Spring 2015

A characteristic-based visual analytics approach to detect subtle attacks from NetFlow records

Weijie Wang
Purdue University

Graduate School Form 30
Updated 1/15/2015

PURDUE UNIVERSITY
GRADUATE SCHOOL
Thesis/Dissertation Acceptance

This is to certify that the thesis/dissertation prepared

By Weijie Wang

Entitled

A CHARACTERISTIC-BASED VISUAL ANALYTICS APPROACH TO DETECT SUBTLE ATTACKS FROM NETFLOW RECORDS

For the degree of Master of Science

Is approved by the final examining committee:

Baijian Yang
Chair

Yingjie Chen

Raymond Hansen

To the best of my knowledge and as understood by the student in the Thesis/Dissertation Agreement, Publication Delay, and Certification Disclaimer (Graduate School Form 32), this thesis/dissertation adheres to the provisions of Purdue University's "Policy of Integrity in Research" and the use of copyright material.

Approved by Major Professor(s): Baijian Yang

Approved by: Jeffrey Whitten 4/27/2015

Head of the Departmental Graduate Program Date

A CHARACTERISTIC-BASED VISUAL ANALYTICS APPROACH TO DETECT SUBTLE ATTACKS FROM NETFLOW RECORDS

A Thesis

Submitted to the Faculty

of

Purdue University

by

Weijie Wang

In Partial Fulfillment of the

Requirements for the Degree

of

Master of Science

May 2015

Purdue University

West Lafayette, Indiana

ACKNOWLEDGEMENTS

I would like to thank the members of my committee for their support and guidance. As my major professor and committee chair, Dr. Baijian Yang has been quite patient and encouraging over the past two years. To Dr. Yingjie Chen, I'm grateful for his invaluable suggestions on visual design. Also, I thank Dr. Raymond Hansen for serving on my committee and his thoughtful insights for the thesis project.

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

Figure Page

LIST OF ABBREVIATIONS

CSS	Cascading Style Sheets
DDoS	Distributed Denial of Service
DNS	Domain Name System
DoS	Denial of Service
FN	False Negative
FP	False Positive
FTP	File Transfer Protocol
HTML	HyperText Markup Language
IDS	Intrusion Detection System
IPv4	Internet Protocol version 4
SVG	Scalable Vector Graphics
TCP	Transmission Control Protocol
TN	True Negative
TP	True Positive
UDP	User Datagram Protocol
VA	Visual Analytics
VAST	Visual Analytics Challenge Committee

GLOSSARY

D3.js: a JavaScript and web-based data visualization library using HTML, SVG and CSS.

Data Exfiltration: illegal transfer of confidential data from a target network (victim) to a location which attacker can access the sensitive information

Efficiency: the ratio of number of detected attacks (i.e. true positives) to all events identified as an attack (sum of true positives and false positives) (Staniford et al., 2002).

Heat Map: a common graphical representation of data. It is a visual analytical technique in which the data values represented in a matrix are given different colors.

NetFlow: a series of packets between two hosts is combined into a single flow record which usually consists of the protocol, source and destination Internet Protocol addresses, the source and destination ports, payload size, session length and so forth (Goodall, 2007).

Visual Analytics: a data analytical approach that combines interactive visualizations with automatic analysis methods for a more comprehensive perception, reasoning and decision making process when processing massive and complex datasets (Keim et al., 2008).

ABSTRACT

Wang, Weijie. M.S., Purdue University, May 2015. A Characteristic-based Visual Analytics Approach to Detect Subtle Attacks from NetFlow Records. Major Professor: Baijian Yang.

Security is essentially important for any enterprise networks. Denial of service, port scanning, and data exfiltration are among of the most common network intrusions. It's urgent for network administrators to detect such attacks effectively and efficiently from network traffic. Though there are many intrusion detection systems (IDSs) and approaches, Visual Analytics (VA) provides a human-friendly approach to detect network intrusions with situational awareness functionality. Overview visualization is the first and most important step in a VA approach. However, many VA systems cannot effectively identify subtle attacks from massive traffic data because of the incapability of overview visualizations. In this work, we developed two overviews and tried to identify subtle attacks directly from these two overviews. Moreover, zoomed-in visualizations were also provided for further investigation. The primary data source was NetFlow and we evaluated the VA system with datasets from Mini Challenge 3 of VAST challenge 2013. Evaluation results indicated that the VA system can detect all the labeled intrusions (denial of service, port scanning and data exfiltration) with very few false alerts.

CHAPTER 1. INTRODUCTION

Presentlt, network systems and infrastructures are often enormous and complex, and tens of hundreds of servers and hosts are working simultaneously. A mass of network intrusions occur on a daily basis, trying to damage network service or compromise servers to steal confidential data and information. Confidentiality, integrity and availability are critical to any enterprise network infrastructures. Therefore, monitoring the status of network system and detecting intrusions are among the highest priorities of network administration and security analysts. Various approaches and solutions have been proposed and studied for intrusion detection, such as statistics-based, pattern-based, and rule-based (Liao et al., 2013).

Although numerous algorithms and systems have been studied and implemented, humans are still critical part in the network security process. Visual Analytics system has attracted attention from industry and academia because of its significant involvement of human analysts. Simply speaking, Visual Analytics combines interactive visualizations with automatic analysis methods for a more comprehensive perception, reasoning and decision making process when we deal with massive and complex datasets (Keim et al., 2008). Therefore, visual analysis of network data will help

human analysts to analyze and explore the mass amount of datasets efficiently, as well as to perceive patterns, trends, and exceptions. With the increasing size of computer networks and continuous appearance of new types of attacks, the research on visualization for network security is facing more and more challenges (Cook et al., 2012). In a large scale of network, detecting “subtle” attacks from massive amount of network traffic is difficult. Preprocessing, extracting, and analyzing the “big data” in visual analytics systems and tools is still an open challenge.

1.1 Research Topic

The concept of information visualization has been proposed for more than 15 years. Formally, information visualization includes the use of computer-supported, visual representations of data in order to enhance cognition with help of human exceptional perceptual capabilities (Card, Mackinlay, &Shneiderman, 1999).

The basic idea of information visualization is to display the data in some visual forms and representations, in order to understand the data, get insights into the data, interact with the data directly and come to conclusions by human analysts. Moreover, visual data techniques have proven to be an effective way in exploratory data analysis and they also show high potential for exploring large volume of databases (Keim, 2002). Information visualization has provided us a different perspective to explore data, the aim of Visual Analytics is to facilitate our way of processing information and data transparent for an investigative discourse. Visual Analytics is a step further than information visualization. It can rather be treated as an integral method to data

exploration and decision-making, combining information visualization techniques, human factors and data analysis (Keim et al., 2008).

This research primarily focused on designing and implementing a visual analytics system prototype to detect network intrusions, including port scanning, denial of service attacks, and data exfiltration. The characteristics of these attacks are heavily emphasized in the visual analytics, thus it is a characteristic-based approach. In particular, detecting subtle attacks from the massive amount of network traffic and how to extract the "right data" from "big data" are the primary interests.

1.2 Significance

Rapidly identifying and classifying malicious activities and intrusions through network traffic is a major challenge for human analysts exacerbated by complexity of data sets and functionally limited manual analysis tools. Even on a relatively small enterprise network, manual processing and analyzing of traffic data is extremely time consuming. Information visualization frees the human analysts from reading and examining large volume of data logs. A well-designed visualization graph can potentially summarize a week's or even a month's worth of network traffic and intrusion alerts, helping human analysts unearth the intrusion events. Additionally, visualization tools generally provide interactive components that facilitates human analyst to examine detailed information of any suspicious activities (Itoh et al., 2006). Comparing to statistics-based and pattern-based approaches, visual analytical approach is usually

intuitive, meaning no significant requirement of complicated mathematical or statistical knowledge (Shiravi, Shiravi, &Ghorbani, 2012).

Although there are many benefits of visualization systems for detecting network intrusions, developing a good analytical system and visualization tool is not an easy job because of the growth of network, the complexity of network traffic and the various types of network intrusions.

The first challenge is the massive amount and complexity of the network traffic data. Camacho et al. (2014) pointed out that network monitoring for security shares a number of features with other Big Data problems, the so-called 4 Vs: Variety, Veracity, Volume and Velocity. Even though there have been numerous studies related to visualization of intrusion detection systems, many of them have difficulties when dealing with large-scale of network data (Shiravi, Shiravi, &Ghorbani, 2012). In a network with hundreds or even thousands of servers and hosts, large amount of normal traffic data will cover the subtle attacks, which may use only a few traffic records, making them extremely difficult to be detected by conventional visualization. Therefore, this project tried to expand the research to focus on how to detect the subtle attacks in a large-scale of network effectively and efficiently.

The second challenge comes from higher demand for the overall view of the entire network status (Zhao et al., 2014). Conventional visualization systems are often limited in providing a relatively low-level view of the network because in most cases it's easier and more straightforward to visualize the low-level data. Consequently, human analysts need to scroll through multiple visual graphs, trying to find the correlation and

identify abnormal events and threats, which is often time-consuming and inadequate. A high-level view of the network status will significantly decrease the impacts of unnecessary details and amplify the underlying security events. Thus, providing an informative high-level view of the network status is another goal of this research.

1.3 Scope

This project primarily consisted of three major phases: design, implementation, and evaluation.

Based on the behaviors of network intrusions, design phase identified the characteristics and patterns of different intrusions and summarize the data of interest that should be preprocessed and visualized. Three types of network intrusions are the primary interests of this research: port scanning, denial of service attacks, and data exfiltration. The system has multi-view of visualization, providing high-level overview as well as low-level details of points of interest.

In the implementation phase, a web server powered by Node.js was set up, a MySQL database was used to store the raw and processed network traffic data, and visual components were implemented with a popular JavaScript visualization library D3.js. It should be noticed that D3.js is not security-specific; it's no more than a web-based data visualization library using HTML, SVG and CSS. The ease of use, powerful visualization capability, and support for all major browsers make D3.js a popular choice in web-based data visualization.

Finally, the system was tested and evaluated with the dataset from Mini Challenge 3 of VAST challenge 2013 (VAST, 2013). The dataset was chosen because the data is NetFlow data, which is more appropriate for high level overview visualization (more discussion on this in Chapter 2). And the dataset is collected from a mid-sized network (approximately 1,200 servers and hosts), thus it's large enough to test the scalability of the approach. The evaluation was conducted in two ways. The first is through analysis of ad hoc use-case attack scenarios, which is to analyze the timeline of some attacks events in the data (like when the attack happened and what hosts were the targets). This is a common technique in evaluation of security visualization (Shiravi, Shiravi, & Ghorbani, 2012). The second is a formal statistical evaluation by identifying all the suspicious activities in the dataset. Because Visual Analytics Challenge Committee (VAST) has provided ground truth for the dataset, it is convenient to evaluate the system in a systematic manner, such as calculating Type I and Type II errors.

1.4 Research Question

Because we're primarily focusing on three types of network attacks, the corresponding research questions that were studied are:

1. Is Visual Analytical approach capable of detecting denial of service attacks from NetFlow records?
2. Is Visual Analytical approach capable of detecting port scan attacks from NetFlow records?

3. Is Visual Analytical approach capable of detecting data exfiltration from NetFlow records?

Here we use statistical hypothesis testing to define the research questions more formally. False positive rate (FP) is the primary criteria and we set the thresholds as 0.10 In other words, if the false positive rate is less than 0.10, we presume that the VA approach is capable of detecting intrusions. For denial of service attacks:

Null hypothesis: FP <= 0.10 for detecting denial of service attacks

Alternative hypothesis: FP > 0.10 for detecting denial of service attacks

This is a one-tailed test and we assume a significance level of 0.05. Similarly, we have hypothesis for port scanning and data exfiltration respectively.

1.5 Assumptions

The following assumptions were identified as part of this research project:

1. The system was evaluated by the dataset from Mini Challenge 3 of VAST challenge 2013. The dataset was treated as real network traffic data, even though how the data was collected was not officially released.
2. Network traffic data collected was faithful to the actual network traffic and was not modified by any types of worms or attacks.
3. The dataset used by the visualization system should follow a relatively strict format and content; other types of network traffic data might not be able to directly work properly in the system.

4. The ground truth released from the VAST challenge committee, which was used to evaluate the visualization, was considered correct and comprehensive.

1.6 Limitations

The research was conducted acknowledging the following limitations:

1. The dataset used in the research only contains two weeks' network data and the types of network intrusions may be not comprehensive.
2. The dataset was artificially designed for visualization security study.
3. In some cases, NetFlow records cannot provide completely trustworthy information of network intrusions. For example, a distributed denial of service (DDOS) attack with spoofed IPs may be seen as a denial of service attack (DOS) from one external attacker.
4. The laboratory computer system had limitations in capacity and throughput, which was only capable of processing and visualizing data in the order of a few gigabytes.

1.7 Delimitations

This research was performed with the following delimitations:

1. The visual analytics system was designed to detect following types of network intrusions: port scanning, denial of service attacks, and data exfiltration.
2. The dataset was chosen to evaluate the system because it can represent major

network attack information and is designed to evaluate modern cyber-security visual analytic approaches.

3. D3.js is suitable for data visualization with HTML, SVG and CSS. D3's emphasis on web standards gives users the full capabilities of modern browsers. D3.js is not a security-specific visualization library.

1.8 Chapter Summary

This chapter has outlined an overview of the research project, including the research topic, significance, question statement and scope. Assumptions, limitations and delimitations of the research have also been presented. In the next chapter, relevant researches on network intrusions and visual analytics for security will be examined and discussed.

CHAPTER 2. LITERATURE REVIEW

In the past twenty years, various techniques and approaches have been proposed and heavily studied to detect network intrusions and attacks, such as statistics-based, pattern-based, rule-based and visual-based (Liao et al., 2013). Among them, visual-based approach attracts significant attention because of the interactive participation of human analysts. Human analysts have remarkable ability to handle novel patterns, outliners and exceptions. Comparing to other automated approaches, a visual analytics system provides human analysts with improved tools to detect anomalies, discover hidden patterns, identify inherent correlations, and communicate findings with colleagues (Goodall, 2008).

2.1 Comparisons of Intrusion Detection Approaches

Traditionally, there are two major approaches for intrusion detection, namely anomaly detection and misuse detection. With proposes and development of various systems and tools, people tend to subdivide these approaches into more subcategories (Liao et al., 2012; Bhuyan, Bhattacharyya, & Kalita, 2011). Four common approaches for network intrusion detection are discussed in this section: algorithmic, rule-based,

threshold-based, and visual-based. Other than these, pattern-based, state-based and heuristic-based are also widely used methodologies of intrusion detection. The algorithmic approach usually uses statistical tests, probabilistic analysis and models, or data mining to analyze network traffic (Jung et al., 2004). Abnormal network traffic and packets can be detected through this approach. Recently, Yen et al. (2013) designed a detection system called Beehive that automatically mines and extracts knowledge and insights from various logs data generated by a variety of network devices in a large enterprise. With the help of a common clustering algorithm (an adapted version of the K-means clustering algorithm), Beehive detected network intrusions (including the port scanning) that went otherwise unobserved by current security tools and personnel. Soft computing is similar to the algorithmic approach in many ways. Many methods in this approach use fuzzy logic-based algorithms, which provide flexible information processing for handing real-life ambiguous situations (Zadeh, 1994).

The second approach is the rule-based approach, which applies various pre-defined rules and policies to detect abnormal and suspicious traffic (Kim & Lee, 2008). In typical rule-based approaches, rules (If-Then or If-Then-Else) are used to build the model and profile of known and common intrusions. For example, in a rule-based system, a blacklist is often provided where all the traffic from the list's IP addresses is automatically blocked by the firewall or intrusion prevention system. Pattern-based approach is similar to rule-based, which uses pattern matching technique in the detection process.

The third approach is threshold-based, which is an intuitive and widely used technique to detect network attacks above certain thresholds by examining happenings of event X across a Y-sized time frame (Gates, 2006). Nonetheless, it is rather difficult and needs careful investigation to set an appropriate detection threshold: a low threshold may mislabel some normal activates; whereas a relatively high threshold would have difficulties to detect some malicious network traffic (Paxson, 1999).

The last approach is the visual-based. Visual-based approach is closely related to "security visualization" or visual analytics. Security visualization is a relative young term and it is a concrete field from the broader domain of information visualization (Marty, 2008). Security visualization has the benefits of information visualization but demands novel and fine-tuned techniques for thorough and in-depth analysis, because common visualization systems have been constructed for use scenarios that are not well supportive of detecting intrusions from network traffic (Shiravi, Shiravi, &Ghorbani, 2011). Visual analytics puts more emphasis and efforts on human side, and it provides a solution that combines the strengths and powers of human analysts and electronic data processing. Because for most enterprise networks, security administration is still a process that needs human involvement, visual analytics approach enable people to derive insights from dynamic, ambiguous and massive data, synthesize information, discover the unexpected and outliners, and communicate efficiently for further action (Keim et al., 2008).

An important advantage of visual analytics approach is the flexible incorporation of other approaches. In other words, techniques in rule-based or threshold-based

approach can be used in visual analytics. Well-designed visual analytics components can significantly help human analysts to process and understand the information and underlying indications from other approaches.

2.2 Categories of Network Security Visualization

Like all the information visualization, network security visualization is data-driven process. Goodall (2007) organized the network security visualization into three major categories based on the level of network traffic data to be analyzed and visualized: packet trace visualization, NetFlow records visualization, and security events visualization. Shiravi, Shiravi, & Ghorbani (2011) provided a detailed list of possible data sources that are accessible and can be used in the implementation of visualization tools and systems, and the related three categories are given in Table 2.1.

The first category is “packet trace visualization”, which is to visualize raw packet traces, the most granular level of network traffic data. Normally, packet trace data can be collected from packet analyzer such as Tcpdump and Wireshark. A network packet consists of two types of data: control information (also known packet header) and user data (also known as payload). A packet is the basic unit of data transmitted in a packet-switched network. Therefore, network packets theoretically contain all the information related to network intrusions.

Table 2.1 Potential Data Sources for Security Visualizations

Event Type	Data Source	Device & Software
Network Traces	Packet Trace	Tcpdump, Tshark, Wireshark
	NetFlow Records	Cisco NetFlow NDE, Cisco NSEL NetFlow
Security Events	Intrusion Detection Systems	Cisco CSA, Cisco IDS, Enterasys Dragon, FortinetFortigate, Juniper ISG, SNORT, NiksunNetVCR, SourceFireSensor
	Intrusion Prevention Systems	ForeScoutConterACT, Juniper NetScreen IDP, McAfee Intrushield, Radware Defense Pro, FireEye, Tipping Point X, IPAngel

Visualization systems like Portall, Radial Traffic, VISUAL, TNV and Svision all use packet traces as the primary data sources (Fink, Muessig, & North, 2005; Keim et al., 2006; Ball, Fink, & North, 2004; Goodall et al., 2005; Onut, & Ghorbani, 2007). Portall uses a “node and link” graph to represent the host and connection in a network. VISUAL utilizes scatter plot and IP matrix to depict the connections, allowing human analyst to check connection patterns between the internal network and external hosts. TNV is trying to provide a focused observation on packet level data without losing the high-level network context. As displayed in Figure 2.1, TNV shows approximately five

thousand network packets in a 90 minutes time period. A matrix with connection is used to depict network activities of hosts over time and each host in the matrix is painted in different colors based on its level of connection activity. Multiple linked views are further used to display details of raw packets (Goodall et al., 2005).

