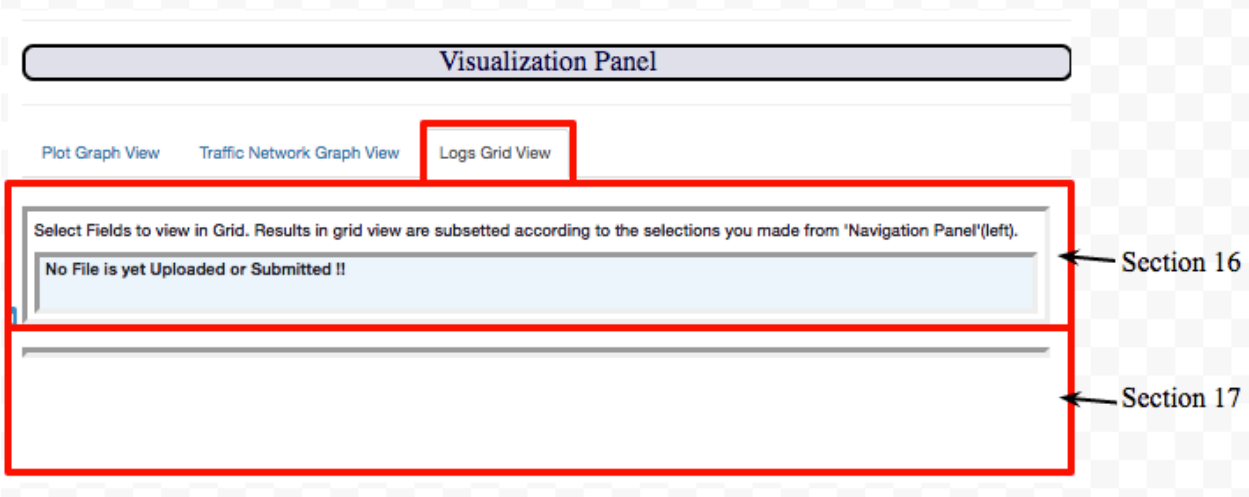


Figure 4E – Logs Grid View of the tool

4.4. OPERATION OF THE TOOL WITH DATA ENTRY

In this section of this chapter, instances of the tool with an entry of uploaded data will be illustrated. The demonstration of the tool with data entry will follow similar order procedure as sections 4.1, 4.2 and 4.3.

A NetFlow file is first uploaded and as observe in Figure 4F, the line graph rendered shows the timestamp along the x-axis and the frequency of flow along the y-axis. The graph shows the overall Network flow activity since no specific selections were made neither in 'Primary Log Field' nor 'Secondary Log Field' options of the Navigation panel.

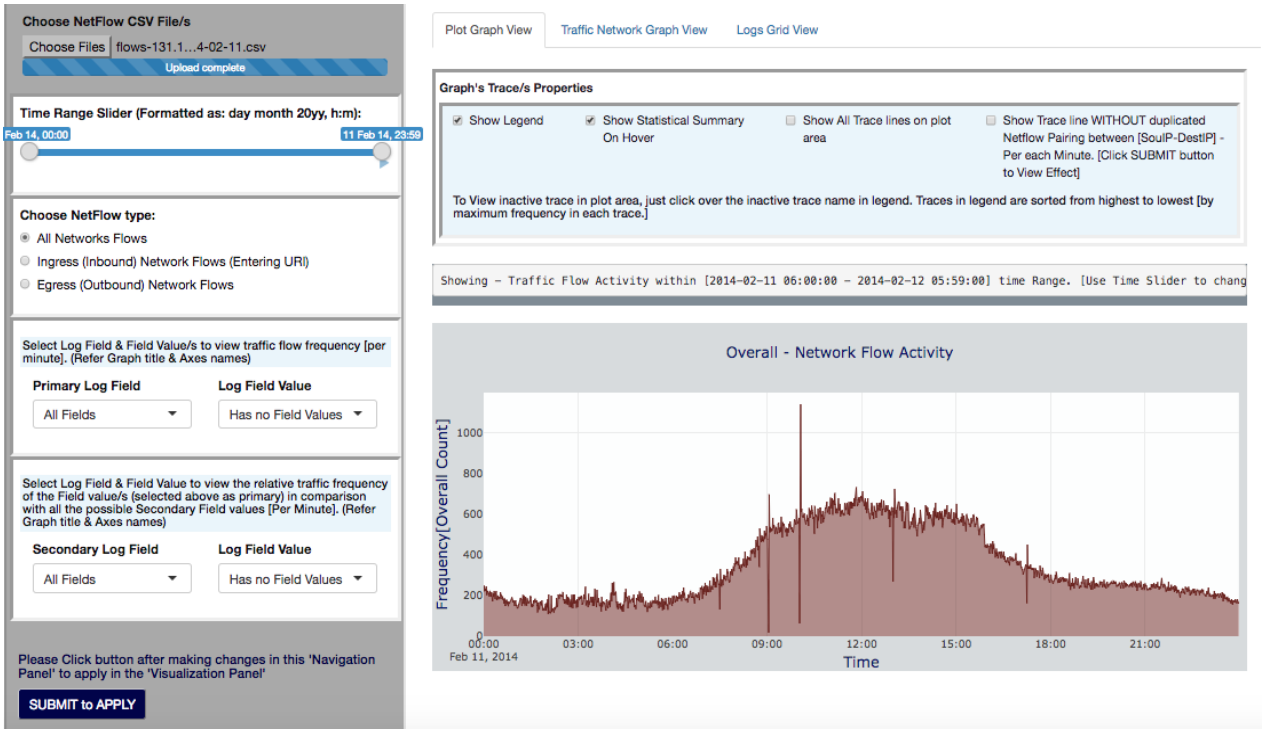


Figure 4F – Overall NetFlow activity as per the data uploaded and Selections made

4.4.1. NAVIGATION PANEL WITH DATA ENTRY

In Figure 4G, it can be observed that a Netflow data is uploaded and the time range slider is updated accordingly. And the drop-down menus are showing that they have Log Fields, but no specific Log field is chosen. In the following subsections we will observe the graphs that are generated as a result of the selections made in the Navigation panel.