Figure 2.1 A screen snapshot of TNV system.

An obvious drawback of packet trace visualization presents an incredible amount of details, which challenges human analysts to understand the data efficiently. Even for

a relatively small network, the amount of involved packets is considerably significant, thus visualization at packet level is rather difficult and presenting too many unnecessary details to human analysts. As a result, packet trace visualization is more suitable in a hierarchy or multiple-layer deign, where detailed information of individual packets can be provided on demand (Goodall et al., 2006). On the other hand, systems like TNV are directly visualizing low-level network data in the overview without any aggregation and filtering. It's rather difficult for human analysts to observe any significant patterns or discover the underlying security events. Thus it is meaningful to provide an informative high-level view of the network status instead of directly visualizing low-level data.

The second category is "network flow visualization", which is to visualize the network flow data. In network flow data, a series of packets between two hosts is combined into a single flow record. In practice, Cisco NetFlow NDE or Cisco NSEL Netflow can be used to collect the NetFlow data for enterprise networks. A NetFlow record is typically consisted of the protocol, source and destination Internet Protocol addresses, the source and destination ports, payload size, session length and so forth. Therefore, NetFlow records are much more compact comparing to packet traces by sacrificing details and real payload data (Goodall, 2007).

Generally, such aggregations of packet trace in network flows remove the heavy burden of visualizing all the granular level details, thus it's widely used in many security visualization systems. NVisionIP, VizFlowConnect, NetBytes Viewer and NFlowVis are all using NetFlow as the primary data source (Lakkaraju, Yurcik, & Lee, 2004; Yin et al., 2004; Taylor, Brooks, & McHugh, 2008; Fischer et al., 2008). For example, Phan et al. (2008)

developed a system called “Isis”, which used progressive multiples of event plots and timelines to provide the iterative examination of network traffic using network flow data.

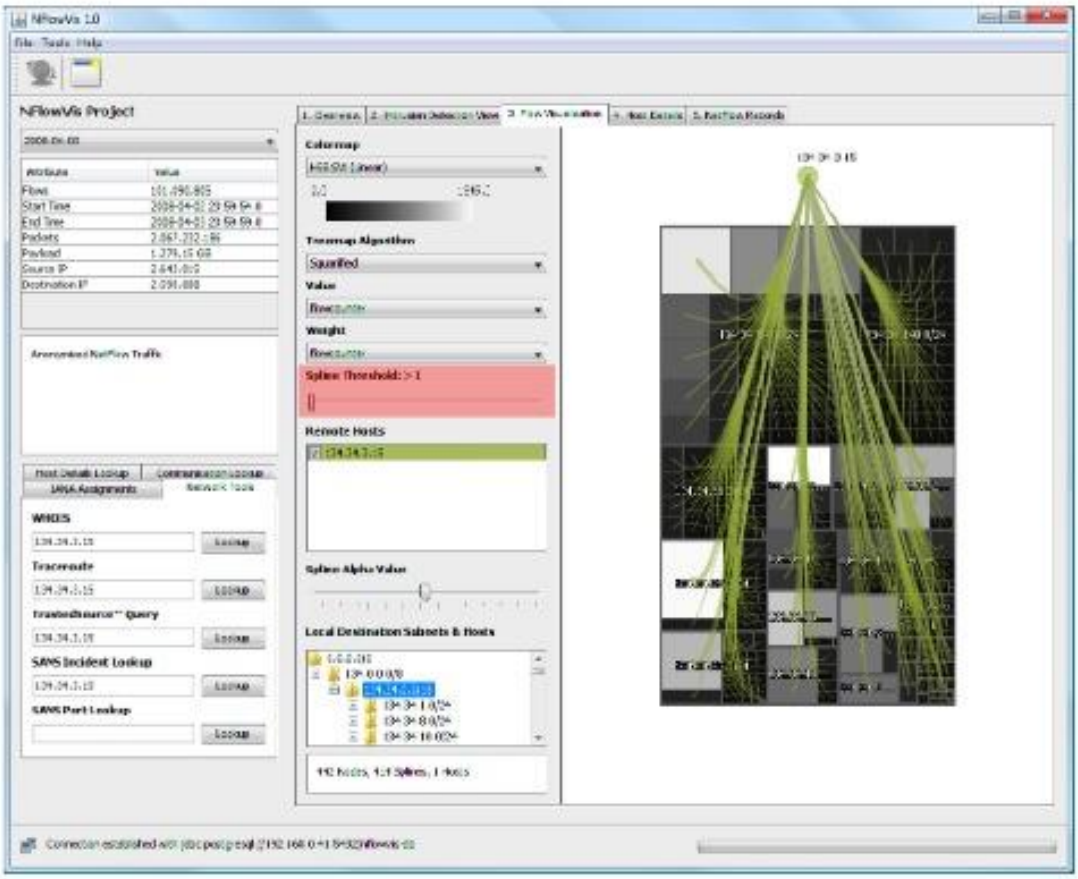

Figure 2.2 User interface of NFlowVis system.

Figure 2.2 displays the user interface of NFlowVis, which combines TreeMap visualization and a clustering algorithm to analyze NetFlow data. The user interface of the visualization system following a drill-down metaphor, guiding human analysts from overall network activity’s overview to aggregated views of IDS and NetFlow data. It

should be noticed that they utilize a clustering algorithm to analyze the traffic data, and the authors believe such clustering technique is suitable for data visualization in large-scale network. This example illustrates an advantage of visual analytics: it can incorporate other techniques and approaches (like clustering algorithm or pattern matching) in the visualization. Because network flows are much more compact than packet traces, we believe that the network flow data is more suitable for explore the temporal relationships of network traffic in large-scale, and the system enables human analysts to organize the visualizations to disclose traffic structure more easily.

The third category is “intrusion alert visualization”, which is to visualize the intrusion alerts from an intrusion detection system (IDS). An IDS monitors network traffic and generates security alerts for malicious activities or policy violations. There are many IDS products, such as Cisco CSA, Cisco IDS, Enterasys Dragon, and FortinetFortigate. Generally, intrusion detection in IDSs is an automated process (rule-based or signature-based) and the alerts data is a higher level of network traffic. Thus intrusion detection alerts data is a very common and important data source for many visualization detection systems.

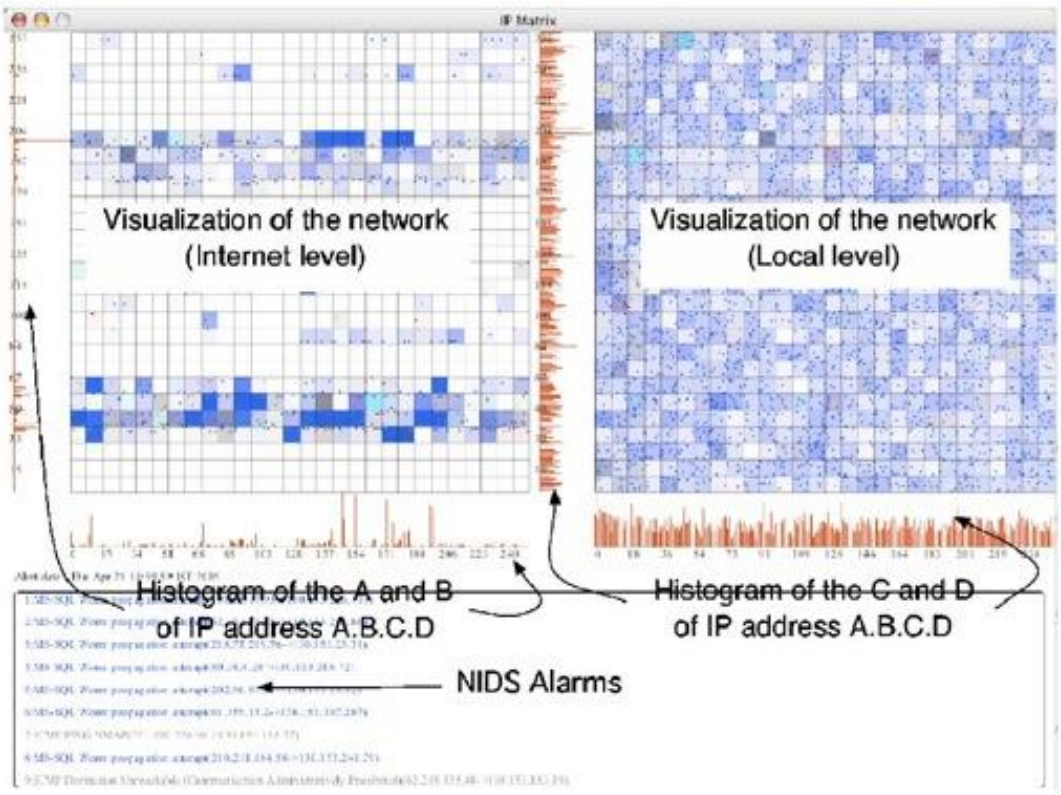

Figure 2.3 User interface of IP Matrix system, where the left view displays network activities at Internet-level and the right view displays activates at local-level.

Figure 2.3 is a snapshot of IP Matrix system, which directly visualizes the security alarms (Koike, Ohno, & Koizumi, 2005). The left view displays network activities at Internet-level and the right view displays activates at local-level. A security alarm generated by an IDS is mapped to a pixel inside the corresponding matrix cell and the color of the pixel indicates the type of the intrusion. There are many more systems utilizing the IDS alerts. For example, NIVA utilizes alert data from various intrusion detectors and use links and different colors to signify attacks (Nyarko et al., 2002). The

link color indicates the severity of intrusions. The system is able to present millions of nodes, but will fail when the alert data is significantly large. VisAlert incorporates a novel visualization paradigm, which displays visual correlation of network attack alerts from various logs (Livnat et al., 2005). The authors argued that an alert must have three attributes, namely: What, When and Where. A chord diagram is used to display the three attributes of intrusion alerts. Zhao et al. (2014) use a similar radial graph in their visualization system MVSec. The radar view provides an overview of alerts events and their inherent associations. Intrusion alert visualization system are solely replying on the IDSs and a major problem of IDSs, is the massive amount of security alerts they generate. The massive alerts on a daily basis can easily exhaust security analysts (Debar & Wespi, 2001). Systems like NIVA and VisAlert have difficulties when visualize intrusion alerts in a relatively long period of time. Additionally, false positives and false negatives are fairly common in IDSs, regardless of their detection mechanisms. Thus it's impossible to solely rely on intrusion alerts. Packet trace or data flow data should be used to assist in reduction of false positives and improvement of the system.

It should be noticed that there are many industry products in this category. Security Information and event management (SIEM) is a popular technology provides analysis of security threats and alerts (Ardito et al., 2000). Many vendors provide SIEM solutions, such as McAfee Enterprise Security Manager, SolarWinds Log & Event Manager, Splunk Enterprise, and HP ArcSight ESM. Although most of them have visual components, they are merely the visualization of the intrusion alarms from IDSs and the performance of system is heavily depending on the IDSs.

From the discussion above, it's clear that these three levels of visualizations have different advantages and emphasis. Packet trace visualization focuses on the most granular level of network data, resulting in introducing too many unnecessary details. Network flow visualization is at a higher level, and more suitable for visualization of larger-size network. Intrusion alert visualization utilizes the alerts from IDS as data source, and often needs lower level of data (packet or flow) to assist in reduction of false alerts and improve accuracy.

Here network flow data is chosen as the major data source because visualization of large scale of network is the foremost interest and concern in this research project. Packet trace and intrusion alert data can be used to extend the visualization or provide another level of information, but it's not the primary goal of this research.

2.3 Network Intrusions and Their Visual Detection Methodologies

2.3.1 Port Scanning

Port scanning is to seek open ports and available services on a network host by observing responses to connection requests (Vivo et al., 1999). Port scanning is very common, possibly the preliminary step in a network intrusion attempt. There are theoretically 65,535 ports for a computer host, where only small portions of them are "well-known" ports, such as 20 for FTP and 80 for HTTP. The many uncommon ports are probably being used by other software or services, depending on the situation. Attackers can uncover vulnerabilities of network hosts and launch corresponding attacks by gathering and analyzing the information of port scanning. Port scanning is normally

not a direct security threat by itself; however, security analysts can prepare for future attacks from the early detection of port scanning. In the practice, launching a port scan to a network or certain hosts is a trivial task with the help of software such as Nmap.

Port scanning can be detected in many visualization systems and tools. One common approach is to visualize host connections to classify patterns of port scanning. Conti and Abdullah (2004) use parallel coordinate plots to visualize the network traffic information, including IP addresses and port numbers. They have found some significant visual patterns, which are results of some common port scan attack software. Parallel coordinate plot is very suitable to visual multi-dimensional data. Network traffic information, such as protocol type (TCP or UDP), source IP address, destination IP address, source port and destination port can be used for parallel coordinate visualizations, and common port scanning would have some obvious fingerprints in the visualization.

In a similar approach, Jiawan et al. (2008) utilize network connections and mapped them to host-based visualization that highlights port scanning patterns in their system "ScanViewer", as depicted in Figure 2.4. In the paper, the host-based visualization used nodes to represent hosts (IP addresses) and lines to represent their inner connections. In the Figure 2.4, the two nodes in circle clearly have the connection pattern of port scanning. However, in reality the network traffic data is extremely large and dynamic, and thus the port scanning traffic in this visualization may be obscured by high-volume ordinary network connections, and hence the port scanning patterns cannot be identified by the human analysts.

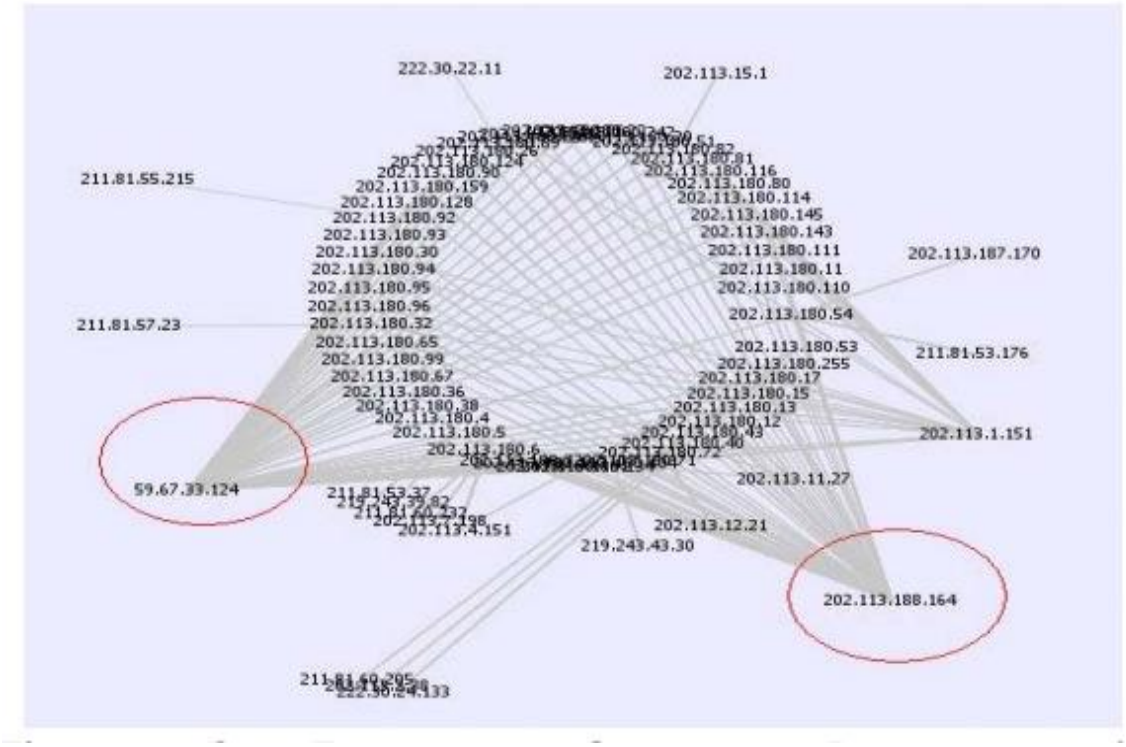

Figure 2.4 A snapshot of ScanViewer. It captures the pattern of port scanning or networking scanning. The two red-circled hosts are external attackers.

Another widely-used visual technique is port based because port activity of a network is essentially vital in port scanning. Fischer and Keim (2013) combined some interactive visualization views in their visualization system, with a tree map to display the most active ports and node-link graphs to represent and examine inner connections between different ports of network hosts. Port based approaches can be easily show the most active ports during a period of time, but the port scan patterns are not obvious.

2.3.2 Denial of Service

A denial-of-service (DoS) or distributed denial-of-service (DDoS) attack generally targets popular sites or services and attempts to disrupt or suspend the connection of the servers. A common method of DoS attack is server overload, which is to saturating the server with large amount of communications requests, making it cannot respond to legitimate requests normally. From the viewpoint of enterprise network, such DoS or DDoS attack is usually easy to be detected because the sudden high volume traffic to the internal servers. Similar to port scanning, DoS or DDoS attack has the obvious connection pattern: one or many external hosts (attackers) attempts to suspend the service of specific internal servers (one-to-one pattern or many-to-one pattern).

Denial of service attacks can be readily identified by many visualization systems because of the extremely massive burst of network traffic. NVisionIP displays a snapshot of the activities of the network hosts, providing the information such as connection activity and port activity. In the cases where internal servers under surveillance are attacked by DDoS, human analysts will notice abrupt bursts in network traffic from these servers (Lakkaraju, Yurcik, & Lee, 2004). However, the authors also pointed out if the DDoS attack was adequately distributed, NVisionIP will not identify the attackers and victims, as the volume of network packets sent by each involved attacker will be too low to differentiate from normal network traffic. NVisionIP focuses on visualizing external hosts, namely the attackers. Another approach is to visualize the activity of internal servers. For a high-profile web server, it’s impossible to flood the service by a

limited of requests. Thus in most cases, denial of service attack in terms of server overload is quite easy to be identified.

Other than the “flooding” method, denial of service can be launched by in many different ways, such as disruption of routing information or crash the server by malwares. These attacks don’t have the significant volume of involved traffic, but it can be detected by visualizing the working status of server (Zhao et al., 2014).

2.3.3 Data Exfiltration

Data exfiltration is another critical intrusion for any enterprise network and the possibility of a sensitive data leak lies among the highest fears of security analysts. Detecting data exfiltration is a challenging problem, because it’s not always easy to identify which data is leaving the enterprise network legitimately, and which data traffic is data exfiltration on purpose (Giani, Berk, & Cybenko, 2006).

Unlike port scanning or denial of service attack, data exfiltration has no obvious pattern from network connections because it often occurs from one internal host directly to another external host. Moreover, network flow data doesn’t have the payload content, thus it’s impossible to directly visualize the payload or its signature. D’Amico & Kocka (2005) pointed out that volume of data transferred can be used to identify data exfiltration. In most cases, when large volume of data transferred occurs from a host that is not recognized for such activities, it might indicate a potential data exfiltration. Thus it is possible to identify data exfiltration through visualizing payload sizes of network flows. It should be noticed that because not all the data exfiltration

involve large data transfer or the exfiltration can be processed in a slow manner (such as slow scan), this detection approach of visualizing data transfer sizes is not comprehensive.

Foresti et al. (2006) used another approach to identify data exfiltration. They believed they could characterize an external attacker with five distinct stages, which are reconnaissance, probe, attack, dig-in and migration. During the five stages, as it moves from normal network activity to data exfiltration, the visualization will show how the node under attack slowly emerges out of the background. For a subtle data exfiltration (no significant payload size), the attack stage might be difficult to be detected and visualized but the probe stage could provide more relevant information, which usually involves a network scanning or a port scanning. Goodall & Sowul (2009) detected data exfiltration in a similar approach. First they identify a slow scan, which took place over a period of about one hour. As they further investigate the event, they found that every connection to port scan attacker consisted of a small payload packet, so it is doubtful that the source host tried to steal any data. In this case, they also used an earlier probe stage to identify the data exfiltration.

In the research project, because packet payload content is not available, large volume of data transferred from a host that is not recognized for such activities may be the only practical way to identify data exfiltration.

2.4 Chapter Summary

This chapter first provides a brief summary of different network intrusion detection approaches, which include algorithmic, rule-based, threshold-based, and visual. Next, it focuses on visual-based approaches and describes three levels of network security visualization based on different levels of networking data to be visualized, which are packet trace visualization, network flow visualization and intrusion alert visualization. In the last part of this chapter, four types of network intrusions (port scanning, denial of service attacks, botnets, and data exfiltration) and their visual detection methodologies are discussed.

CHAPTER 3. METHODOLOGY

This chapter will cover the methodology of the research project, primarily introducing and outlining the major research phases, system components, data sources and analysis, and system evaluation used in the project.

3.1 Design Framework

The design phase was the first and the foremost important step of the research. Based on the network flow data and characteristics of different types of intrusions, the visual analytics design was characteristics-based. On other words, the design primarily focused on how to represent and visualize patterns of intrusions based on the NetFlow data. Because the data is typically in large-scale, techniques such as aggregating and filtration were used to amplify the underlying security events.

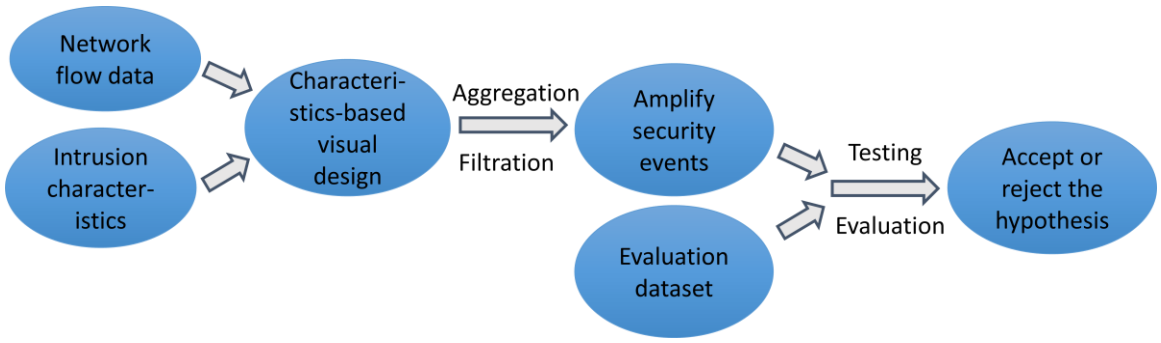

Figure 3.1 Methodology for this research project

3.2 Major Research Phases

The research project was consisted of three major phases as in Figure 3.1, though they're not inherently independent.

1. Designing the visual components of the system. This is the most important step of the research project. Generally speaking, the designing phase determined what information in the network traffic data would be used and how they would be visualized in the system. Based on the characteristics and patterns of different network intrusions, corresponding information and metrics of network traffic data were retrieved, summarized and visualized.
2. Implementing the visualization system. Implementation phase included following procedures: preprocessing the data and importing it to a database, setting up the web server and coding the visualization part of the system.
3. Evaluating the visualization system. For the dataset that was used in the project, information and facts about the network intrusions are listed in a separate file. Thus by comparing with the "ground truth sheet", the visualization system can be evaluated. In fact, based on the feedbacks of evaluation, the designing and implementation of visual components were modified accordingly to achieve better results. This is like the typical iterative cycle of software development.

3.3 System Workflow

A typical workflow for Visual Analytical systems is displayed in Figure 3.2 (Zhao et al., 2014). For this project, the first workflow was data preprocessing. After cleaning and aggregating the raw data, the preprocessed data were imported into database. The visual analytics component connected to the database, retrieving necessary information from the aggregated data. Meanwhile, the visual analytics components provided an interface for human analysts to interact with the system.

Essentially, the system was a web application, which consisted of a backend database, a web server, and frontend web interface.

1. A MySQL database was used to store the network traffic data.
2. A simple web server powered by Node.js. Node.js is an open-sourced platform to build fast, scalable network applications.
3. Frontend web interface was built with standard HTML, CSS and JavaScript. In particular, an open-sourced JavaScript library d3.js was heavily used to build the visualization components. It should be noticed that D3.js is not security-specific; it's no more than a web-based data visualization library using HTML, SVG and CSS.

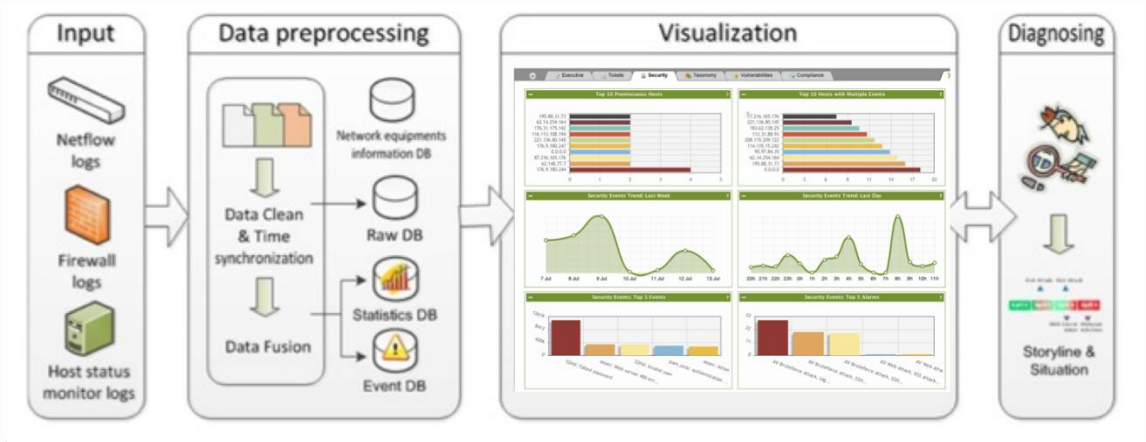

Figure 3.2 A typical of workflow for the visual analytics system.

3.4 Data Sources and Data Analysis

In the evaluation phase, a dataset from Mini Challenge 3 of VAST challenge 2013 was heavily used. The original dataset includes three separate CSV files: network traffic data, network hosts status data and network intrusion alert data. The network traffic data is the primarily data used in the research project. It contains of two weeks of network traffic data (flow-level) for an enterprise network, which includes approximately 1,200 hosts and server. The data has roughly 70 million data records, and takes about 4GB disk space. More specific, the network traffic data (NetFlow) contains IP addresses, port numbers, transaction payloads, transaction time length and other information about the network transaction through the local network.

3.5 System Evaluation

The evaluation was conducted in two ways. The first is through analysis of ad hoc use-case attack scenarios, which is a common technique in evaluation of security visualization (Shiravi, Shiravi, &Ghorbani, 2012). In the ad-hoc evaluation, different types of intrusions, including port scanning attack, denial of service attack, and data exfiltration were fully evaluated. Ad-hoc evaluation is a typical way to evaluate a security system. Also, how the system is interacted with users (security administrators and analysts in this case) is also important, such as the process speed of the system, ease of use, and so on.

The second is a formal statistical evaluation by identifying all the suspicious activities in the dataset. Aground truth sheet has been provided by the Visual Analytics Challenge Committee (VAST), and the security events have been labeled into two categories: “obvious” and “subtle”. Generally, subtle attacks is much harder to be detected, thus they are the primary interests of the evaluation process. It is convenient to evaluate the system in a systematic manner with ground truth, such as measuring efficiency, calculating Type I and Type II errors.

Efficiency is defined as the ratio of correctly detected attacks (i.e. true positives) to all events flagged as an attack (sum of true positives and false positives) (Staniford et al., 2002). Type I and Type II errors are often used interchangeably with the general notion of false positive and false negative, respectively. False positive (Type I error) and false negative (Type II error) are two indicators to assess the accuracy of IDS system. False positive arises when the system mistakenly classifies normal activity as being

malicious, whereas the false negative occurs if the system fails to classify malicious activity.

FP and FN are very import factors when assessing an IDS system, but they are closely related to the volume of normal traffic compared with the volume of attack traffic (in other words, the acceptable rate for FP and FN are quite different for different networks). For example, given that attacks are fairly infrequent comparing to the normal traffic, even when the FP rate is quite low, there are still massive false alerts generated by IDS (Bhuyan, Bhattacharyya, & Kalita, 2011). In the case, the FP rate (10% or less) is still not acceptable for the network intrusion system. However, for evaluating this system prototype, false positive rate 10% is adequate and acceptable in most cases considering the relatively small number of alerts raised. Therefore, in our research, the hypotheses (the proposed visual analytics approach is capable of detecting subtle attacks from network flow records) will be accepted when the false positive rate is less than 10%, otherwise it will be rejected.

There are other merits widely used in evaluating IDS, such as efficiency and detection rate. They were all calculated in the evaluation part, but not as the criterial to assess the hypothesis of the research question. Goodall (2009) uses a popular packet capture analysis tool called Wireshark in the evaluation for the comparison purpose. In his experiment, Wireshark's efficiency is 72.9%; meaning 72.9% of all the alerts are truly attacks.

3.6 Chapter Summary

This chapter has provided an overview of research methodology of this research project. At first, it covered the three major phases for this research: namely designing phase, implementation phase, and evaluation phase. This is a typical cycle for software development. Next, the software and visualization libraries used in the research project were introduced and they either have free versions or are open-sourced software. At last, the data source and data analysis approach used in the system evaluation phase was outlined.

CHAPTER 4. VISUAL ANALTYICAL AND OVERVIEW DESIGN

Overview design is the first and most important step in the project. Before that, we must gain some understanding of and insight into NetFlow data, which is the primary data source format for this project. The insights will guide and facilitate the overview design process, and in the meantime feedback from the overview will broaden our knowledge about NetFlow data. This is a “mutual benefit” process.

In the process of creating the overview design, three primary goals are of particular interest. The first is to highlight suspicious events or hosts from normal background traffic data. This is the fundamental purpose and function of an overview. In particular, it is challenging and significant to search for subtle attacks in the overview because they usually hide deeper in the traffic and are more difficult to detect. By “subtle” we mean attacks with few related inconspicuous NetFlow records, or in a very short period of time, or with some characteristics that make them more difficult to detect.

Second, the overview should be scalable. As discussed in the literature review section, some effective intrusion detection methods have difficulty with a large amount of network data. This is generally normal for visual analytics because many VA solutions rely on human analysts to discover abnormal and suspicious parts, while the

abnormality will become inconspicuous in a large amount of data. If VA solutions rely only on the raw data to visualize and do not provide an effective method (such as data mining) to reduce the negative impact of the data size, scalability is rather hard to satisfy.

In our overview design, we are not using the conventional visual techniques (such as time series or connection visualization) directly; instead, we are trying to highlight security events based on the attributes of aggregated NetFlow records and characteristics of intrusion attacks. Therefore, the overview in our approach has some degree of scalability.

The last goal is to make the overview extendable. Even though three types of attack are the primary interest of the project, the VA solution in our project should provide general information about network status (situational awareness) and the ability to be extended to detect other network security attacks in the future.

Therefore, in Chapter 4, with these three motivations in mind, the process of visual analytical and overview design and details of the VA system are presented and discussed, with emphasis on the characteristics of NetFlow and the features of network attacks. In particular, conventional visual analytical techniques, such as time series plots and heat maps, are discussed to address why they do not fit into the overview directly but can be useful in other ways. Then two primary overviews of this project are presented, explaining why they can enable human analysts to identify network intrusions and provide scalability and extensibility at the same time.

4.1 Time Series Analysis

NetFlow is the primary data source for the project, and a significant characteristic of NetFlow data is that every entry is associated inherently with a timestamp that indicates when the NetFlow is recorded. Therefore, time series analysis is a straightforward method to inspect NetFlow data.

A simple way to visualize data associated timestamp is to display the number of NetFlow records (NetFlow count) over a time interval. Figure 4.1 illustrates a time series of NetFlow count on April 13 for Big Marketing Network. Because this time series simply counts how many NetFlow entries were recorded per minute, it can reveal intensity of network activity: A strong peak in time series generally indicates a sudden event involving a great number of NetFlow entries, which may be a normal network traffic peak, a denial-of-service (DOS) attack, a port-scan attack, or a number of other options.

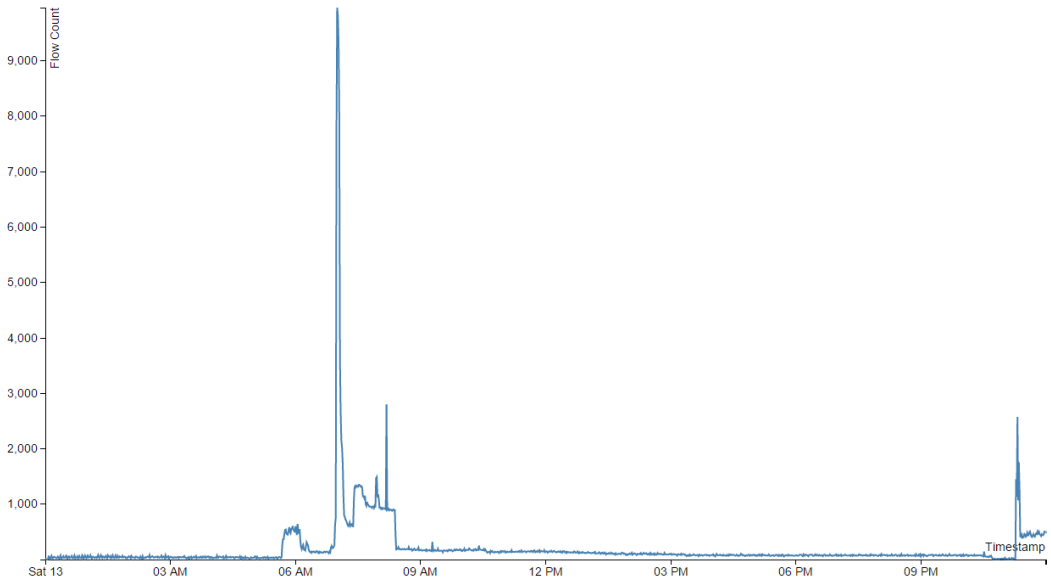

Figure 4.1 Time series of NetFlow count for Big Marketing Network in Apr 13

From Figure 4.1, it is clear that there are several significant peaks between 6 a.m. and 9 a.m., the number reaching almost 10,000 per minute at one point. This is abnormally huge, considering the baseline number. Most of these peaks in the early morning are actually the consequence of a port-scan attack targeting a few internal servers. Other than the noticeable peaks, however, time series analysis provides little insight regarding network status. For example, from 9 a.m. to 9 p.m., the time series seems flat, and if there were a data exfiltration event (which usually uses very few NetFlow records to communicate), it would be difficult to detect from the NetFlow count time series. Therefore, the drawback of time series analysis in this case is obvious: Only NetFlow count is inspected and under consideration, whereas subtle network threats involving only a few NetFlow records cannot be seen easily from the overview. Attacks such as data exfiltration and malware infection are generally related to a very limited number of hosts and NetFlow records.

Analysis of Figure 4.1 shows that it is impossible to find various attacks based on one parameter (NetFlow count). An improvement over this approach is to present different parameters (attributes of NetFlow) for time series analysis at the same time, such as number of distinct IP addresses, number of distinct port numbers, and so forth, as in Figure 4.2. In this multiple time series, the blue graph at the top represents NetFlow count, the orange graph in the middle represents distinct source IP addresses, and the green graph at the bottom represents distinct source port numbers. Observing and comparing these three graphs simultaneously reveals more details of network activity. The blue and green time series are very similar to each other in their shapes

and trends, indicating that attackers might use different ports to increase the throughput of attacking traffic, making the attack more severe.

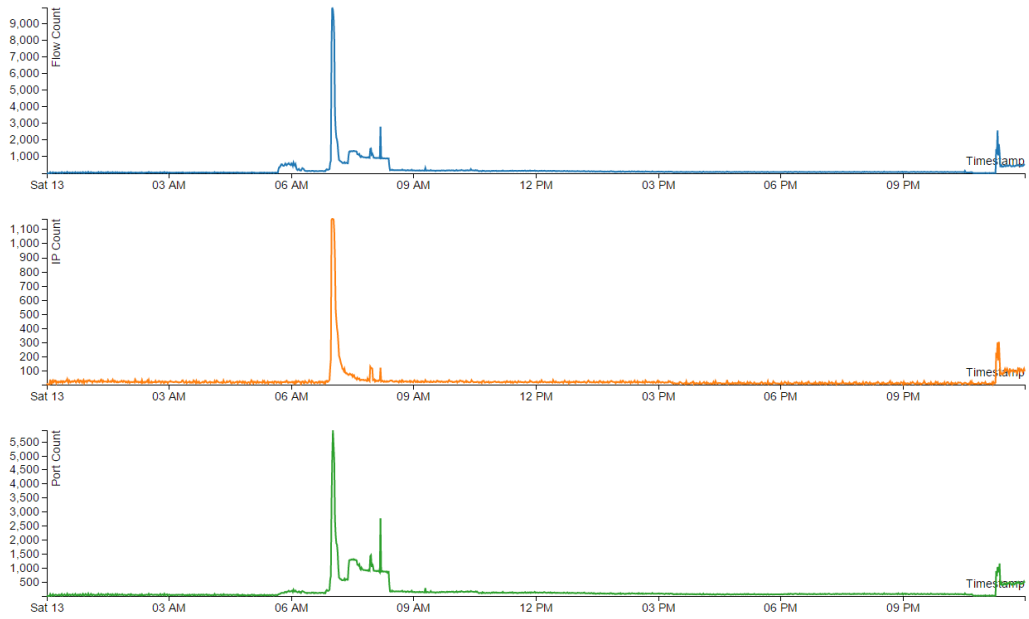

Figure 4.2 Multi-Time series of NetFlow count, IP count, port count for Big Marketing Network in Apr 13

Multi-time series can provide information about different aspects of network traffic. Potentially, it can depict one time series for each NetFlow attribute. This is still not a great solution to the problem, however. First, it is not easy to find correlations between the graphs. If we present six time series graphs at the same time, finding the relationships among them is not easy for human analysts. Second, multiple time series still present the same problem as a single time series: Subtle security events are often

overshadowed by obvious and significant network traffic and events. Therefore, even though time series analysis is a good approach for analyzing network traffic trending from different angles and providing general situation awareness information, it is not always scalable when dealing with a large amount of network traffic data.