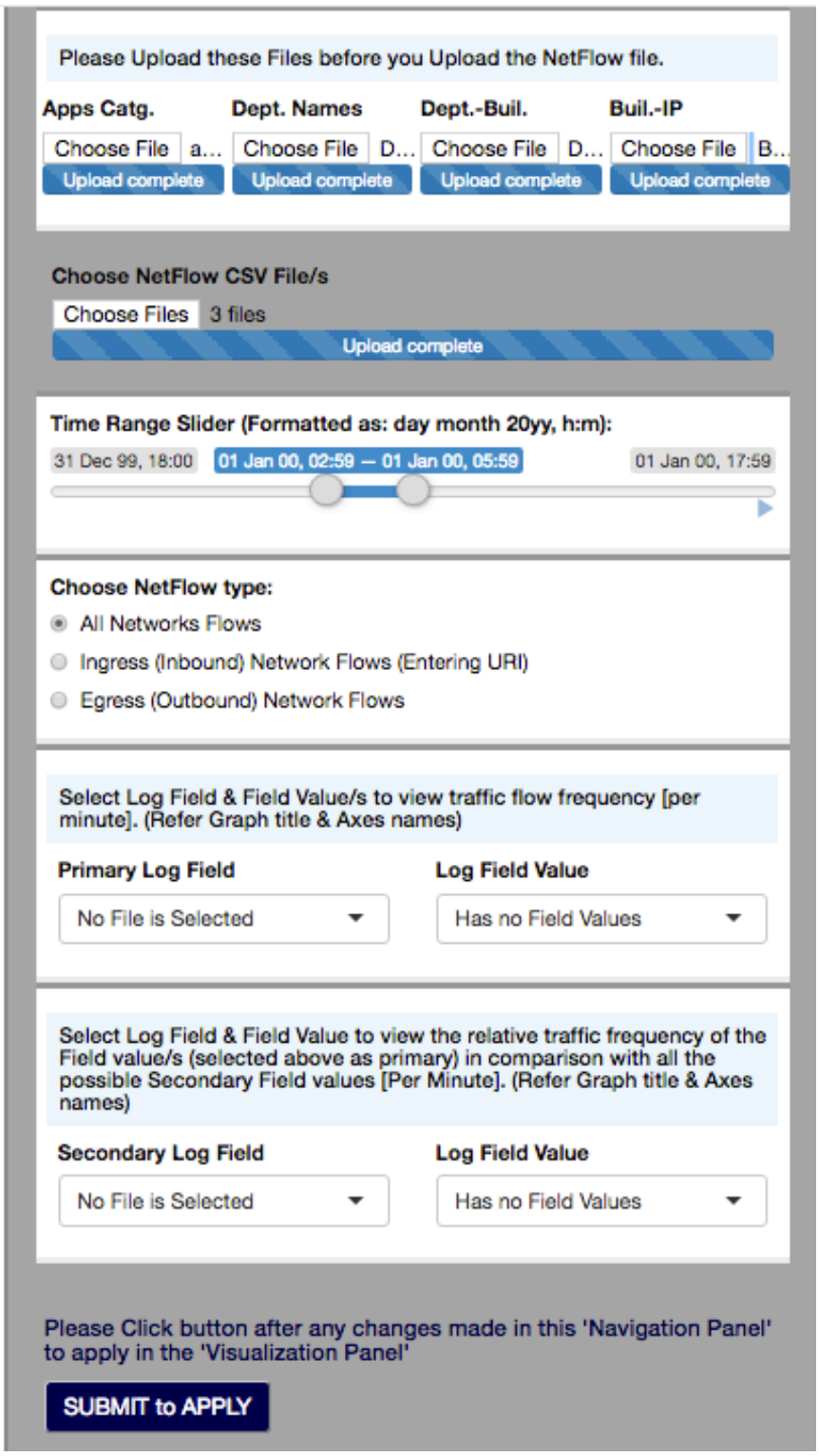


Figure 4G – Navigation panel with Data entry

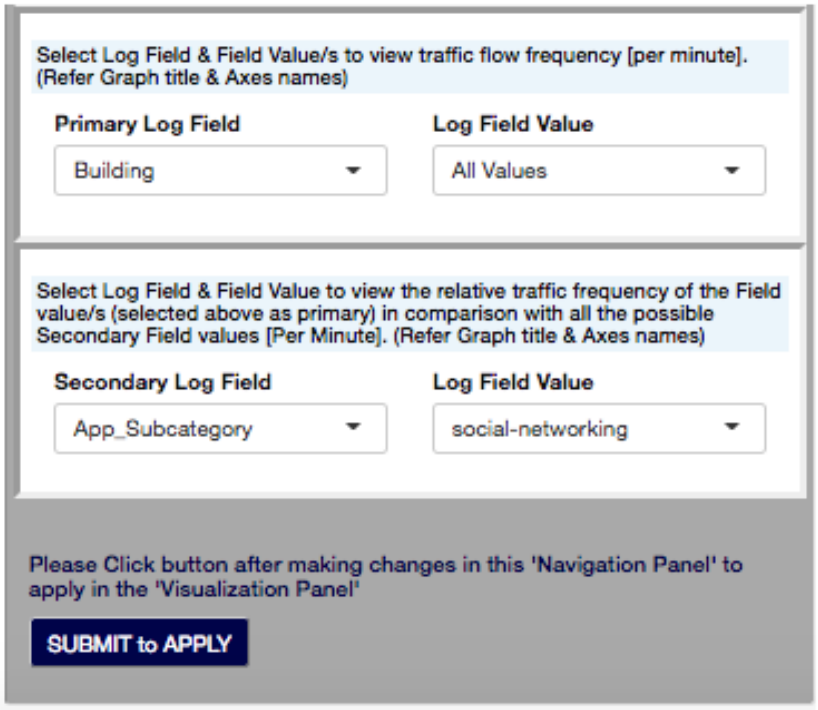


Figure 4G.1 – Navigation panel with Data entry - Log Fields Specification

4.4.2. PLOT GRAPH VIEW TAB WITH DATA ENTRY

Most commonly preferred type of graph when we intend to analyze a trend or trace pattern spread over a time period is Line Plot. Furthermore, it is also suitable to plot and compare when one needs to observe relative changes in quantities across some variable. In this paper, the tool is intended to guide users to identify patterns and compare outcomes along given timestamp through Line plot. Not only will user witness the activity flow frequency through time, but also observe the relative frequency. This implies determination of the distributional correlation of one Log Field value to another Log Field Value. For instance, this tool provides a comparative relative frequency if user demands to have an idea of the usage of "social-networking" among all URI Buildings. All needed to be done is, select '*Building*' from '**Primary Log Field**' drop-down menu

and select '*All Values*' from '**Log Field Value**' (drop-down menu to its right) then select '*App_Subcategory*' from '**Secondary Log Field**' and '*social-networking*' from '**Log Field Value**' (drop-down menu to its right) (Illustrated in Figure 4G.1 and Figure 4H.1). This will generate the relative frequency of the usage of 'social-networking' compared to other '*App Subcategories'* by each URI buildings over a period of time. From this comparative perspective of various Fields, a supportive analysis result could be derived that might have been missed by only targeting IP/Port/Bytes/Packets level analysis.