4.2 Heat Maps Analysis

A heat map is a common graphical representation of data. It is a visual analytical technique in which the data values represented in a matrix are given different colors.

Essentially, time series spread the data along one parameter, which is usually a timestamp. Because heat maps display data in a matrix, it is appropriate to present the distribution of data in two dimensions. As with time series, an inevitable disadvantage is that subtle events will be overshadowed by other more significant events and normal background traffic. In a 2-D heat map, we can break NetFlow into different matrix cells according to their attributes to highlight both obvious events and subtle ones. Ideally, subtle security events can be extracted from the background data by some attributes. This is also a core notion in our VA design, which relies heavily on the characteristics of intrusion attacks.

In NetFlow content, in addition to timestamp and IP information, a few fields are closely related to attacks of interest: port number, payload size, and session duration. Port-scan attacks are clearly associated with targeted ports because the purpose of a port scan is to retrieve information about the port activities of hosts. Payload size is a primary factor in determining data exfiltration because NetFlow does not have any

information about the real content of payload, and we can only rely on the payload size to find suspicious traffic; an unusually large payload size generally indicates a potential data exfiltration. Meanwhile, monitoring payload size has some possible use; many companies strictly forbid uploading or downloading large files through enterprise networks. In the NetFlow data, there are two fields related to flow payload: "firstSeenSrcPayload" and "firstSeenDestPayload." This is because NetFlow represents a series of packets transferred between two hosts, possibly indicating direction. In this case, "firstSeenSrcPayload" means the total payload of all packets to "firstSeenSrcIP," and "firstSeenDestPayload" means the total payload of all packets to "firstSeenDestIP."

Furthermore, attacks such as denial of service and port scanning often present smaller payload size and shorter session length than usual. This is easy to understand; A normal user visits a website and would expect to retrieve content from the website, but an attack that launches a denial-of-service or port-scan attack attempts to shut down the service or simply obtain port information as quickly as possible, resulting in small payload size and short session duration.

Similarly, it is impossible to detect redirection behaviors via HTTP status code or packet headers because NetFlow does not provide such information. In a common server redirection, an external user connects to the infected server, and usually the page will be redirected to external malicious websites right away, resulting in extremely small payload size and short duration. As a result, small flow size and short flow duration are also two primary characteristics of server redirection.

This analysis demonstrates that payload size and session length are two key parameters in the detection of network attacks such as data exfiltration, denial of service, and server redirection.

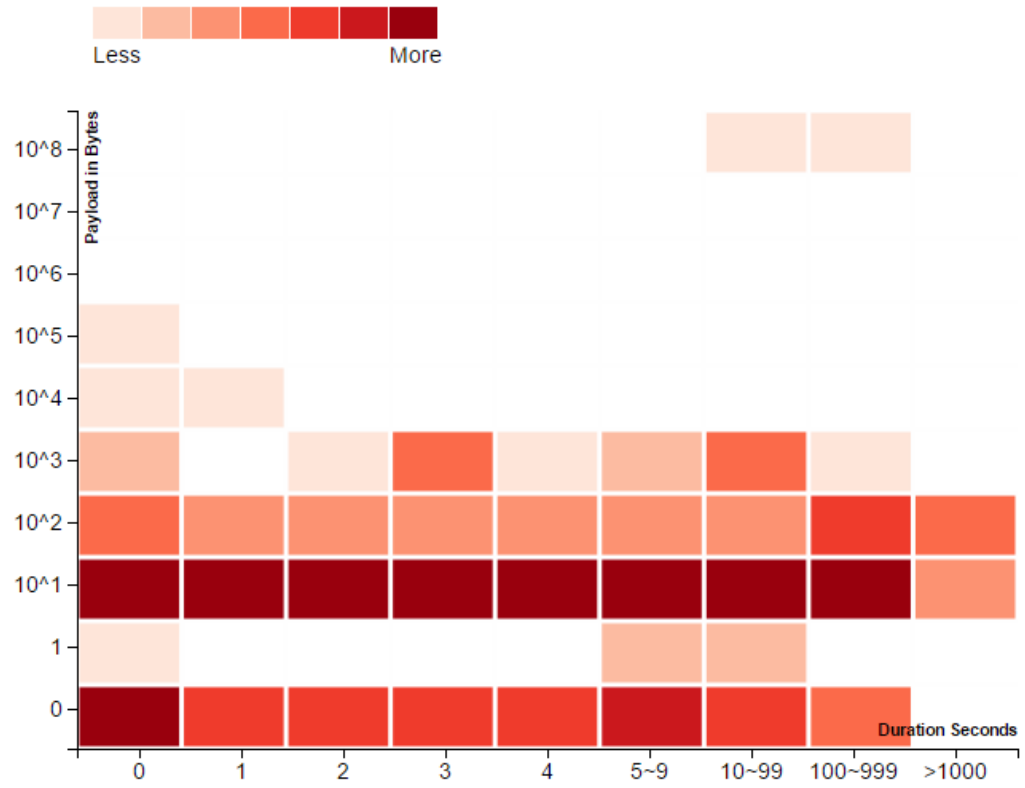

Figure 4.3 A heat map for NetFlow in Big Marketing

Therefore, NetFlow duration and payload can be used as primary attributes in heat maps. Figure 4.3 shows a heat map for Big Marketing Network's NetFlow, where the x-axis represents session duration length (in seconds) and the y-axis represents payload size (in bytes). The different shades of color indicate different amount of

NetFlow and darker cells indicate more NetFlow recorded in that range. In Figure 4.3, some cells in the matrix are blank with no color pixel, which means that there were no NetFlow in those ranges.

A heat map can be used to represent distribution of NetFlow in a time frame, such as 1 hour, 1 day, or even 1 week. A potential problem here is that one heat map displays only a chosen time period; it cannot clearly present changes in network status (trending during the time period), which is important in analyzing security events. Security analysts need to go through many heat maps to monitor a network. For example, if one heat map represents 1 hour of NetFlow, 2 weeks of data for Big Marketing Network will produce approximately 336 graphs, and examining all of them is a tedious job. On the other hand, if the time frame if too long, such as 1 day or even 1 week, the overview potential loses a lot of information because it is impossible to use color to represent one day's network status; should it represent the maximum, minimum, or average value?

4.3 Duration-Payload Overview Design

From the analysis of time series and heat maps, we see that both have inherent disadvantages for dealing with NetFlow data. In this analysis we combine the two visual techniques for our overview, producing a heat map that contains time series plots in the matrix. Heat maps can help separate data into groups based on their attributes, and time series plots provide baseline information to identify any abnormal parts for each group.

As we discussed in the previous section, session length and payload are primary attributes that can help distinguish malicious NetFlow records from normal ones. As seen in Figure 4.4, session duration and payload size are presented on the x-axis and y-axis, respectively, and each cell marked as a rectangle represents the time series for NetFlow count in the corresponding range during 2 weeks. For example, the bottom left corner cell represents a time series for NetFlow with zero duration and zero payload size.

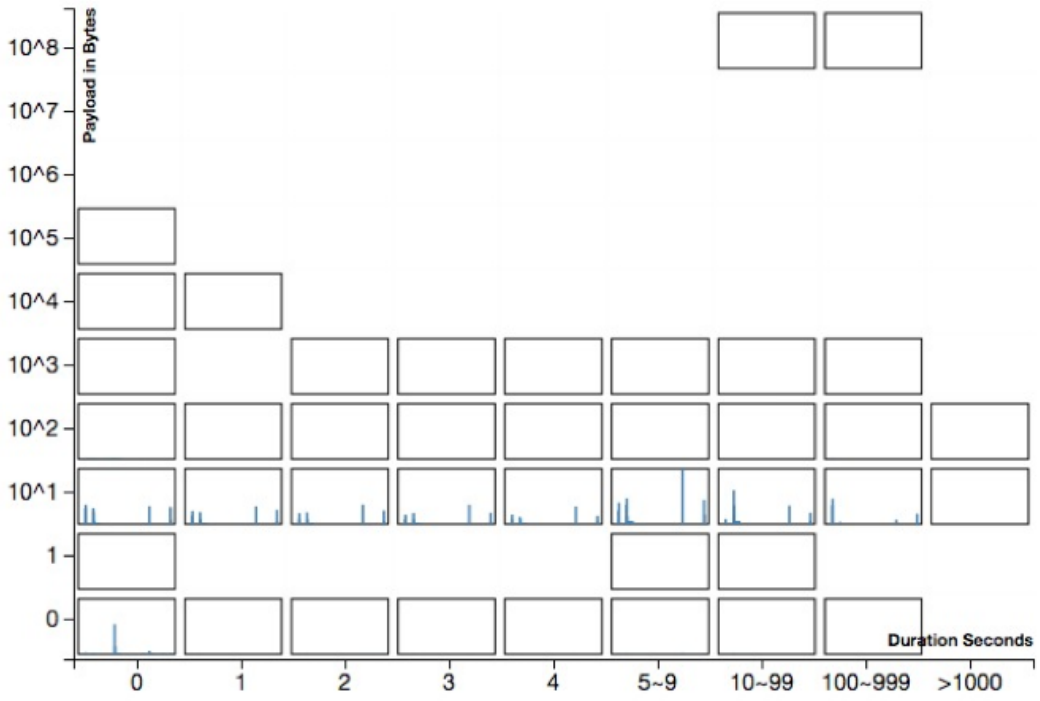

Figure 4.4 Duration-Payload Overview Design with uniform linear scale

Surprisingly, there are only a few peaks in the range of 10 bytes payload, while most rectangular cells have no visible peaks. In this implementation, the time series map has no color representation, and each cell in the map uses the same linear scale on the y-axis; in other words, the highest count determines the y-axis domain for all cells. In this way, every cell can be compared easily because they use the same scale. A significant drawback of such a design is that some small-count events cannot be seen easily from the overview because the largest count can be a thousand times or even a million times larger than others. This is why only a few cells have visible peaks—because their NetFlow counts in these cells are significantly larger than other ranges' values. In fact, further study confirmed that those significant peaks represent denial-of-service or port-scan attacks.

A better method is to use a logarithmic scale in the y-axis. A logarithmic scale can display small numbers better, as in Figure 4.5. Obviously, more peaks become visible with a log10 on the y-scale in each rectangular cell. Nevertheless, people generally have difficulty perceiving the values in a logarithm. For example, $\log_{10}(100) = 2$ and $\log_{10}(1000) = 3$. Though 100 and 1000 are very different quantities, 2 and 3 are relatively close when using visual representation. In Figure 4.5, it is difficult to tell the difference among the peaks (many seem to have very close values, but in fact they vary a lot).

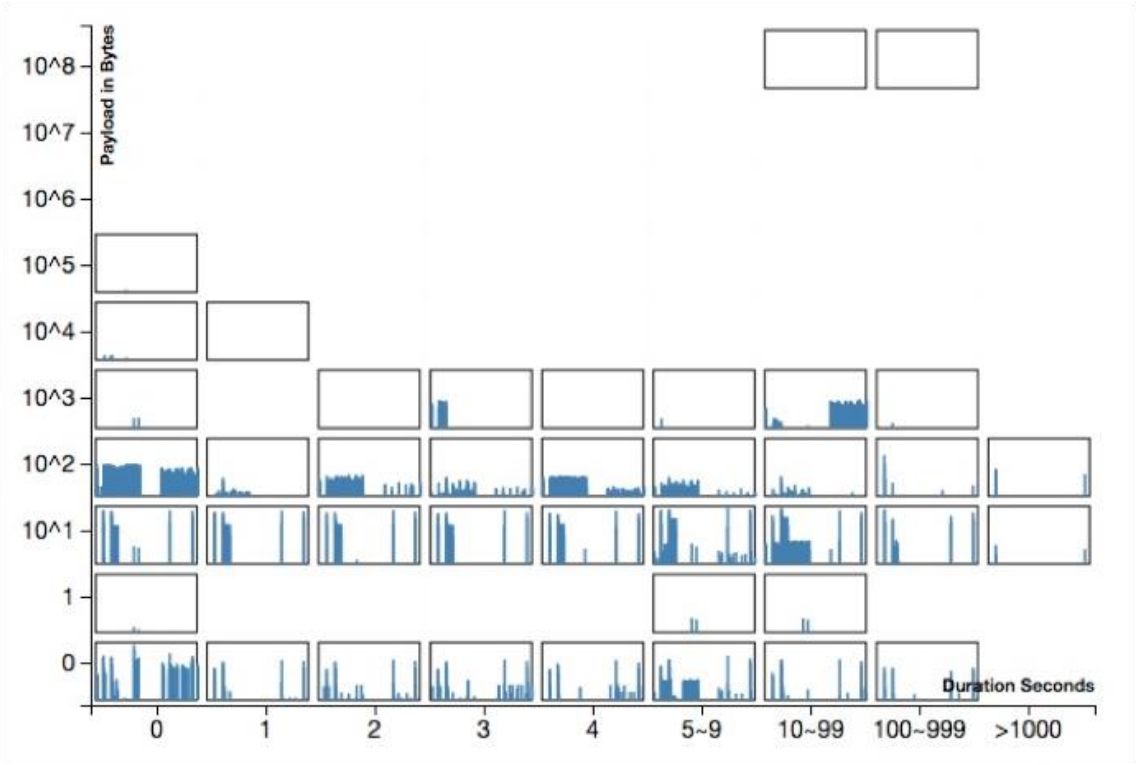

Figure 4.5 Duration-Payload Overview Design with uniform logarithmic scale

If all the matrix cells use the same scale on the y-axis, it is difficult for analysts to understand the overview. As in heat maps, color can be used to further distinguish different ranges of time series values. In other words, different scales are represented on the y-axis with colors, as in Figure 4.6. The shades of green represent the scale in each cell; the darker the shade, the larger the scale. If two cells have the same green background, they are using the same scale on the y-axis. The scale value increases by approximately an order of 10.

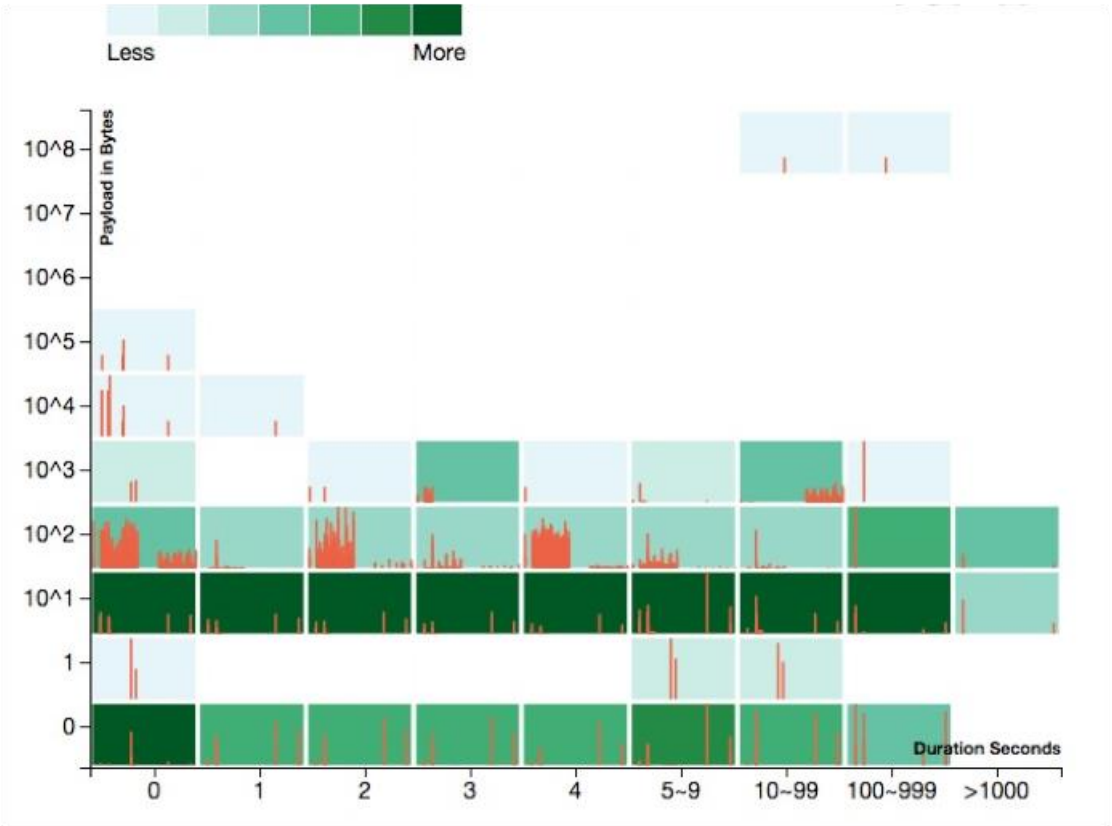

Figure 4.6 Duration-Payload Overview Design with colored scale

Essentially, Figure 4.6 contains the same data that Figures 4.4 and 4.5 do, but the data in this visualization, even very small amount of NetFlow, can be seen and compared easily. A closer look at Figure 4.6 indicates that the cells with the background of the darkest green are the ones visible in Figure 4.4, because those are the time series with the largest values (NetFlow count).

In Figure 4.6, the values chosen for different ranges need careful consideration. On the y-axis, the payload size in bytes basically increases by an order of 10—0, 1, 10, 100, and so on—to the largest payload size. This is a very straightforward way to divide

the payload values. The situation is quite different for duration length on the x-axis, however. Many normal NetFlow sessions last 3 or 4 seconds, and very few last 1 or 2 seconds, so we pay more attention to the lower durations by dividing length into 9 ranges for Big Marketing Network: 0, 1, 2, 3, 4, 5~9, 10~99, 100~999, and > 1000 seconds. Ideally, analysts should be able to define the ranges in their own implementations because normal ranges will vary for different networks.

After discovering potential problems from the overview, we provide a second-level visualization that offers a zoomed-in view of the cell graphs in the overview. When the user clicks on any cell graph in the overview, the system will automatically open a larger, zoomed-in image of that cell graph, as seen in Figure 4.7. The larger graph is a scatter plot, where each point represents the count for 1 hour. We use 1 hour as the time frame because we are dealing with 2 weeks of data. If the data is for 1 day, the time frame can be changed to minutes.

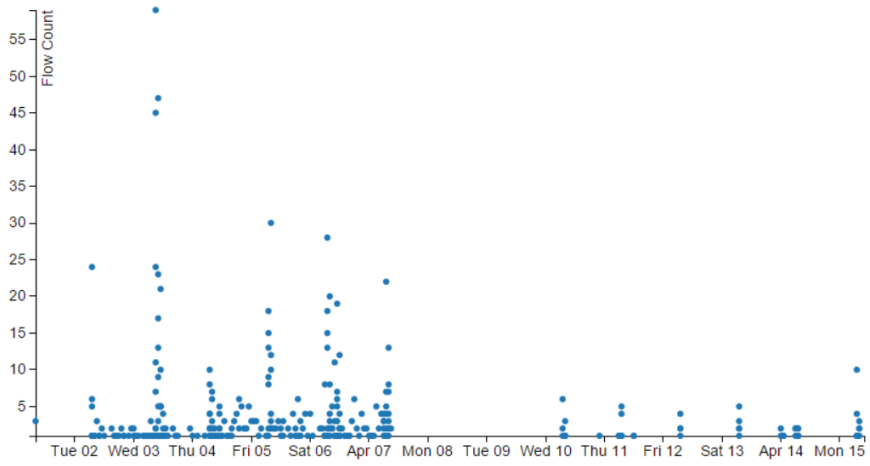

Figure 4.7 Scatter plot of one cell graph from the VA overview

Furthermore, an important feature of the VA overview is that it enables users to analyze multiple cell graphs simultaneously. Analysts can choose a few cell graphs in the overview, and the second level will visualize the content in the overview at the same time by applying different colors to the scatter plot points. To distinguish the points more clearly, each cell graph will be randomly assigned a border color that is highlighted when the cell is chosen, as shown in Figure 4.8(a), and the corresponding points in the second layer visualization will use that color, as shown in Figure 4.8(b). We think this feature is essential in the VA overview design because it enables users to discover potential patterns and relationships between the cell graphs in overview instead of regarding them as isolated graphs. In the next chapter, we will see that many security events are visible in multiple cell graphs and can be detected more easily from the overview.

Though Figure 4.8(b) provides multiple colored points for comparison, it is not easy for human analysts to see the small-count points because most of them are together at the bottom of the graph, as the green points in Figure 4.8(b) are. Instead of just using a linear scale, users can choose a logarithmic scale to spread the points, as in Figure 4.8(c), to see the large-value points as well as the small-value points.

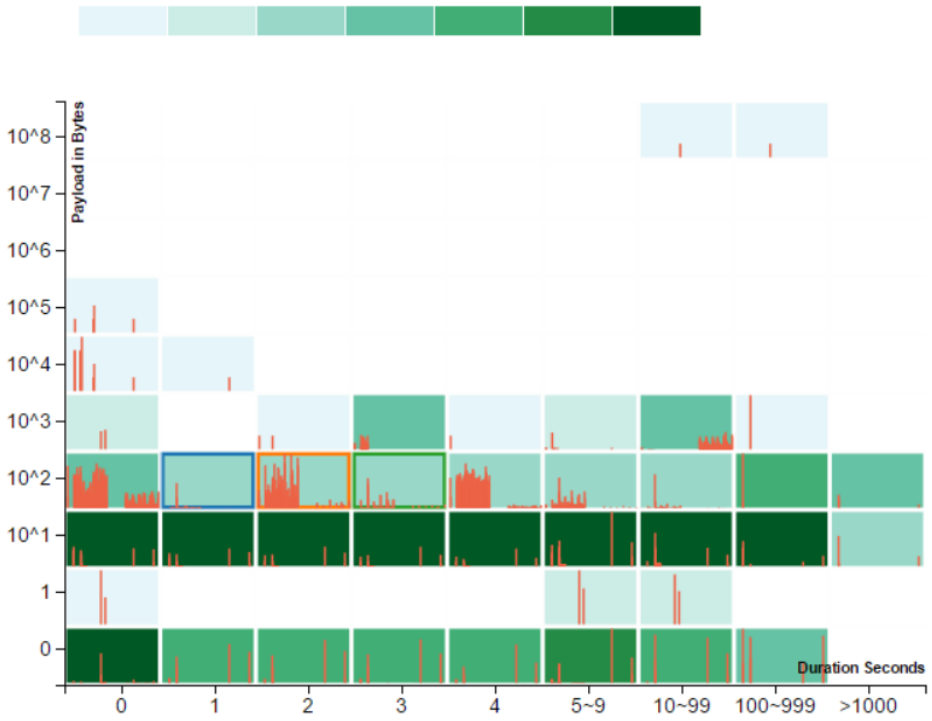

Figure 4.8(a) Duration-Payload Overview Design with three chosen cells

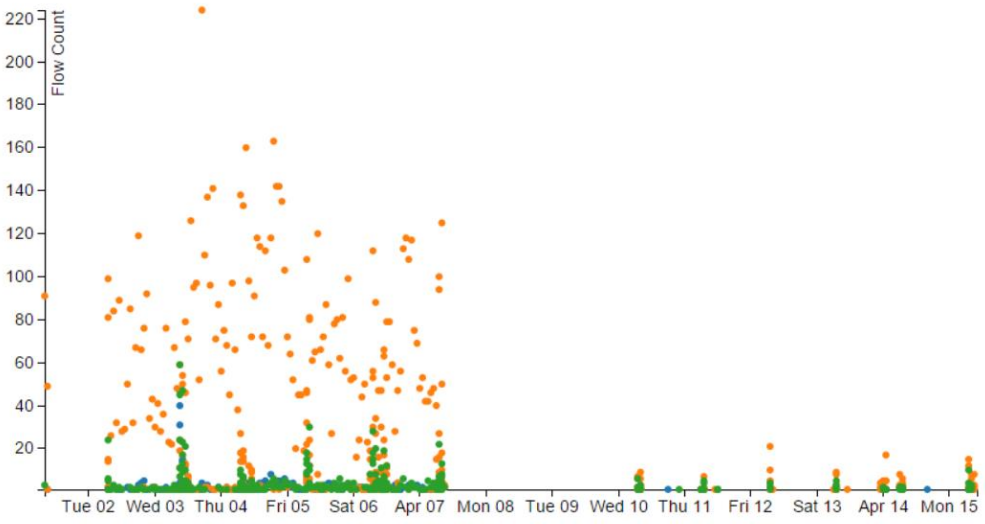

Figure 4.8(b) Scatter plot for the three selected cell graphs with linear scale

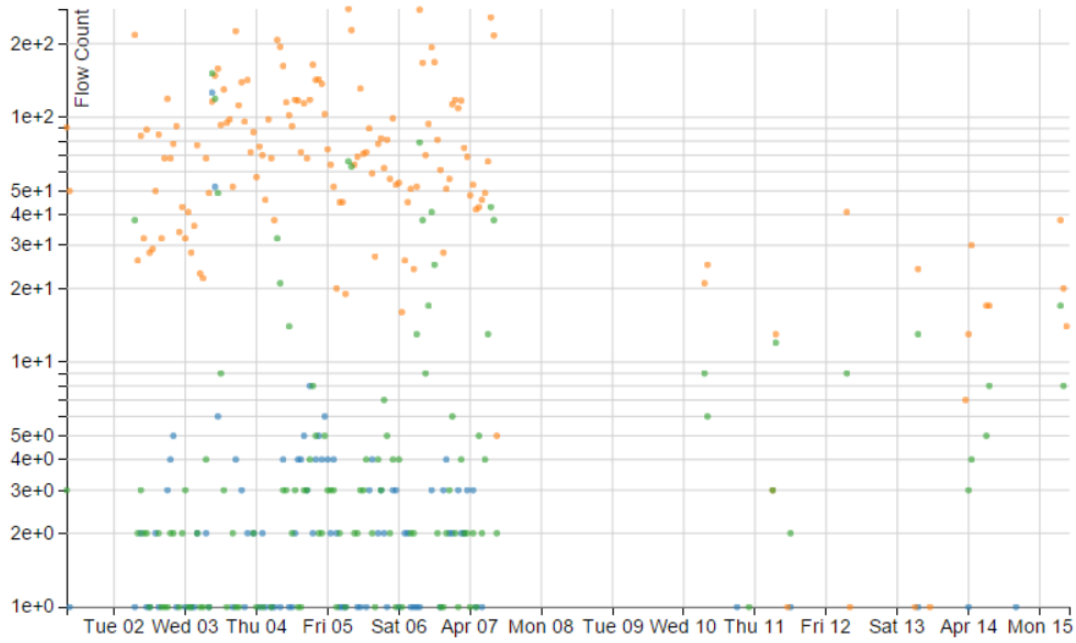

Figure 4.8(c) Scatter plot for the three selected cell graph with logarithmic scale

In the next chapter, we will discuss in detail how the Duration-Payload overview can effectively help human analysts identify network intrusions and provide situational awareness.

4.4 Host-Flow View

Performing visual analytics on a big data set is akin to using a microscope. When using a microscope, people generally find the area they are interested in under low magnification and investigate it further under high magnification. It is difficult to use high magnification directly. In performing visual analytics for network security, the

overview plays the role of low magnification. It identifies the interesting or suspicious time point and hosts, but it needs a means to investigate the events in detail.

The second level of visualization provides a zoomed-in view of the time series graph, but it is also necessary to visualize the content of NetFlow. This brings us to the third-level visualization, a Host-Flow view to provide information at the NetFlow level. When users select any point in the second level's scatter plots, the Host-Flow view provides the content of NetFlow related to the selected data point.

As seen in Figure 4.9, the most detailed level is a visualization of network traffic flow between two groups of IP addresses, generally one group of external hosts and one group of internal hosts. Parallel coordinates visualization is a popular approach to visualizing multidimensional information. NetFlow records contain multiple attributes, such as source IP address, source port number, destination IP address, destination port number, flow payload, and session duration. Each of these attributes is represented by a vertical coordinate in the graph.

A NetFlow data record will be visualized as a line, with corresponding intersections at the coordinates. The default attributes in a parallel coordinates visualization are source IP address, source port, destination port, destination IP address, source payload, destination payload, session duration, and timestamp. In practice, users can change the setup by adding or removing coordinates to satisfy their needs. The lines are colored in the Host-Flow view to provide better and clearer visual representation.

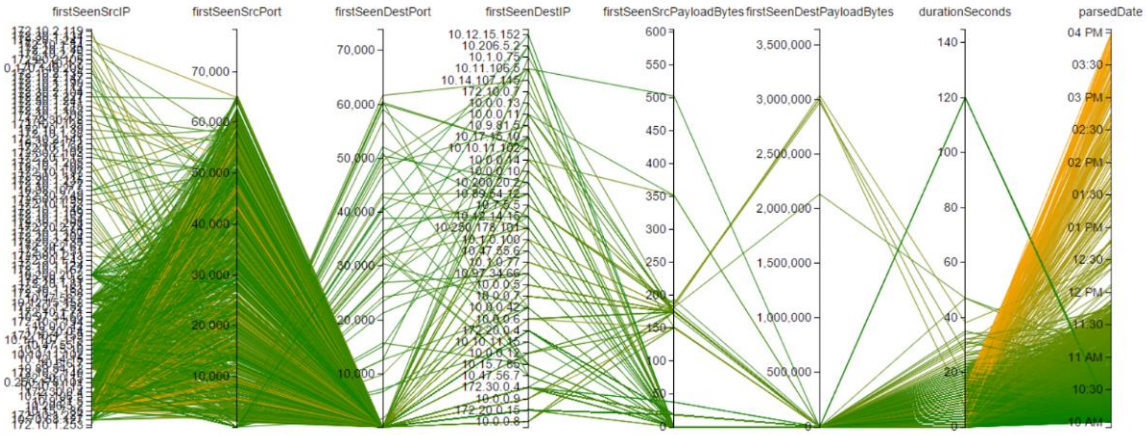

Figure 4.9 Parallel Coordinates visualization with 1000 NetFlow records

A potential problem for parallel coordinates visualization is the number of NetFlow records to be visualized in one graph. If the number is too high (1,000 or more), the visualization will be very difficult to analyze, as seen in Figure 4.9. Here we use a very simple approach to solve the problem. Starting with the data point in level 2, instead of visualizing all the NetFlow records, we randomly choose 100 records and use them in the parallel coordinates visualization. Can this sample represent the population? In most cases, especially in attack scenarios, the answer is yes. The primary reason is that the first two levels of visualization have separate NetFlow records in different groups, so the records for one data point in the second level (scatter plots) should exhibit uniform behavior to some degree. In some rare cases, such as in a distributed denial-of-service attack, some attackers may become invisible after visualization, but analyzing all NetFlow records to the internal targets will easily reveal the whole picture. In the next chapter, we will cover this topic in more detail.

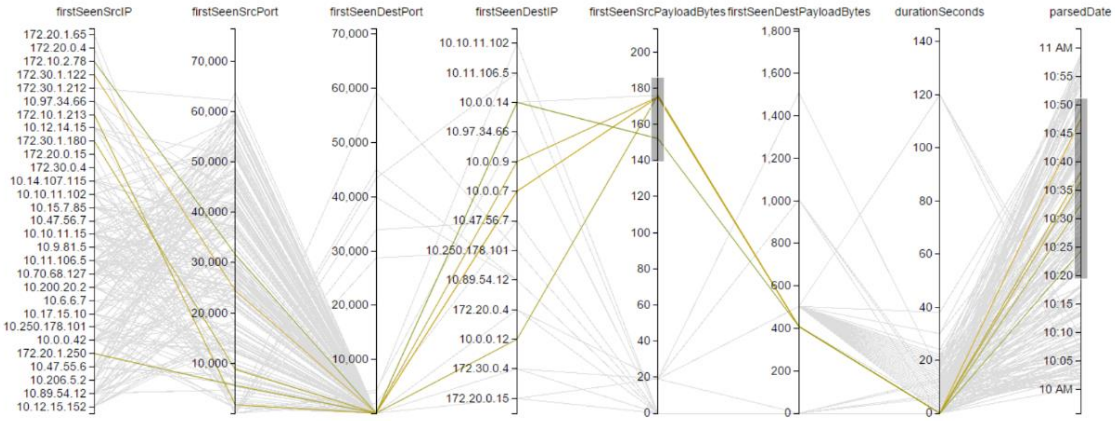

Figure 4.10 Parallel Coordinates visualization with selection feature

An important feature in Host-Flow view is the user-defined selection. Users can select any range of coordinates to highlight in NetFlow records, as in Figure 4.10. In this example, the user chose to investigate traffic from 10:20 a.m. to 10:50 a.m. with a source payload of 140~180 bytes. Any NetFlow records in those ranges is highlighted automatically.

In Host-Flow view, if two records have the same values for all attributes, they will merge into a single line and cannot be distinguished from each other. Therefore, the user can define a coordinate to represent the amount of NetFlow. The NetFlow count is important in analyzing potential security events, such as denial-of-service and port-scan attacks. For example, hundreds or thousands of packets sent to the server in a very short time may indicate a denial-of-service attack. If there are only a few NetFlow records, it is highly possible it is just normal web traffic. Therefore, NetFlow count can be very useful in some cases.

4.5 Host-MaxConn Overview

The Duration-Payload overview provides network traffic patterns for detecting intrusions, but it does not provide any information about the internal hosts at the overview level. A host-based (IP-based) visualization provides information or the status of each host of interest. We believe such visualization is important in the overview design because the status of internal hosts (such as important servers) is vital to network administrators for monitoring the network system.

Host-based visualization is suitable for detecting intrusions that target certain hosts because it treats every host separately. From the previous overview, some slow or subtle port scans cannot be seen easily. The primary characteristics of a port-scan attack are largely related to how many distinct internal “addresses” are accessed by an external host during a short time of period. The “address” here is actually representing IP address and port number because we’re dealing port scanning here. To detect port scanning, we visualize the characteristics of a port scan for each internal host. Similarly, such host-based visualization can be extended to denial of service, data exfiltration, or simply server status information such as response time and CPU usage.

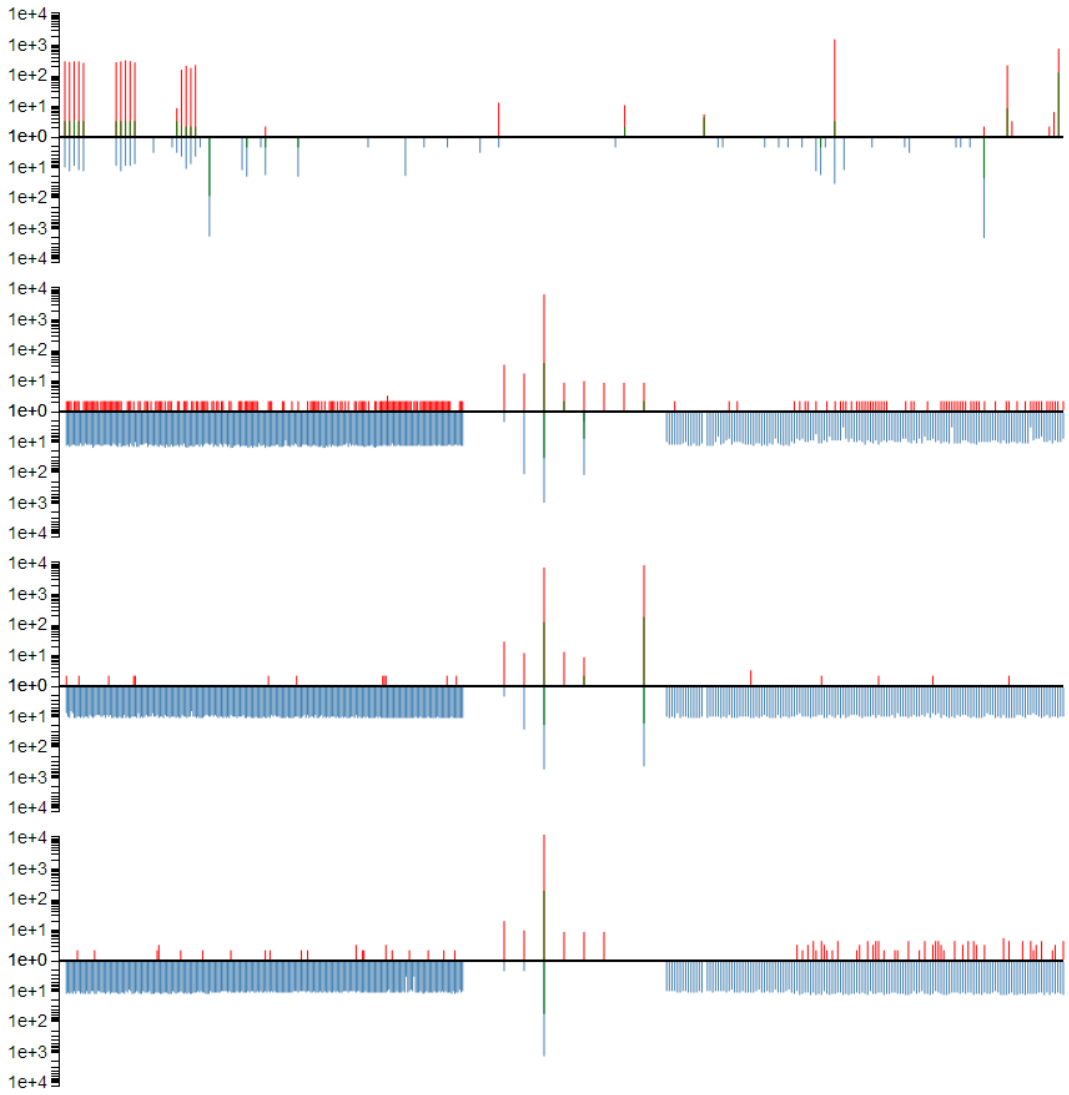

Figure 4.11 Host-MaxConn Overview

Because it is a host-based design, a potential problem is scalability. If an internal network has only a few hosts, visual representation of the hosts is relatively straightforward. But when the number increases beyond about 200, the visual design will be much more difficult. Our general idea is to highlight the more important servers

by allocating more space to them. Servers are the central part of enterprise networks, and we should pay more attention to them. For a typical enterprise network, even one with several thousand hosts, the number of servers is limited. For example, in the data set of VAST 2013, Big Marketing Network has approximately 1,200 hosts, and 22 of them are servers (web server, DNS server, e-mail server, and so on). In Figure 4.11, the very first chart represents external IP addresses, and the other three below denote Big Marketing Network's three internal network sites. In the three internal horizontal coordinates, IPs for the servers are placed in the middle and take more space; personal hosts are placed at the two tails of the coordinate evenly.