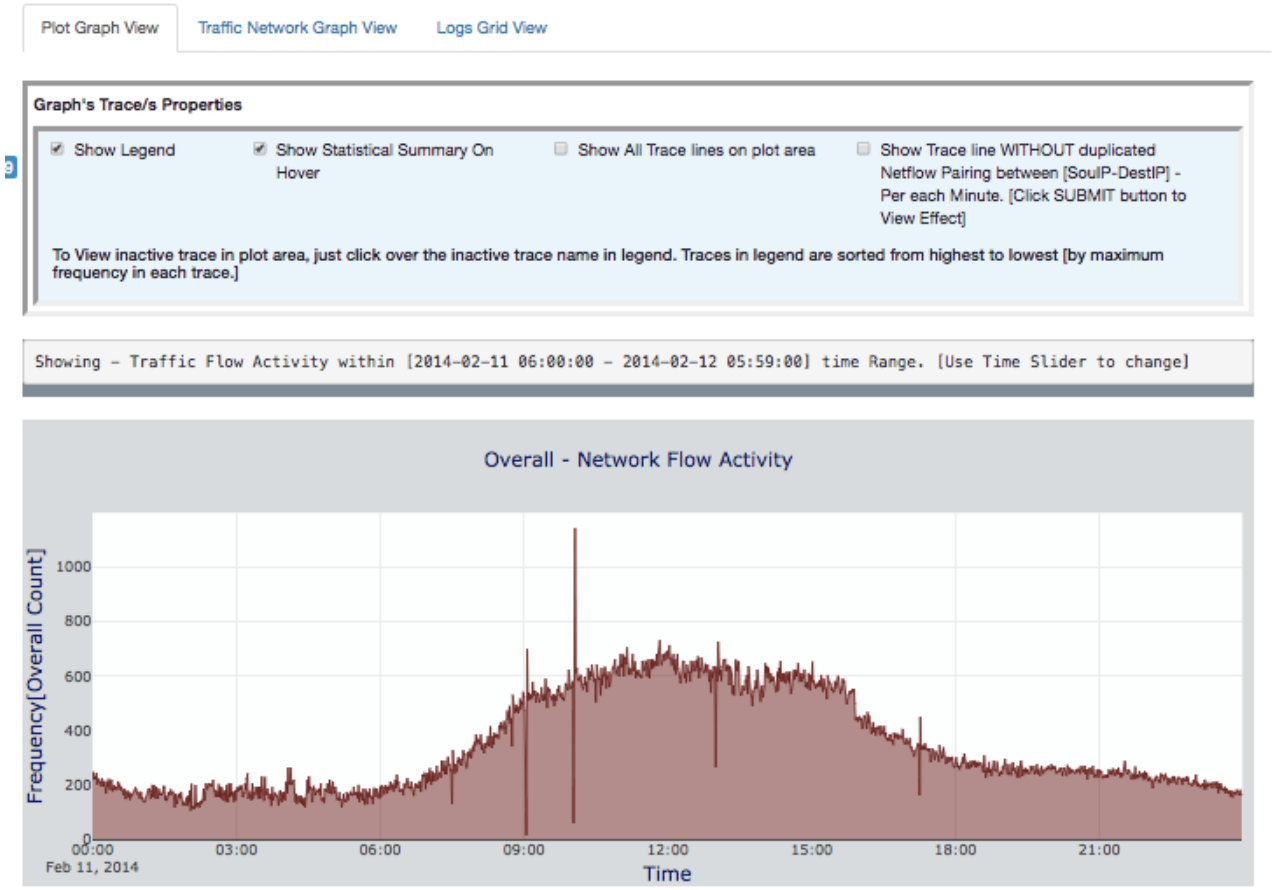


Figure 4H – Plot Graph View with Data entry displaying Flow frequency

The plot graph view tab displays an 'Overall Network Flow Activity' in Figure 4H and all buildings' in Figure 4H.1 of the uploaded data and selections made through the

Navigation panel indicated in Figures 7 and 7.1 respectively. The graph driven in Figure 4G shows an instance of a normal flow frequency along a range period of time.

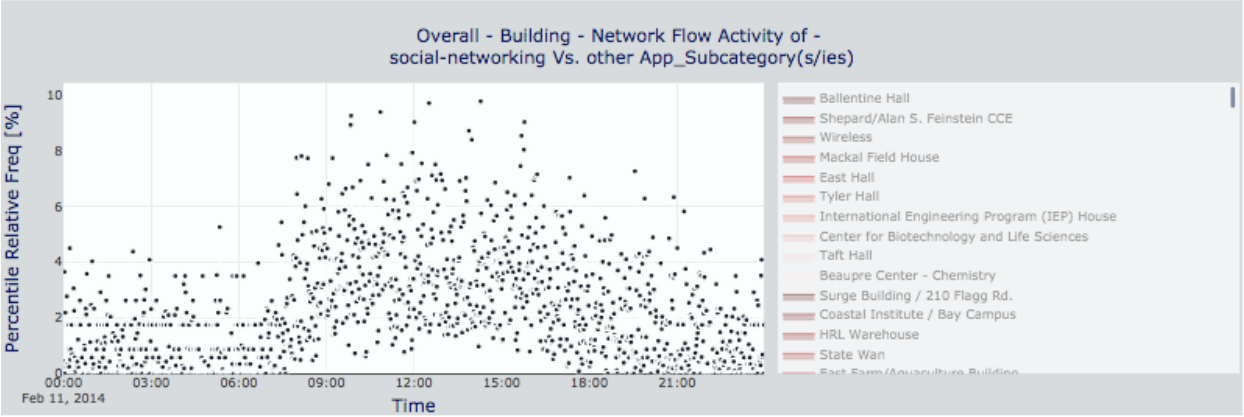


Figure 4H.1 – Plot Graph View with Data entry displaying relative frequency (%) of URI buildings on the usage of Social-networking in contrast to other Application-categories

4.4.3. TRAFFIC NETWORK GRAPH VIEW TAB DATA ENTRY

After uploaded data is submitted for visualization in the Navigation panel, the Traffic Network Graph view renders two types of graphs. As shown in Figure 4I this takes place only when a user updates the node degree with 

and selects the preferred degree to be visualized and authorizes to draw graphs. Preferably, user can click on inactive Legend names to view activity of each trace shown in the Legend of the graph. As it is shown in the title of the layout of the Network Graph image (top image of Figure 4I), the Network Graph is demonstrating the flow direction from of Source IP to Destination IP with source nodes to destination nodes ration of 5 to 413. This shows the interaction of low number of source hosts to high number of destination hosts. From this result, one can infer that there could be an involvement of Port Scan attack. This can