The bigger concern is the external IP visualization. In our design, one chart is used to visualize status of external IPs because attackers should be identified (at least partially) from the overview. However, theoretically, there are approximately 4 billion legitimate IP addresses (this is only for IPv4), and it is impossible to visualize them in any way. In reality, a significant amount of them are not active (in other words, they are not visiting any internal servers or reached by internal hosts). Therefore, in our design, we only visualized active external IP addresses and put them uniformly in the horizontal chart. In the case of Big Marketing Network, only about 200 active external hosts appear in the 2 weeks' data. If the number increases to a certain amount, external IPs will be grouped to decrease the number of display IPs. Therefore, even though this is a host-based design, it still has some scalability to easily handle a small or midsize network.

In Figure 4.11, the vertical axis represents the maximum connection rate for each IP address, with upward lines representing inbound and downward lines representing outbound. Connection rate represents how many distinct connections were built during a unit time, where connection is defined by IP address and port number. For a particular host, the value of the blue line represents the maximum number of distinct outbound connections the host has made, and the red line represents the maximum inbound connections. Host-MaxConn is used to find a host that has an uncommon connection rate. When a user clicks on a suspicious host, a corresponding Host-TimelineConn view provides time series information for the selected host, as in Figure 4.12.

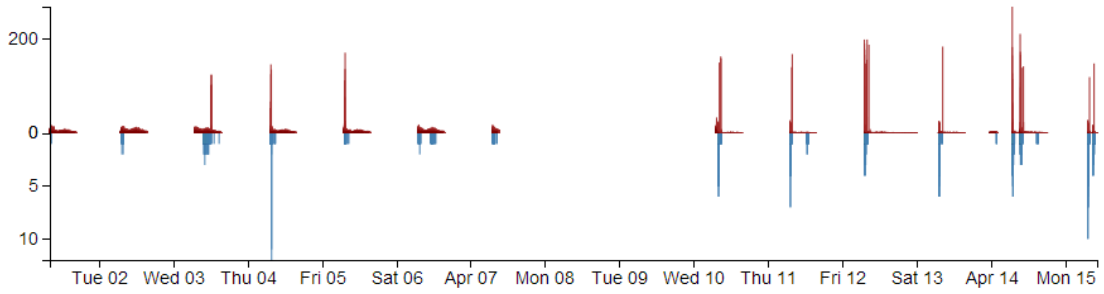

Figure 4.12 Host-TimelineConn view for one host

Host-TimelineConn visualizes time series data for a chosen host, instead of the maximum value. In Figure 4.12, it's easy to see that the X-axis represents time stamp, and Y-axis represent inbound and outbound connection rate. In order to show their own trending, these two time series are using different scales in Y-axis. This second-level

view provides trends and patterns for a host's connections, which could be very helpful when analyzing a host's activities. From our study, most of port scan attackers don't connect to the internal frequently, and the sudden increase of connection count can be an important characteristic of potential external attackers. Finally, we can also use Host-Flow view to analyze any data points of interest in a Host-TimelineConn visualization. Therefore, for the two overviews (Duration-Payload and Host-MaxConn), analysis procedures are similar: identify suspicious activities or hosts in the overviews, analyze timeline in the second-level visualization, and investigate security events in Host-Flow view.

A host-based overview can provide direct and precise information and status for each host of interest, allowing analysts to discover intrusions targeted at particular hosts. Even though here we use port scan as an example, the host-based overview can be also be used to detect other kinds of network intrusions and provide situational awareness information for important servers.

4.6 Chapter Summary

In this chapter, we primarily introduce and describe two types of overviews: Duratoin-Payload Overview and Host-MaxCoon Overview. We believe the former one is capable of analyzing network traffic patterns and trending, whereas the latter one is more suitable for providing information and status on per host basis. Other than the two overviews, there are several zoomed-in visualizations, to investigate any suspicious activates in details. Such multiple-level design offers scalability for the system, letting

human analysts to identify IP or time point at overview, and investigate further with zoomed-in views. In the next chapter, we'll describe how the VA system can be applied to detect different types of intrusions.

CHAPTER 5. SECURITY EVENTS ANALYSIS

Based on characteristics of NetFlow data and network intrusions, we have designed two overviews for the project: Duration-Payload and Host-MaxConn. The former is designed to present traffic patterns according to NetFlow's session duration and payload size, and the latter will display maximum incoming and outgoing connection rates on a per-host basis. In this chapter, we will discuss how to apply these two overviews to identify network intrusions, especially focusing on denial of service, port scanning, and data exfiltration. Multiple-level visualizations, which provide more detailed information, will also be used.

5.1 General Guides to Read Duration-Payload Overview

The Duration-Payload overview provides facts on the distribution of NetFlow records and timeline information. Moreover, it can help human analysts determine network status, such as incoming traffic patterns and trends. Thus, it also provides both situational awareness and intrusion detection alerts. Situational awareness is important for intrusion detection systems because even the best system cannot cover all types of attacks. Therefore, to detect possible abnormalities, we have to know what the network looks like normally.

As we discussed in the previous chapter, session duration and payload size are two important attributes in NetFlow. An unusually short duration and small payload generally indicate a port-scan or denial-of-service attack; while long duration and large payload mean a large file transfer, which is possible data exfiltration. It should be noted that there is no strict line for the duration length and payload size; they will vary for different networks. Overall, however, most NetFlow records with mid-range session duration and payload size are normal traffic. This conclusion is applicable only when we analyze NetFlow data. If we have the real content of the payload, that requires a totally different approach.

In Figure 5.1, the cell graphs in the blue rectangle are the “normal” zone, and we are assuming most of the records within it are normal traffic to servers. Apart from the mid-range criteria, an important characteristic of a normal cell graph is that it contains consistent and regular traffic data, which means if the NetFlow data in that range are normal, it should be visible many times during the 2 weeks, not just a burst of traffic or a significant isolated peak. Further study identified some NetFlow records related to intrusion in this normal zone, but most of them are regular network traffic according to the ground truth.

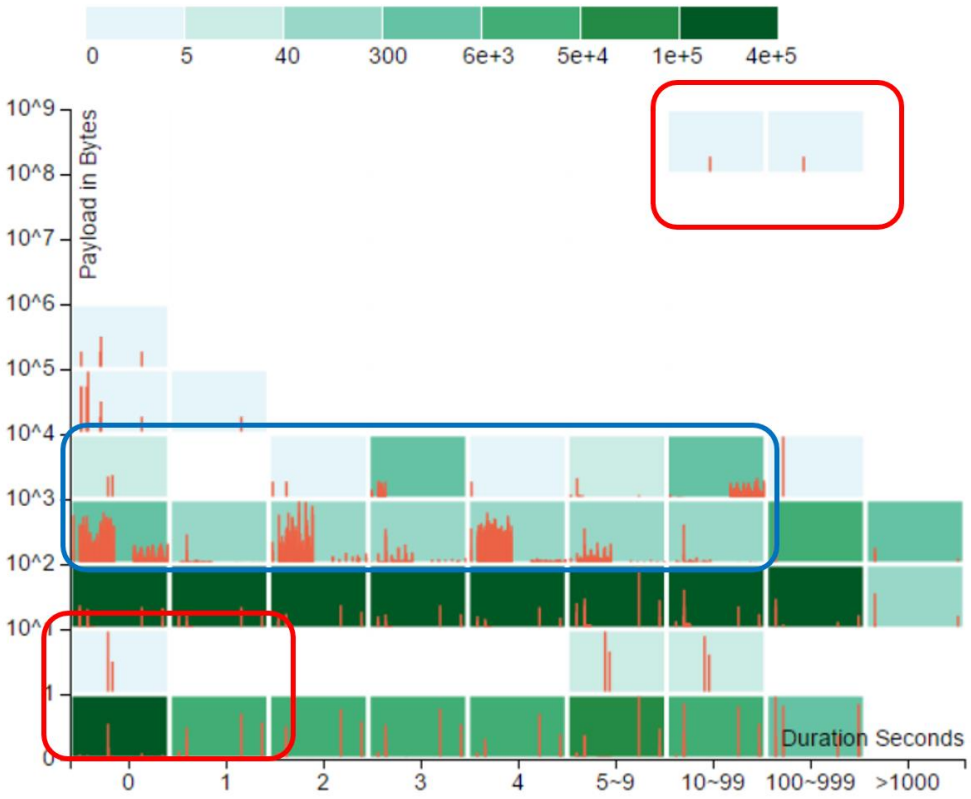

Figure 5.1 Duration-Payload Overview with normal zone (blue rectangle) and suspicious zones (red rectangles) highlighted

On the contrary, the bottom left corner with very short session duration and very small payload size is probably the result of denial-of-service or port-scan attacks, while the top right corner with very long session duration and large payload size may be related to data exfiltration. We will discuss the details of these graph cells in the following sections. From the graph, it is clear that many of those cells contain isolated peaks, which means that normally there are no NetFlow records in that range, and the peaks may indicate unexpected security events.

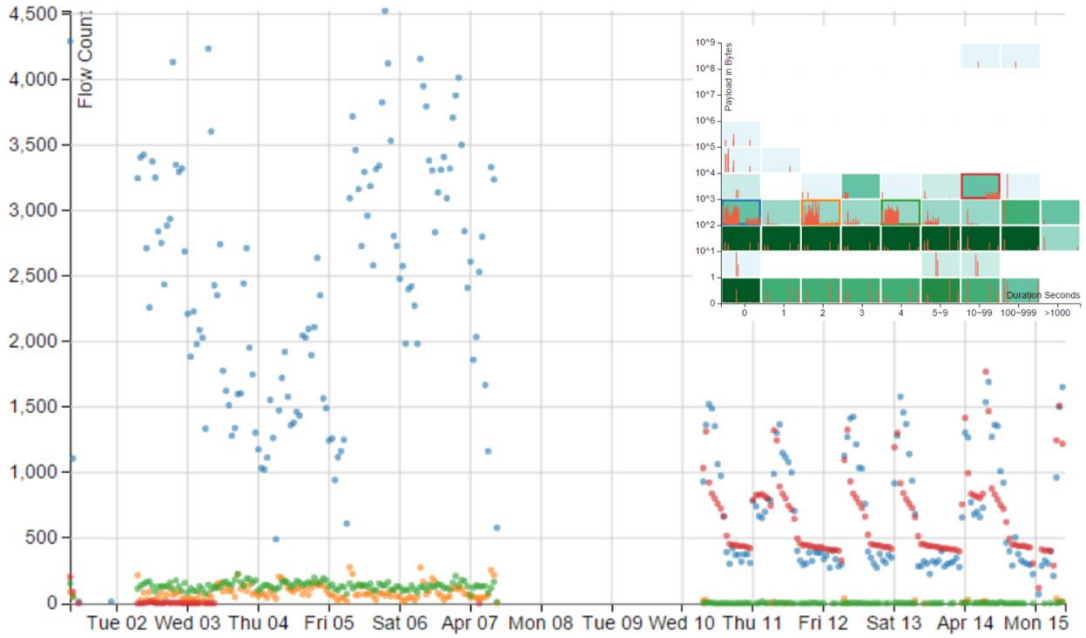

Figure 5.2 A zoomed-in scatter plot for “normal” activities

A Duration-Payload overview also provides situational awareness to some degree. From the overview in Figure 5.1, a careful investigation indicates that all the cell graphs have a blank period during the 2 weeks. When we zoom in to the scatter plot in Figure 5.2, from the afternoon of April 7 to the end of April 10, there is no sign of any server activity. The ground truth states that Big Marketing Network had network maintenance during the weekend, so no servers were running during that period. From this example, it is clear that the overview can provide network status in addition to detecting intrusions.

Another example usage of Figure 5.2 is to analyze trends in server access. In Figure 5.2, the red dots in the second week have an obvious pattern. The red dots peak at approximately 8 a.m. and then consistently decrease to a stable level, about 500.

Further analysis from the host-flow view indicates that traffic related to red dots is actually mail service. As a result, we have a general knowledge of the users' pattern of accessing mail service in Big Marketing Network.

5.2 Denial of Service Attack Analysis

A denial-of-service attack is generally relatively easy to detect because it is usually related to a large number of NetFlow records. An unexpected burst of incoming network traffic may indicate a denial-of-service attack on internal servers. Therefore, the number of NetFlow records is a primary characteristic for identifying denial-of-service attacks. Furthermore, attackers usually use very little or no payload for attack in order to use network bandwidth to send more packets. Thus, a small request payload is another important characteristic of a denial-of-service attack. In Figure 5.3, the blue rectangular zone contains NetFlow records potentially related to denial-of-service attacks for Big Marketing Network.

Because background color in the cell graphs represents the scale of the y-axis, we should start with the darkest green to find the highest traffic rate. In the blue rectangle in Figure 5.3, many of the cell graphs are the darkest green, indicating that their NetFlow amounts are very large. The payloads are all under 100 bytes, but the session durations vary from 0 to over 100 seconds.

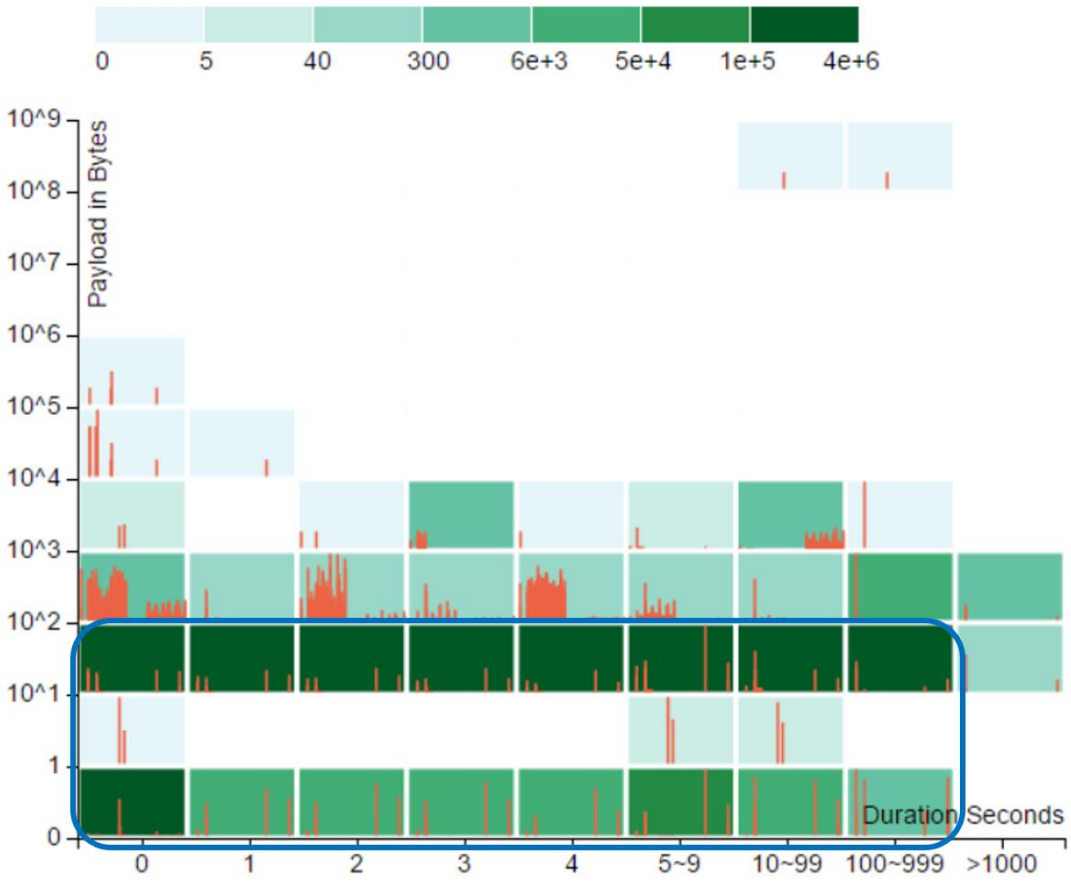

Figure 5.3 Duration-Payload overview with DOS attack zone

Because denial-of-service attacks usually come with a significant traffic burst to one or many internal servers, we should focus on peaks from the overview. With the help of a zoomed-in scatter plot, Figures 5.4(a) and 5.4(b) display the NetFlow records potentially related to denial-of-service attacks. Figure 5.4(a) highlights cell graphs with the largest y-scale, and Figure 5.4(b) shows the corresponding records count per hour. The colors for highlighting in Figure 5.4(a) correspond to the colors in Figure 5.4(b).