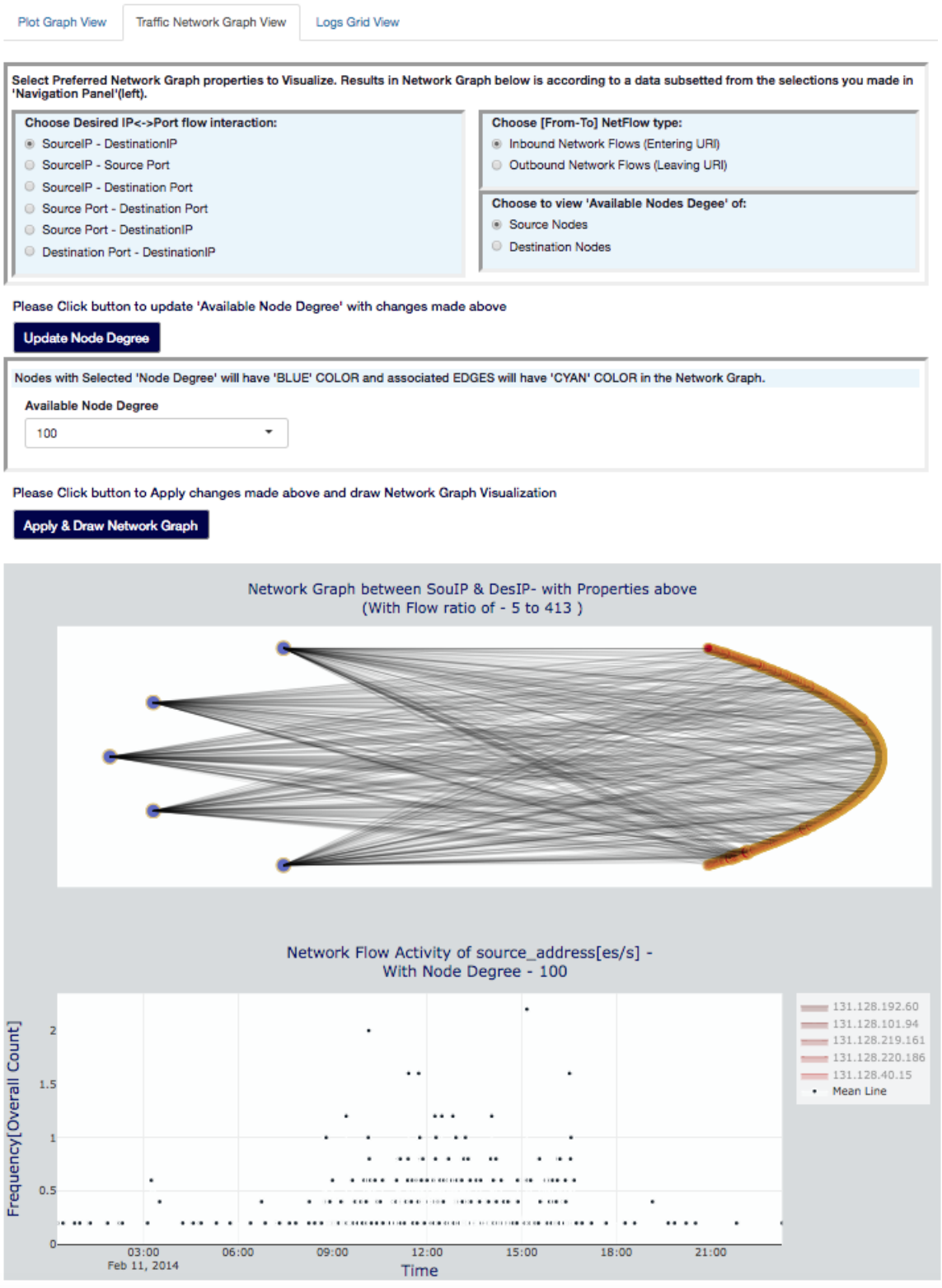

Plot Graph View
Traffic Network Graph View
Logs Grid View
Select Preferred Network Graph properties to Visualize. Results in Network Graph below is according to a data subsetted from the selections you made in 'Navigation Panel'(left).
Choose Desired IP<->Port flow interaction:
SourceIP - DestinationIP
SourceIP - Source Port
SourceIP - Destination Port
Source Port - Destination Port
Source Port - DestinationIP
Destination Port - DestinationIP
Choose [From-To] NetFlow type:
Inbound Network Flows (Entering URI)
Outbound Network Flows (Leaving URI)
Choose to view 'Available Nodes Degee' of:
Source Nodes
Destination Nodes
Please Click button to update 'Available Node Degree' with changes made above
Update Node Degree
Nodes with Selected 'Node Degree' will have 'BLUE' COLOR and associated EDGES will have 'CYAN' COLOR in the Network Graph.
Available Node Degree
100
Please Click button to Apply changes made above and draw Network Graph Visualization
Apply & Draw Network Graph
Network Graph between SouIP & DesIP- with Properties above
(With Flow ratio of - 5 to 413)
Network Flow Activity of source_address[es/s] -
With Node Degree - 100
Frequency[Overall Count]
0
0.5
1
1.5
2
03:00
Feb 11, 2014
06:00
09:00
12:00
15:00
18:00
21:00
Time
131.128.192.60
131.128.101.94
131.128.219.161
131.128.220.186
131.128.40.15
Mean Line

Figure 4I – Traffic Network Graph View Tab Data Entry - Network Graph (Top) and Line Plot Graph (Bottom)

further be inspected to see if the flow is distributed and consistent and the node's anonymity.

4.4.4. LOGS GRID VIEW TAB WITH DATA ENTRY

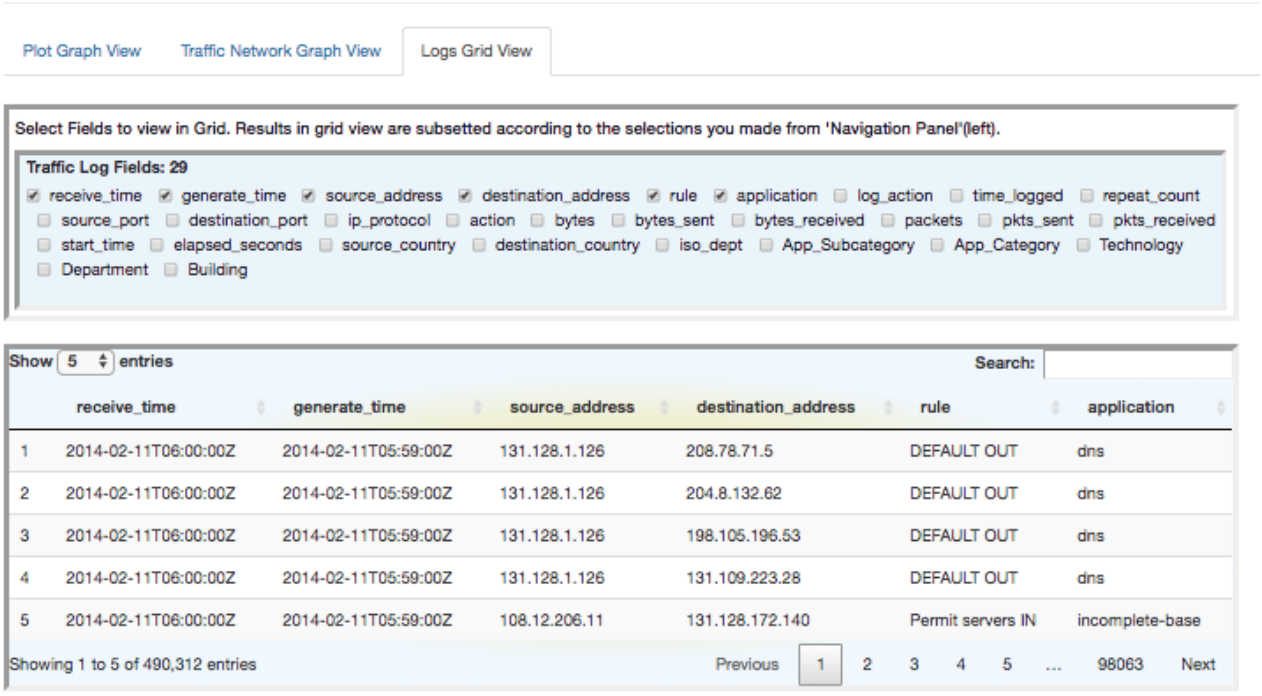


Figure 4J – Logs Grid View Tab with Data Entry

With its functionality features, this tab of the tool provides users to preferably include and exclude NetFlow Log Field/s from a table grid view by Checking and Unchecking the Checkboxes that represents Log Fields (shown in Figure 4J). Apart from this, user can control entries per page with an option that helps specify entries (

) located at the top-left corner of the grid table and search box option to

look for a particular value. It gives a user an option of status bar with a result of total data entries (Showing 1 to 5 of 490,312 entries) (left-bottom) and numeric buttons (Previous 1 2 3 4 5 ... 98063 Next) to navigation through data pages of grid view.

4.5. SAMPLE ANALYSIS AND DISCUSSION

The tool generates a graphic illustration of a NetFlow activity across time. It presents this graphical representation in different forms that are accessible to analysts. Some compelling results were generated when logs of 5 days from 2014 were uploaded for analysis. The tool presents the data flow both in plot graphs and network graph visualization. As such giving analysts the opportunity to look at the NetFlow activity and see the distribution that indicates anomalies or activities that are off grid.

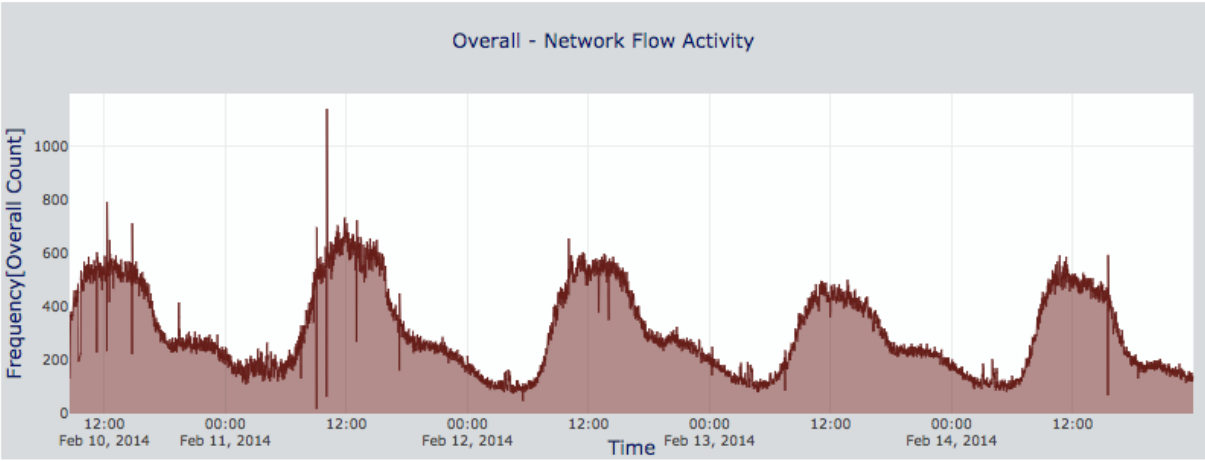


Figure 4K – Network Flow activity in weekdays (Sample days taken on Feb 10 - 14, 2014)

In Figure 4K, the graph presents a Network activity of five days of the week starting from the 10th of February through 14th of February in 2014. As illustrated in the

graph, a peak activity can be noticed on the 11th of February at exactly 10:04. Hence if analyzers want to check on to the nodes involved in this activity, they can first zoom-in to that particular time with suspicious activity happened. To do this, users have to use the time slider located in the Navigation panel of the tool and then submit selection for such effects in Figure 4K to take place. Users then have to switch to 'Traffic Network Graph View' tab to see and carefully observe the nodes.

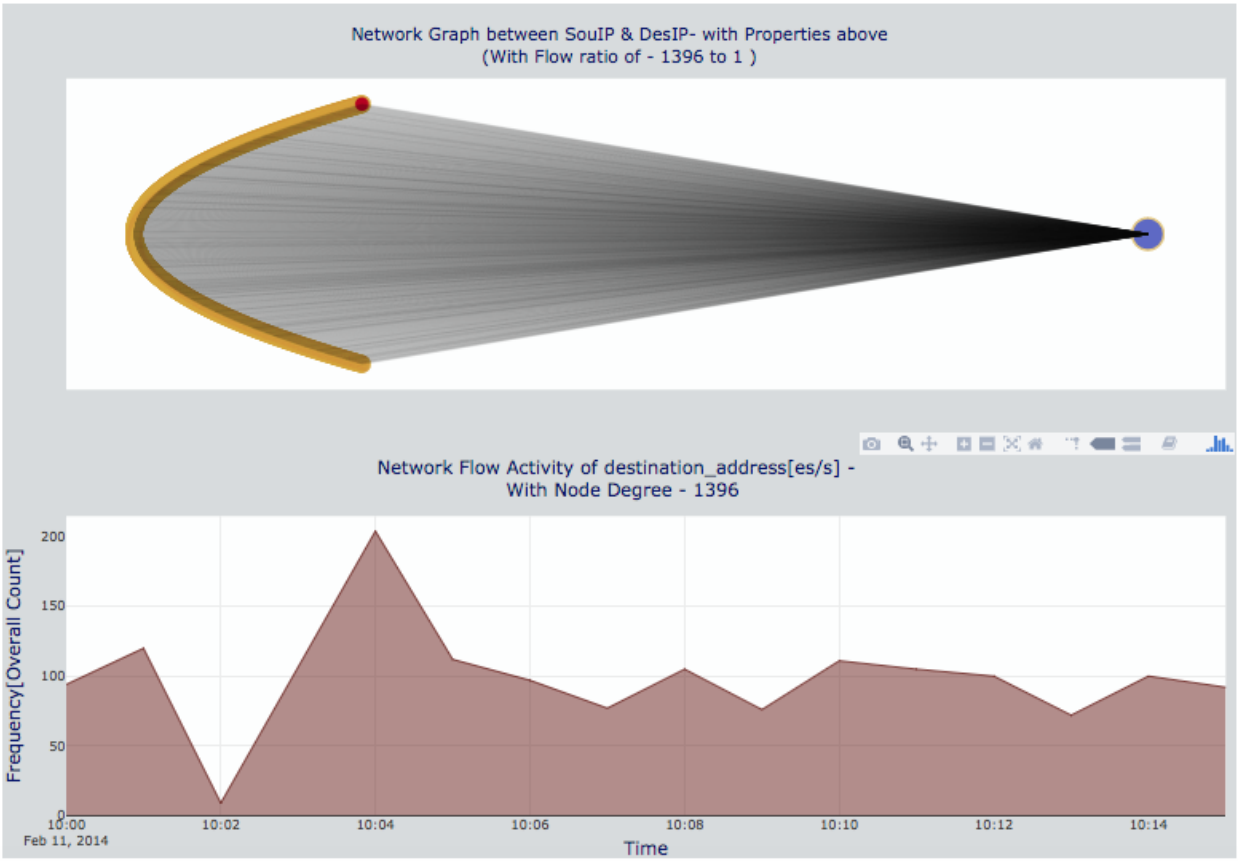


Figure 4L – Traffic Network Graph View of Time Range 10:00 - 10:15 of Feb 11, 2014 (with Node (host) communication degree 1396)

What is demonstrated in Figure 4L is the Traffic Network graph view of the nodes that were obtained when a suspicious time range was selected for further analysis. On real time use of the tool, we could be able to see the node colored in blue

(131.128.196.75) communicated by multiple nodes. In Figure 4L, it is clearly identified that 1396 nodes (none URI IP addresses) are interacting with just one node (URI IP address). It is also significantly important to notice that the communication took place in less than fifteen minutes. As such, analyzers could physically take action according to what is communicated as well as identity of the URI host which is involved in hosting a swarm of incoming request shown in the graphic representation of the NetFlow.