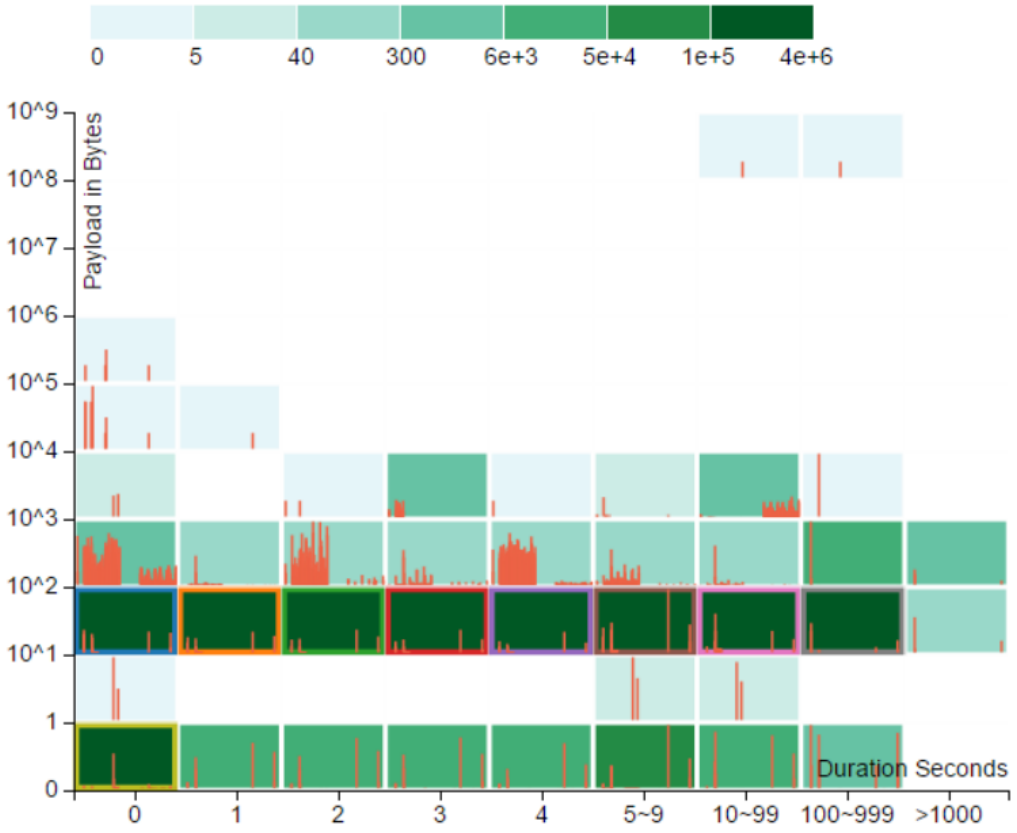

Figure 5.4(a) Duration-Payload overview with nine cell graphs highlighted

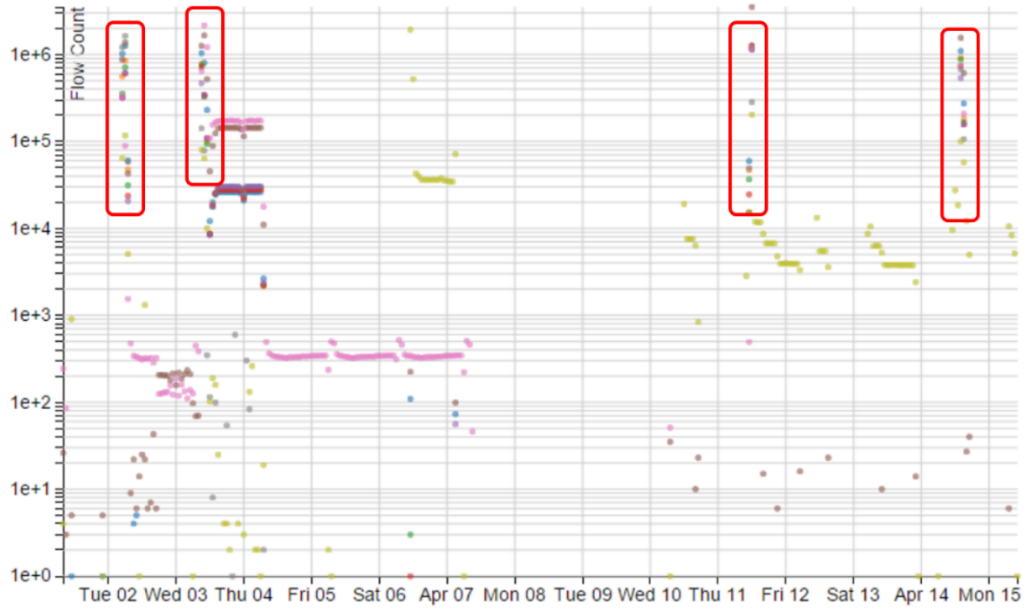

Figure 5.4(b) Scatter plot with four potential DOS events highlighted

In Figure 5.4(b), there are four obvious peaks formed from points with different colors, all marked with red rectangles. Potentially, these four events are denial-of-service attacks because the flow count reaches 1 million per hour, which is significantly larger than normal. Further study showed that the first two events are indeed denial-of-service attacks, but the last two events were port scanning. In this case, these two port-scan attacks involved an unusually large amount of traffic, and we will cover this more in the port scan section.

After identifying the suspicious events, we can use host-flow view to further investigate the first two denial-of-service attacks; Figure 5.5(a) displays a data point from the first event on April 2. There are 10 external hosts (in the first coordinate) using different ports to target port 80 of internal web server 172.30.0.4. In a sense, this is a distributed denial-of-service attack because there are multiple external attackers. Further investigation of other data points in the same region indicated that this attack was targeting only one web server (172.30.0.4), indicating that there are over 10 million records (sum of the data point amount in Figure 5.4[b]) associated with this web server during 1 hour or so. We can use a user-defined host-flow view to investigate this server's behavior during the attack. In this step, we can remove attributes such as source port, payload, and session duration and add a count coordinate to represent the number of NetFlow records. This will help clarify the web server's activity during the attack period.

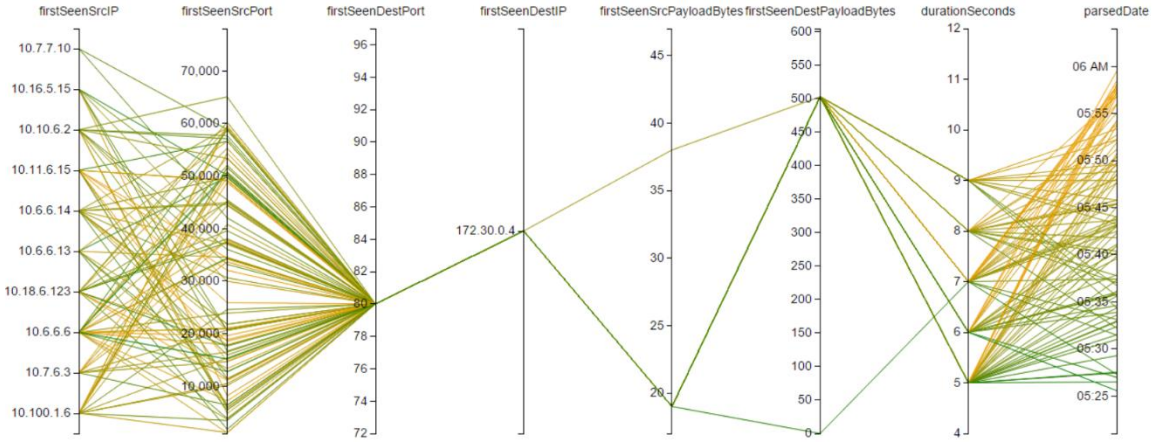

Figure 5.5(a) Host-Flow view analysis for a data point

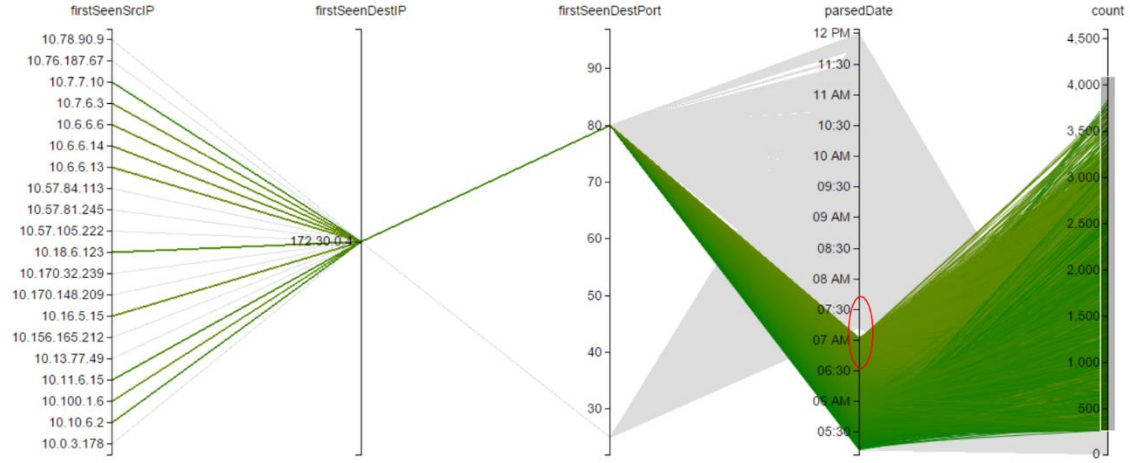

Figure 5.5(b) Host-Flow view analysis for activities of web server 172.30.0.4

Figure 5.5(b) depicts the activities of web server 172.30.0.4 on April 2 and highlights the high-count NetFlow records. These highlighted records are related to the denial-of-service attack. From the first coordinate, which represents external IPs, we can see that there are a total of 10 external hosts involved and that the attack started at

approximately 5:20 a.m. and ended at 7:00 a.m. At approximately 7:02 a.m., there is a 10-minute gap with no records, marked by a red circle in the figure. This is possibly the result of a denial-of-service attack that temporarily crashed the web server. After 7:15 a.m., the server was back to normal and the denial-of-service attack was over.

The analysis above shows that denial-of-service attacks are relatively easy to detect primarily because of the large amount of traffic. A Duration-Payload overview and scatter plot can be combined to identify the attacks, while a host-flow view provides a clear timeline for the attack. Human analysts can identify the external attacker, the internal victim, when the attack happened, and the consequences through a multiple-level visualization.

5.3 Server Redirection Analysis

Compared to denial-of-service attacks, a server-redirection attack is usually more difficult to detect. There are generally two phases in a server-redirection attack: hacking and redirection. In the first phase, servers are hacked by external attackers through a virus or malicious code injection. This is usually very difficult to discover based solely on NetFlow data because the corresponding packets may have normal payload and duration attributes. In the second phase, redirection, the primary characteristics are unusually short session duration and small payload. It is difficult to define a “usual” range clearly because the attributes vary for different hosts and servers. Moreover, many complex websites are hosted on multiple servers, and many automatic redirections are benign and should be distinguished from malicious ones.

For normal web servers, session lengths should follow a relatively consistent pattern. Session lengths vary, but there should be certain values that are distributed in a range. When a server is hacked, however, the incoming traffic will usually be redirected to another external malicious website after a short time.

Therefore, the average duration will decrease significantly, and the payload will normally decrease because users do not send further requests (packets) to servers after redirection.

This analysis suggests that each cell graph in the Duration-Payload overview should have relatively consistent traffic. Variation is acceptable as long as it is within a reasonable range; users' visits may vary. If there is a significant traffic gap, or if there is noticeable traffic some of the time but none at all at other times, it is reasonable to assume that something happened to the servers.

It is possible that the servers are configured differently (such as to host other websites) or have been hacked, but otherwise a server's NetFlow data and behavior cannot change significantly during two weeks or a short period of time. For example, in Figure 5.6, the cell graph in the blue rectangle shows significant traffic in the first few days but no sign of traffic afterwards.

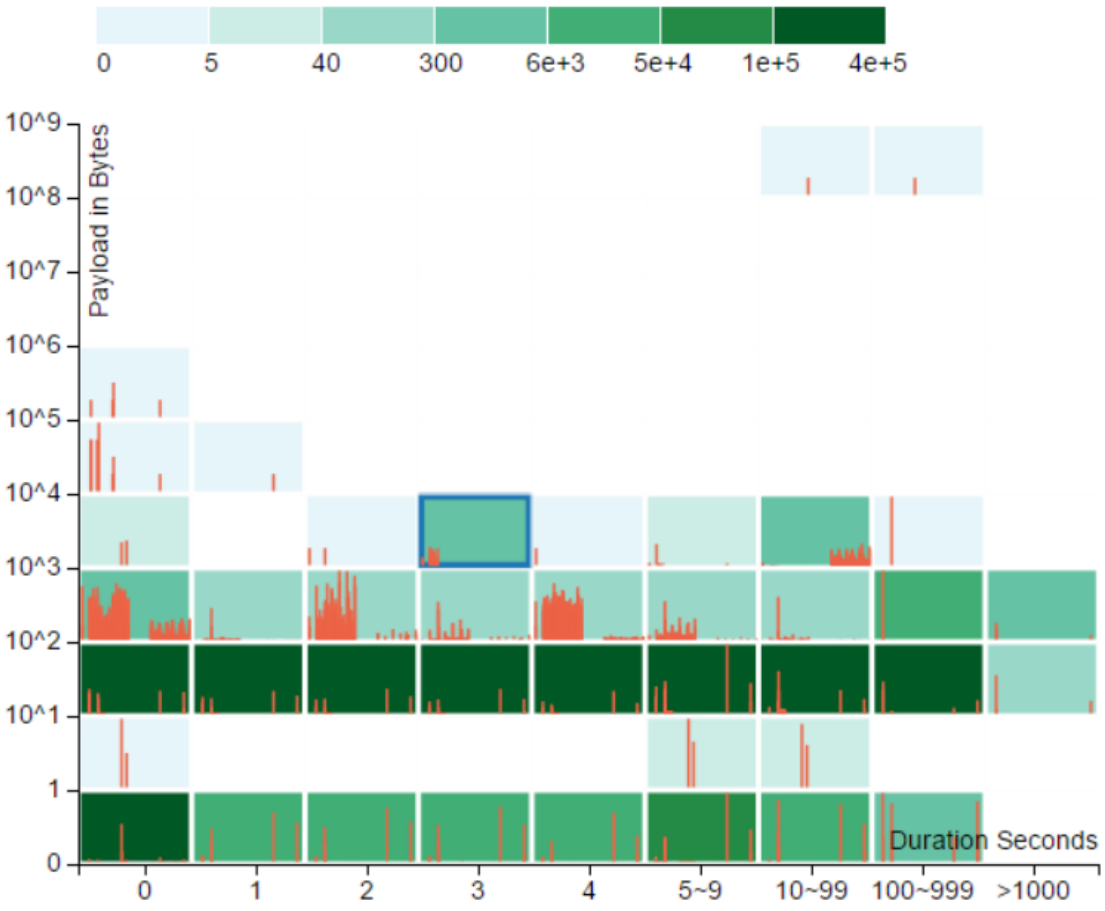

Figure 5.6 Duration-Payload Overview with a cell graph highlighted

The activities in blue rectangle of Figure 5.6 are suspicious because traffic distribution for normal-running servers does not usually change all that much. Inspecting the NetFlow records shows that it is all incoming traffic to web server 172.20.0.4. Because the overview allows us to check the Duration-Payload distribution for a specific server, we can take a closer look at the NetFlow records for server 172.20.0.4, as seen in Figure 5.7(a) and Figure 5.7(b). It should be noticed that Figure 5.6 and Figure 5.7(a) are very similar, but 5.7(a) only visualized data for one server.

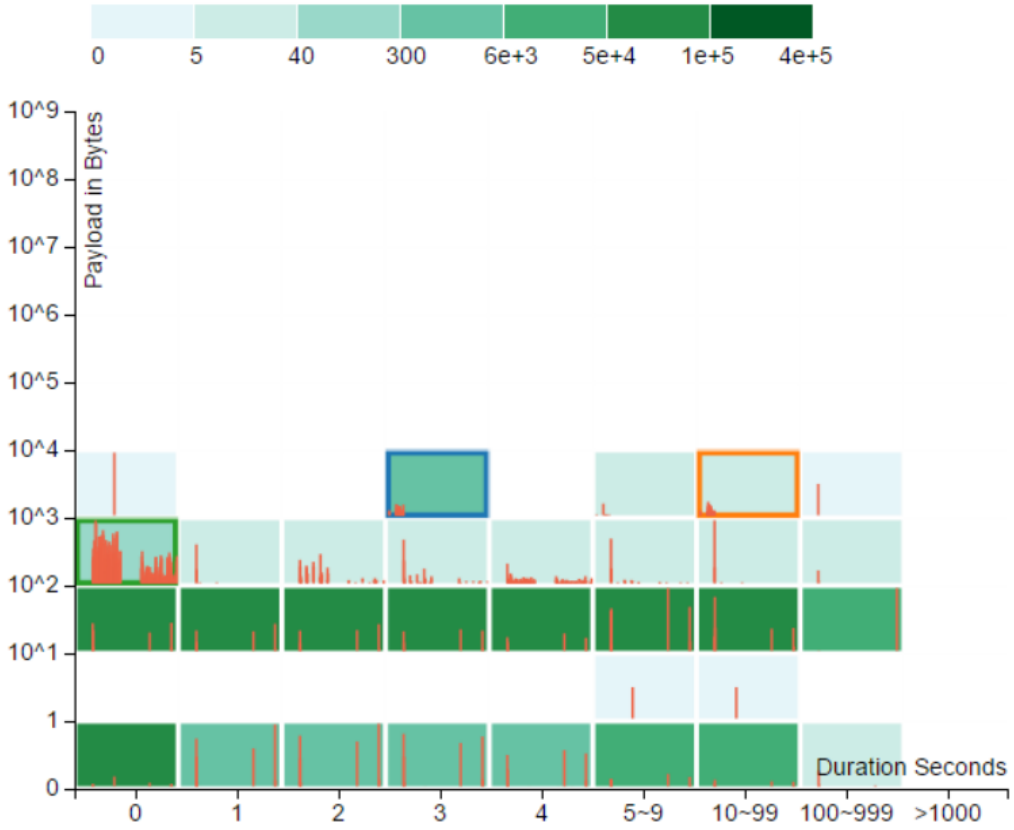

Figure 5.7(a) Duration-Payload Overview for web server 172.20.0.4

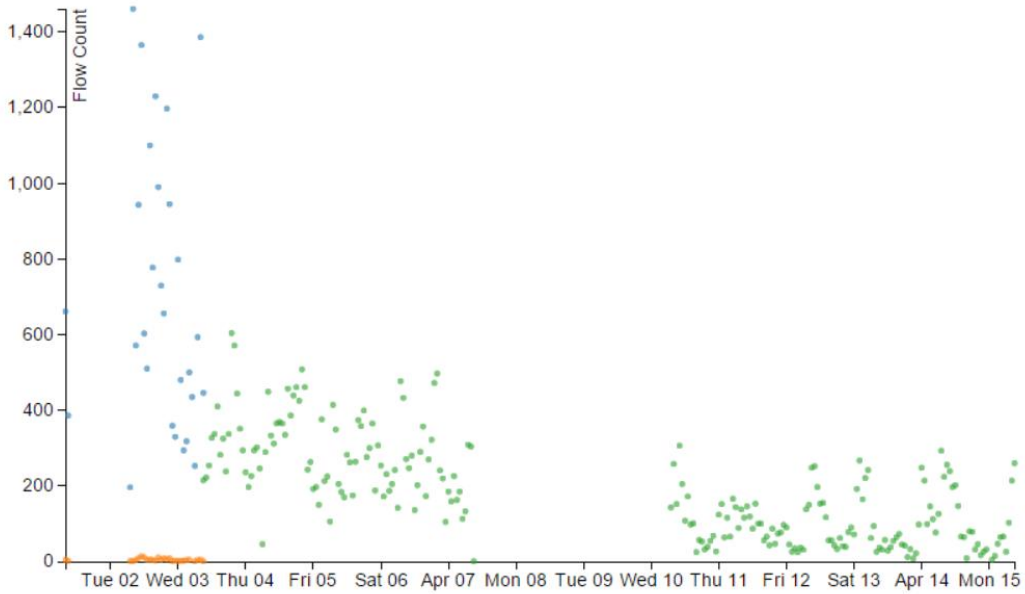

Figure 5.7(b) Scatter plot for web server 172.20.0.4

Web server 172.20.0.4 clearly exhibits a significant traffic shift from the blue and orange rectangles to the green rectangle. Here we ignore the high peaks in the darker green cell graphs because they usually indicate denial-of-service or port-scan attacks. To clearly present the shift, we can put data from those three cell graphs into a second-layer visualization, as seen in Figure 5.7(b).