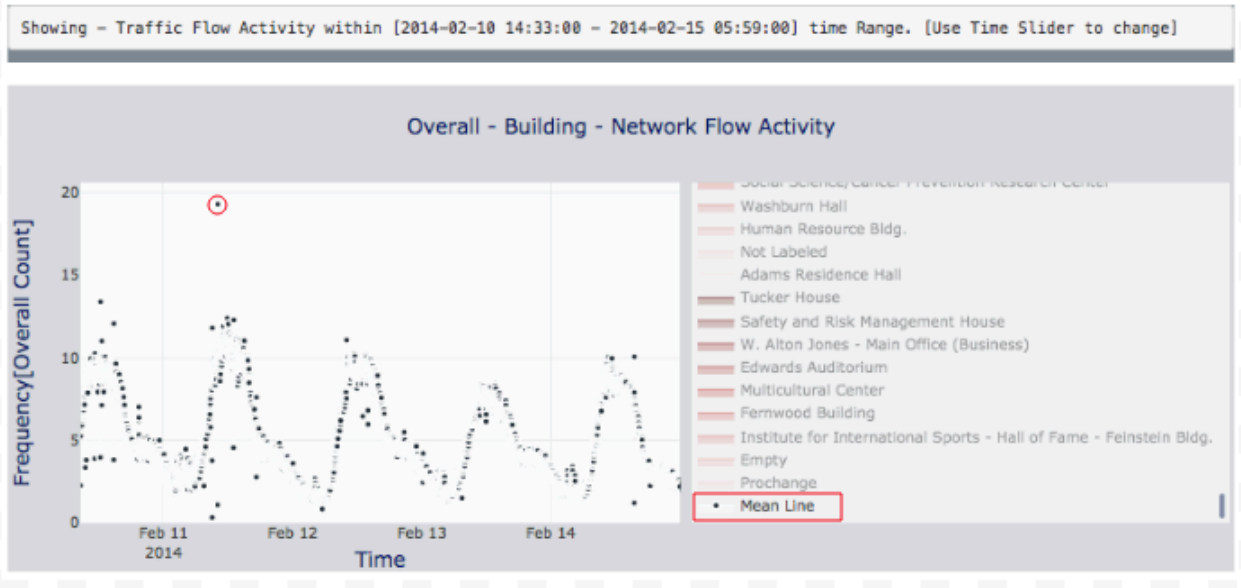


Figure 4M – Mean Line of All URI Buildings' NetFlow Activity

As described in section 3.3.1 of chapter 3, Figure 4M shows a mean trace line obtained from NetFlow captured at each timestamp by every Building at URI. The NetFlow appears steady throughout the week with slight drop-down as weekdays approach weekend. In general, except for that mean flow frequency indicated in a red circle in the plot area, all peaks have crests of similar level. The peaks in the NetFlow graph, as seen in the plot graph of Figure 4M take place in the hours between 8:00am thru 5:00pm, where a high flow is expected. But the flow outlaid at 10:04 AM on February 11

circled red in Figure 4M is abnormal when compared to the other peaks. In this scenario, the tool has granularly traced, from the general NetFlow, a frequency that appears aberrant. Hence, the tool can help analysts to look into the Netflow of each building for

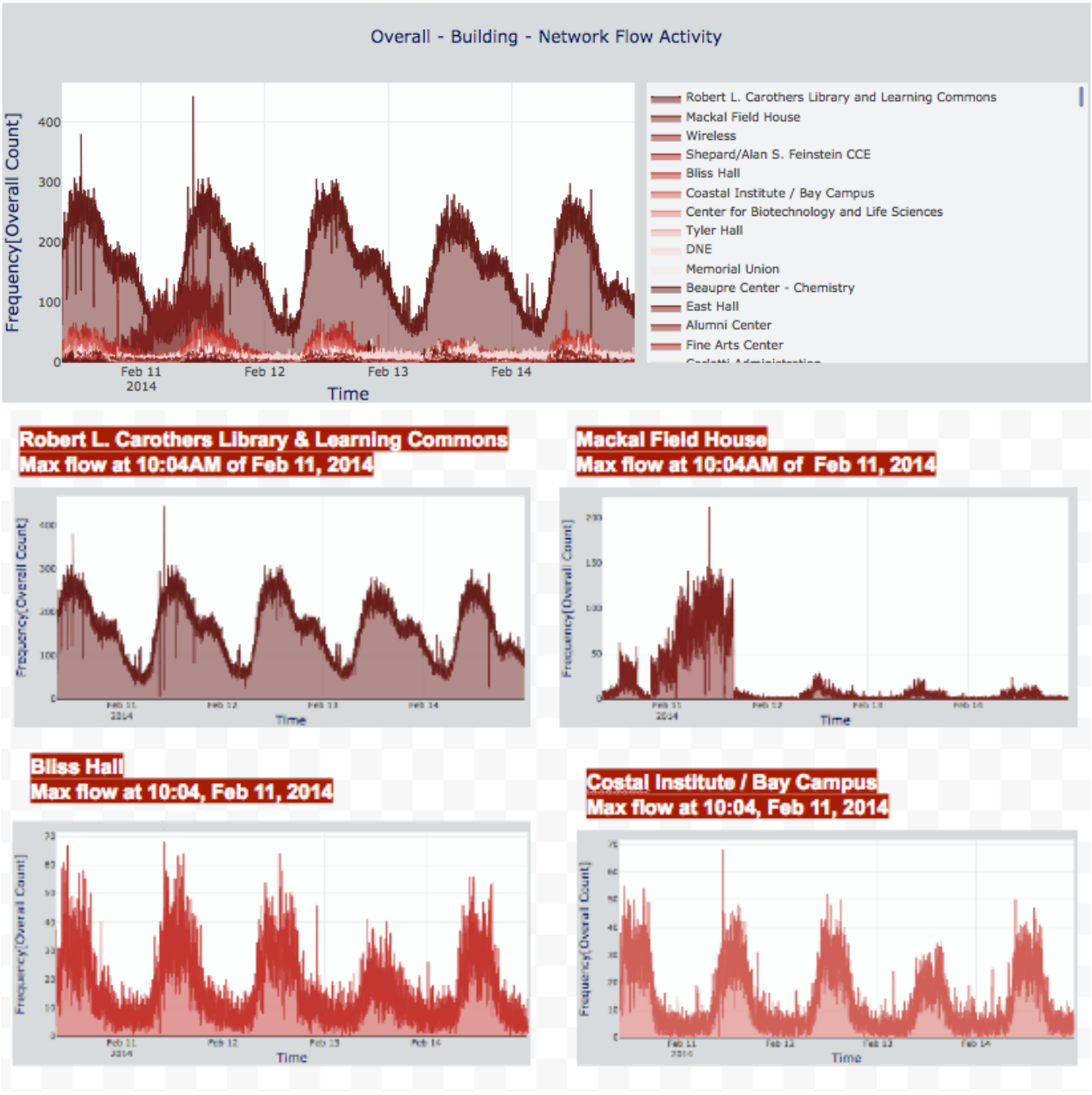


Figure 4N – Results of URI Buildings NetFlow Activities recorded from Feb 10 - Feb 14 of 2014

further investigation. Here, the tool not only traces at the buildings that are involved but also other URI Log fields such as applications, App_Subcategory [32], App_Category [32], Technology [32] and others as seen in Figure 4P below.

If we look at each building's individual flow and observe whether the flow is coincidental, deliberate or targets one particular building, then the result shows that the target URI building is university's library (Robert L. Carothers Library and Learning Commons) and few others with slight input. As shown in Figure 4N, after turning on the option of the tool to 'Show All Trace lines in plot area' the figure reveals the reality on ground. The findings of the result are shown in Figure 4N with traces of four top performing buildings taken separately.

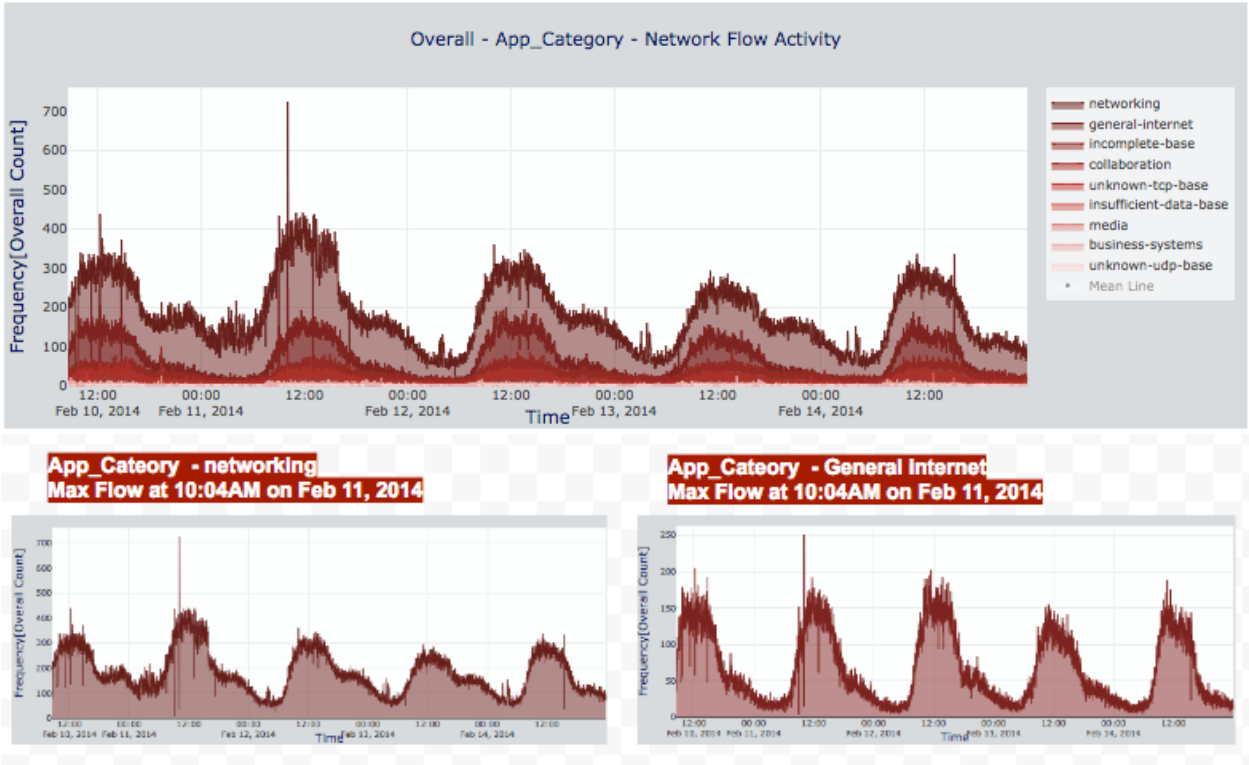


Figure 4P – Results of URI NetFlow Activities from App_Category point of view (Feb 10 - Feb 14 of 2014)

CHAPTER 5

5. CONCLUSION

In their book titled *Managing Cyber Threats: Issues, approaches, and challenges*, Kumar et al. assert that data mining is key in managing and identifying cyber-attacks. It is based on the data that is already collected that one can trace the pattern in the data and look for possible threats through the careful observation of the anomaly behaviors that can be seen on a pattern. Based on this assumption, it could be stated that one should carefully observe a data set for clearer understanding of what is happening in the web. We humans have, by nature, the capacity to process data through maps and plots. And a picture is truly worth a thousand words. The tool, as described above, is used in presenting cohesive images of activity of NetFlows that contains large amount of data to security analysts. Network traffic is complicated and requires high vigilance and involvement of multiple security incident detection tools. Just by observing the Network flow, Security analysts can gain a deeper understanding of the flow that can provide a picture of what is happening in the network. In the tool that is provided here, security analysts can zoom-in from high-level profile of network to more detailed views of flow. Thereby giving them the ultimate dissected look at the NetFlow activity that is taking place in the network. The tool is much easier to use as users are not forced to pull down menus and deal with multiple windows or dialogue boxes that are not user friendly. The tool contains all the controls and semantic levels integrated at once. This makes it much easier for the users to

access the information regarding network activity. The tool ultimately works by including all the possible connections and values involved across a span of time and frequency.

The design of the tool accounts for the consideration of many Log fields to enhance the flexibility of analyzing Netflow at URI. Instead of jumping into the bigger picture of analyzing Netflow by IP/Port, the goal of this tool is to first aggregate logs by Fields and then perform IP/Port analysis. The analysis of IP/Port in this tool is different from other analysis because it presents IP/Port nodes with their degree of communication in a network graph manner. This makes it easier for analysts as it allows them to retain perception of threats by prevalence and to sift their IP/Port by their communication degree. It is also significant to note that we don't have a ground truth at this point which makes it a little harder to conclude as to which behavior is to be considered as pure threat or otherwise. So, the future works in this field would compare the results of this tool and results obtained from analyzing methods currently at hand at URI. This would be helpful in the drawing some baselines (benchmarks) for report generation of further analysis.

LIST OF ACRONYMS

C&C - Command and Control

CSV - Comma Separated Values

DDoS - Distributed Denial of Service

DoS - Denial of Service

ICMP - Internet Control Message Protocol

IDS - Intrusion Detection Systems

IOT - Internet of Things

IP - Internet Protocol

IPFIX - IP Flow Information Export

IPS - Intrusion Prevention System

ISPs - Internet service providers

IT - Information Technology

NBA - Network behavior analysis

NIDS - Network Intrusion Detection System

NIPS - Network Intrusion Prevention System

P2P - Peer-to-Peer

POPs - Point of Presence

SSL - Secure Sockets Layer

TCP - Transfer Control Protocol

TNV - Time-based Network Traffic Visualization

UDP - User Datagram Protocol

BIBLIOGRAPHY

[1] "NetFlow Basics and Deployment Strategies." *Solar winds.com.* Solar winds, 2012. http://www.solarwinds.com/documentation/NetFlow/docs/NetFlowBasicsandDeploymentStrategies.pdf

[2] "PAN-OS® Administrator's Guide." *Syslog Field Descriptions*, Palo Alto Networks, Inc, Oct.2017,www.paloaltonetworks.com/documentation/61/pan-os/pan-os/reports-and-logging/syslog-field-descriptions.html. Date Accessed 2017. https://www.paloaltonetworks.com/content/dam/pan/en_US/assets/pdf/framemaker/61/pan-os/pan-os/section_5.pdf

[3] "US-CERT Security Trends Report: 2012 in Retrospect." *Homeland Security*. https://www.us-cert.gov/sites/default/files/US-CERT_2012_Trends-In_Retrospect.pdf, Feb 2013.

[4] Akoglu, Leman, Hanghang Tong, and Danai Koutra. "Graph based anomaly detection and description: a survey." *Data Mining and Knowledge Discovery* 29.3 (2015): 626-688.

[5] Ball, Robert, Glenn A. Fink, and Chris North. "Home-centric visualization of network traffic for security administration." *Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security*. ACM, 2004.

[6] Barford, Paul, and David Plonka. "Characteristics of network traffic flow anomalies." *Proceedings of the 1st ACM SIGCOMM Workshop on Internet Measurement*. ACM, 2001.

[7] Berthier, Robin, et al. "Nfsight: netflow-based network awareness tool." *Proceedings of LISA'10: 24th Large Installation System Administration Conference*. 2010.

[8] Bhuyan, Monowar H., Dhruba Kumar Bhattacharyya, and Jugal K. Kalita. "Network anomaly detection: methods, systems and tools." *Ieee communications surveys & tutorials* 16.1 (2014): 303-336.

[9] Bianca I. Colón-Rosado, et al. "Techniques for Anomaly Detection in Network Flows."University of Puerto Rico, 2015.

[10] Chae, Younghun, "Representing Statistical Network-Based Anomaly Detection by Using Trust" (2017). Open Access Dissertations. Paper 557.

[11] Cheng, En, et al. "Network-based anomaly detection using an elman network." *Networking and Mobile Computing*. Springer, Berlin, Heidelberg, 2005. 471-480.

[12] Collins, Michael. *Network Security Through Data Analysis: Building Situational Awareness.* 1st Edition. OReilly Media. Inc, 2014.

[13] Coull, Scott E., et al. "On Web Browsing Privacy in Anonymized NetFlows." USENIX Security. 2007.

[14] Covington, Michael J., and Rush Carskadden. "Threat implications of the internet of things." *Cyber Conflict (CyCon), 2013 5th International Conference on*. IEEE, 2013.

[15] Directory for Buildings & Departments, The University of Rhode Island. http://map.uri.edu/# . Accessed date Oct 20, 2017.

[16] Dua Sumeet et al. *Data Mining and Machine Learning in Cybersecurity.* Taylor and Francis Group. LLC, 2011.

[17] E.D. Kolaczyk, "Statistical Analysis of Network Data: Methods and Models," Springer, New York, 2009.

[18] Estevez-Tapiador, Juan M., Pedro Garcia-Teodoro, and Jesus E. Diaz-Verdejo. "Anomaly detection methods in wired networks: a survey and taxonomy." *Computer Communications* 27.16 (2004): 1569-1584.

[19] Francois, Jerome, et al. "Botcloud: Detecting botnets using mapreduce." *Information Forensics and Security (WIFS), 2011 IEEE International Workshop on*. IEEE, 2011.

[20] Goodall, John R., et al. "Preserving the big picture: Visual network traffic analysis with tnv." *Visualization for Computer Security, 2005. (VizSEC 05). IEEE Workshop on*. IEEE, 2005.

[21] Hu, Jiankun, et al. "A simple and efficient hidden Markov model scheme for host-based anomaly intrusion detection." *IEEE network* 23.1 (2009): 42-47.

[22] Koike, Hideki, and Kazuhiro Ohno. "SnortView: visualization system of snort logs." *Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security*. ACM, 2004.

[23] Kumar, Duhurba et al. Network Anomaly Detection A Machine learning Perspective. CRC Group, 2014.

[24] Kumar, Vipin, Jaideep Srivastava, and Aleksandar Lazarevic, eds. *Managing cyber threats: issues, approaches, and challenges*. Vol. 5. Springer Science & Business Media, 2006.

[25] Lakhina, Anukool, Mark Crovella, and Christophe Diot. "Diagnosing network-wide traffic anomalies." *ACM SIGCOMM Computer Communication Review*. Vol. 34. No. 4. ACM, 2004.

[26] Lakkaraju, Kiran, William Yurcik, and Adam J. Lee. "NVisionIP: NetFlow visualizations of system state for security situational awareness." *Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security*. ACM, 2004.

[27] Lomax, Richard G., and Debbie L. Hahs-Vaughn. *An Introduction to Statistical Concepts*. Routledge Academic, 2012.

[28] M. Thottan and C. Ji, "Anomaly detection in IP network," *IEEE Transactions on signal Processing,* Vol. 51, no.8, pp. 2191-2204, 2003.

[29] McPherson, Jonathan, et al. "Portvis: a tool for port-based detection of security events." *Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security*. ACM, 2004.

[30] Munz, Gerhard, and Georg Carle. "Real-time analysis of flow data for network attack detection." Integrated Network Management, 2007. IM'07. 10th IFIP/IEEE International Symposium on. IEEE, 2007.

[31] *Objects > Applications*, www.paloaltonetworks.com/documentation/71/pan-os/web-interface-help/objects/objects-applications.html , Feb. 2018.

https://www.paloaltonetworks.com/content/dam/pan/en_US/assets/pdf/framemaker/71/pan-os/web-interface-help/section_6.pdf

[32] Palo Alto Networks. https://applipedia.paloaltonetworks.com/ , Accessed date May 13, 2017.

[33] Patcha, Animesh, and Jung-Min Park. "An overview of anomaly detection techniques: Existing solutions and latest technological trends." *Computer networks* 51.12 (2007): 3448-3470.

[34] Penttinen, Tuomo. "Distributed denial-of-service attacks in the Internet." (2005).

[35] Raj, Sant, Neeraj Gupta, and Sumeet Gill. "Network Security using Soft Computing Technique." *International Journal of Electro Computational World & Knowledge Interface* 1.2 (2011).

[36] Ren, Xunyi, Ruchuan Wang, and Hejun Zhou. "Intrusion Detection Method Using Protocol Classification and Rough." *Computer and Information Science* 2.4 (2009).

[37] Santos, Omar. *Network Security with NetFlow and IPFIX: Big Data Analytics for Information Security*. Cisco Press, 2016.

[38] Scheck, Michael, and Cisco CSIRT. "NetFlow for incident detection." *Proceedings of Forum of Incident Response and Security Teams (FIRST), Kyoto, Japan*. 2009.

[39] Srinoy, Surat, and Werasak Kurutach. "Anomaly Detection Model Based on Bio-Inspired Algorithm and Independent Component Analysis." *TENCON 2006. 2006 IEEE Region 10 Conference*. IEEE, 2006.

[40] Stern, H. "A survey of modern tools." *Proceedings of the fifth conference on email and anti-spam*. 2008.

[41] Taylor, Teryl, et al. "Flovis: Flow visualization system." Conference For Homeland Security, 2009. CATCH'09. Cybersecurity Applications & Technology. IEEE, 2009.

[42] Yin, Xiaoxin, et al. "VisFlowConnect: NetFlow visualizations of link relationships for security situational awareness." *Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security*. ACM, 2004.

[43] Yurcik, William. "VisFlowConnect-IP: a link-based visualization of Netflows for security monitoring." *18th Annual FIRST Conference on Computer Security Incident Handling*. 2006.

[44] Zikopoulos, Paul, and Chris Eaton. *Understanding big data: Analytics for enterprise class hadoop and streaming data*. McGraw-Hill Osborne Media, 2011.