It is easy to see that on April 3, incoming traffic undergoes a very suspicious change: The session duration significantly decreases to 0 seconds, and incoming payload decreases as well. What does a 0-second duration mean? We know that NetFlow combines a series of packets between two hosts, and 0-second duration means that at most a couple of packets are transferred between the hosts before the connection is terminated. To further investigate the changing NetFlow records, Figures 5.8(a) and 5.8(b) shows NetFlow data; 5.8(a) is from a blue dot in 5.7(b), and 5.8(b) is from a green dot in 5.7(b).

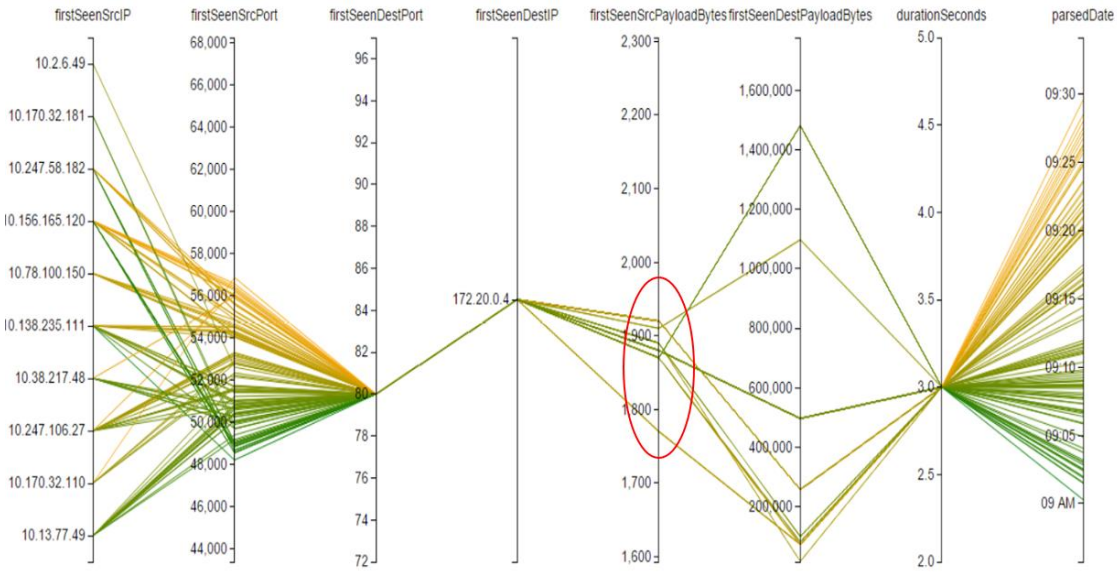

Figure 5.8(a) Host-Flow view for server 172.20.0.4 before redirection

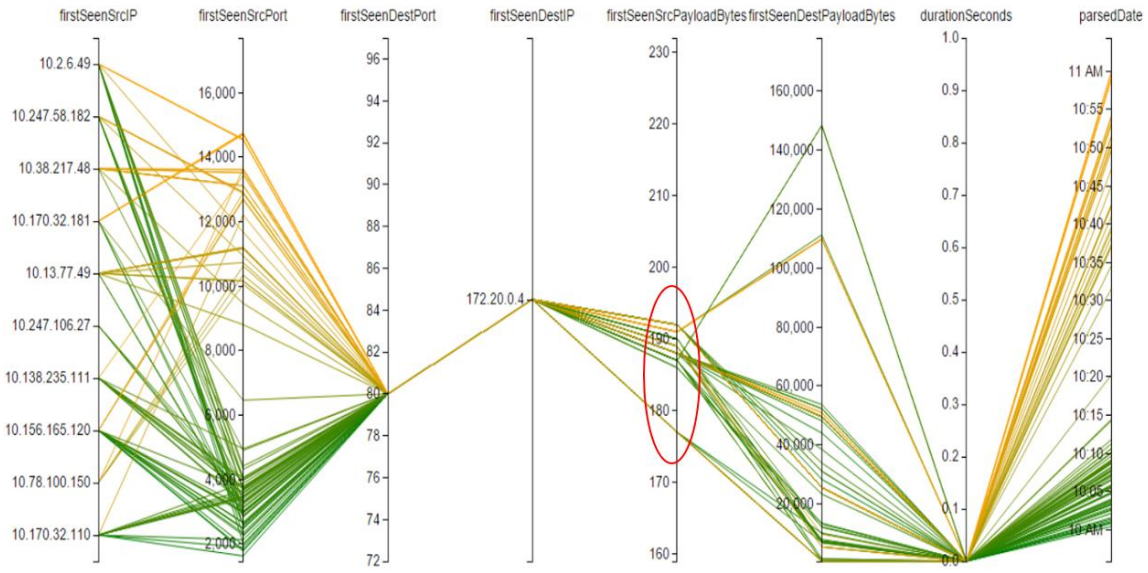

Figure 5.8(b) Host-Flow view for server 172.20.0.4 after redirection

As seen in Figures 5.8(a) and 5.8(b), source payload size (incoming traffic to servers) of NetFlow decreases from 1,900 bytes to 190 bytes, and session duration declines from 3 seconds to 0 seconds. The payload size decrease is significant, but the session duration decrease is not so obvious because it is only 3 seconds. Such consistent changes still call for careful investigation, however.

It is difficult to explain the shift if it is not the result of a server reconfiguration or a redirection attack. We can see that the shift happened at around 10:00 a.m. on April 3, but it is difficult to confirm when the server was infected with malware because attackers can initiate the redirection any time after infection, and the NetFlow that contains malware cannot be detected without the payload. Ground truth confirms that the hacking phase cannot be seen from the NetFlow data.

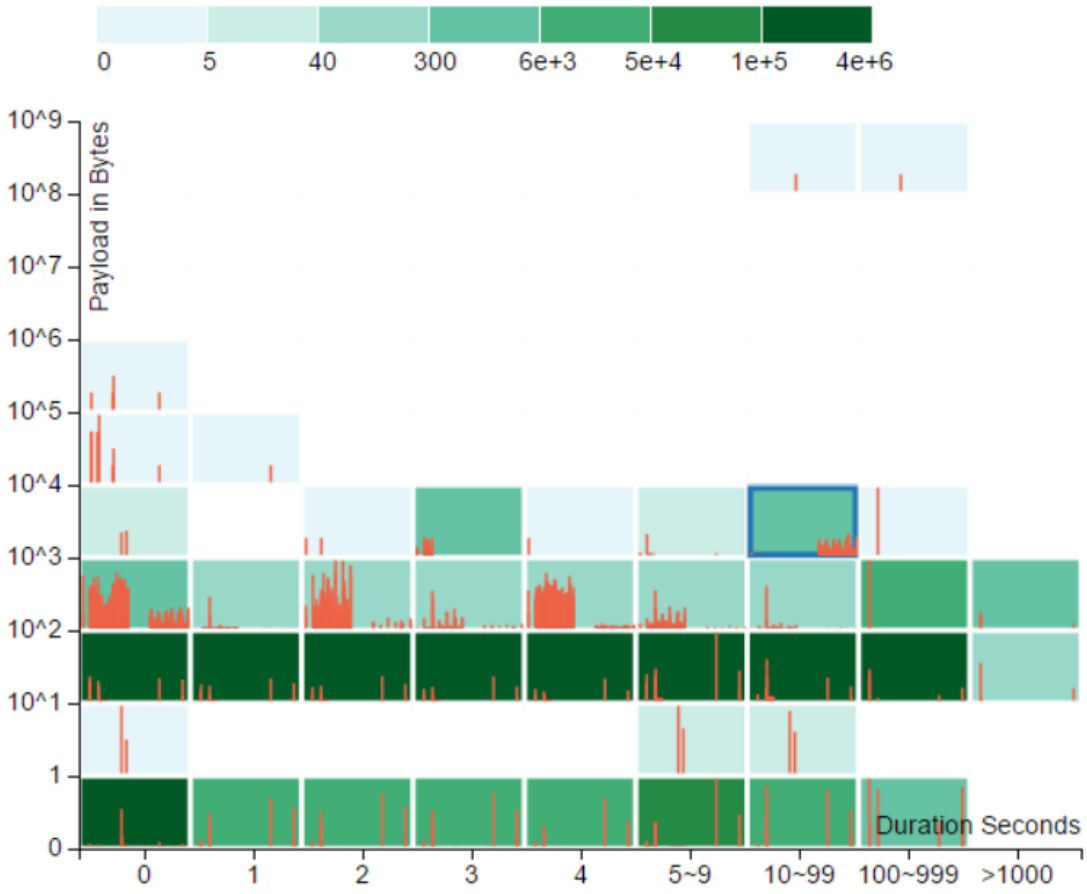

Figure 5.9 Duration-Payload Overview with one cell graph highlighted

From the analysis, we can see that server redirection can be identified by short session duration and small payload, but a traffic pattern shift may not necessarily indicate an attack. For example, in Figure 5.9, a cell graph is highlighted in a blue rectangle because data are shown in the second week but that cell is completely blank in the first week. Such a situation usually requires attention because it indicates the server's traffic pattern changed significantly during the 2 weeks. Further investigation

reveals that three mail servers in Big Marketing Network are responsible for the pattern shift. Our example is mail server 172.20.0.3, seen in Figure 5.10.

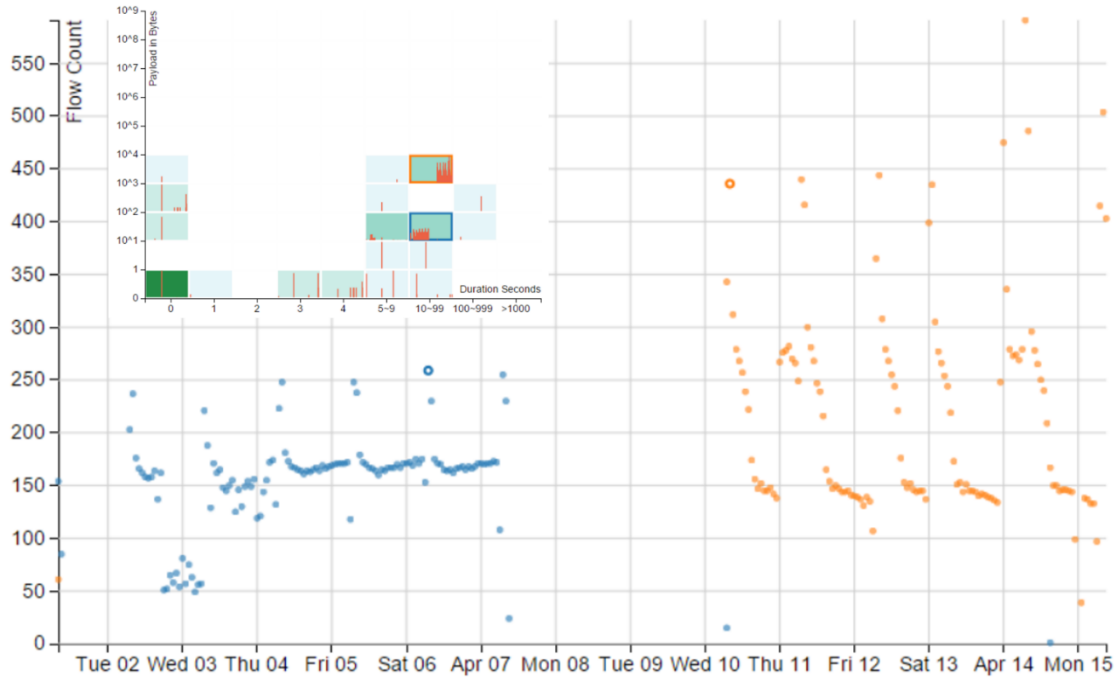

Figure 5.10 Duration-Payload overview and scatter plot for mail sever 172.20.0.3

In Figure 5.10, the small graph in the top left corner is the Duration-Payload overview for mail server 172.20.0.3. Two cell graphs, highlighted in blue and orange, clearly contain most of the normal traffic. We ignore other cell graphs with peaks here, because we know those peaks are produced by denial-of-service or port-scan attacks. In Figure 5.10, the blue points represent NetFlow records with a payload of less than 100 bytes, and the orange points represent NetFlow records with a payload of more than 1,000 bytes. Though the session duration for the two groups is the same, the payload

alteration is still relatively large. Nevertheless, there are two indications that this is not an attack.

First, despite the blue and red points, their distribution patterns during a day are very similar: high peaks in the morning that decline to a relatively consistent level. This corresponds to people's behavior. We check our e-mail when we start to work in the morning. Because usually some e-mail was received the previous night, the number of e-mails we need to send in the morning is relatively high. During the day, we frequently check e-mail or send them to others, and overall the traffic to mail servers will be at a consistent level.

Second, the changing point is on April 8 and April 9, when we know server maintenance took place and Big Marketing Network was down. Therefore, it is very possible that the administrators reconfigured the servers during that time, resulting in a shift in traffic. Based on these two observations, it is highly possible this traffic shift for mail server 172.20.0.3 is normal. Ground truth does not list any related attacks to this server.

5.4 Data Exfiltration Analysis

In the Duration-Payload overview, a very suspicious zone for data exfiltration attacks is the top right corner; those cell graphs show relatively long sessions and large payloads. These two characteristics give us the main way to distinguish a data exfiltration attack because real content of the payload is not available in NetFlow data.

In data exfiltration, an attacker will generally steal the information through a web or file transfer protocol (FTP) tunnel. Because NetFlow data are generated from a series of packets between two machines, a data exfiltration attack will normally generate a record with large payload and long session duration. If an attacker transferred only minimal data, it would be almost impossible to detect using NetFlow data because the data exfiltration record would be the same as that of other normal records.

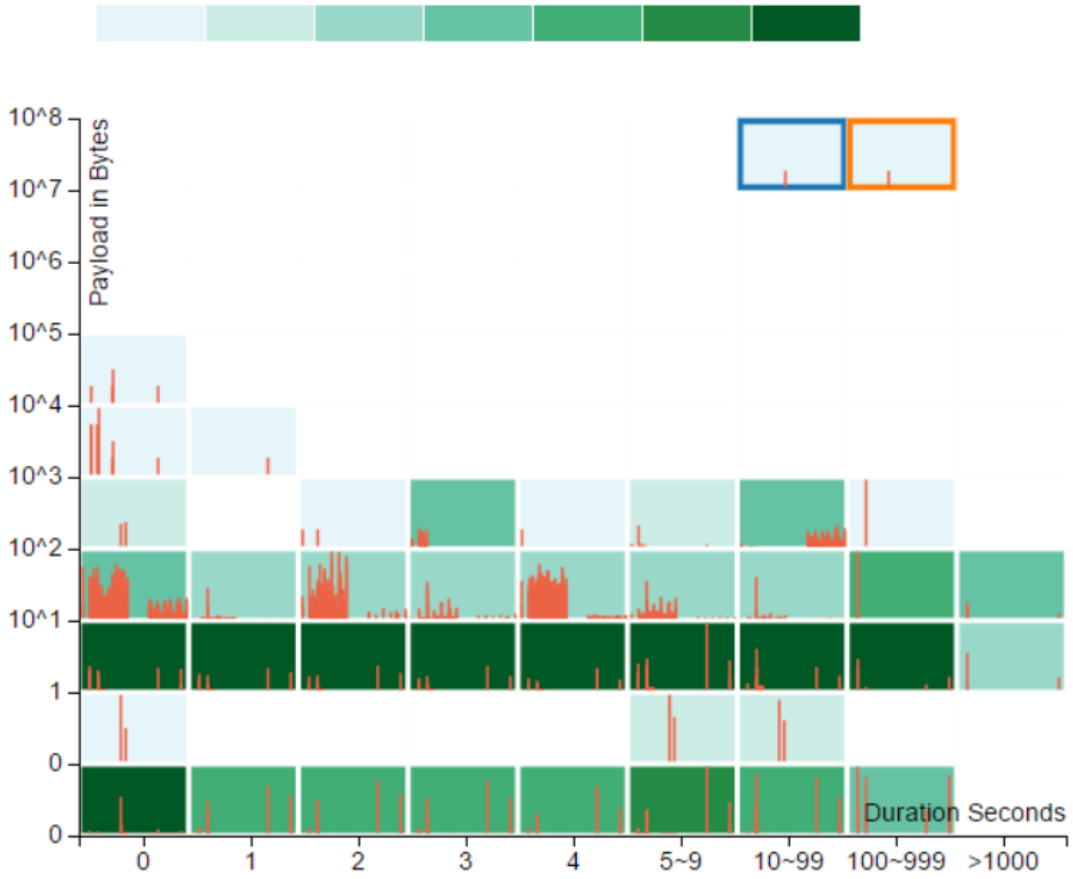

Figure 5.11(a) Duration-Payload Overview with two suspicious data exfiltration events highlighted

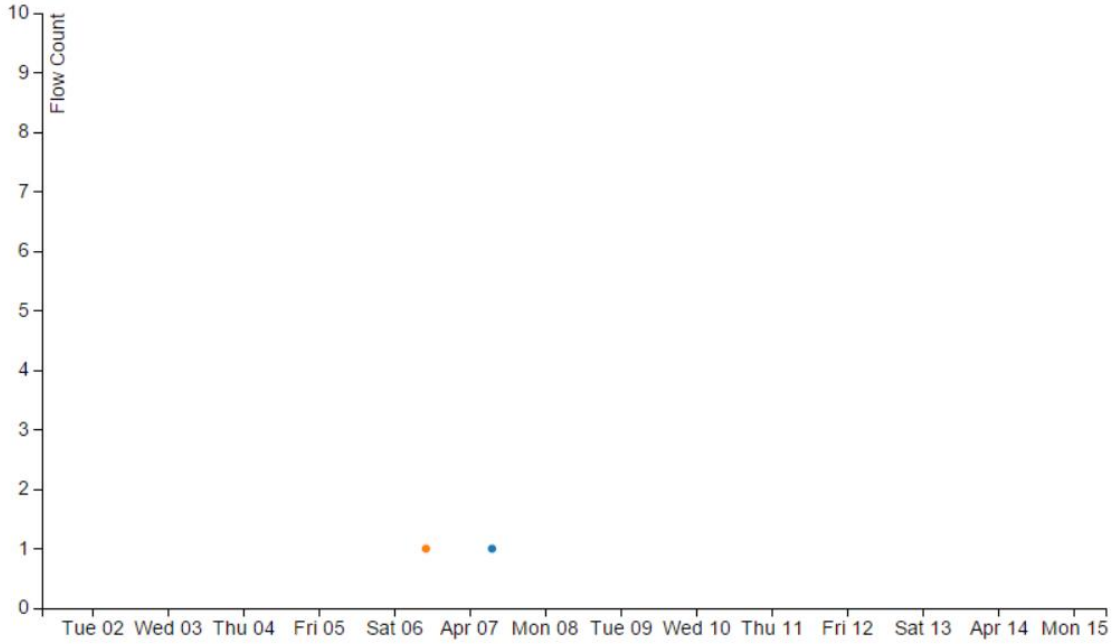

Figure 5.11(b) Second level visualization: enlarged time series for the two NetFlow

In Figure 5.11(a), two cell graphs with records are in the top right corner, significantly away from the others. Their session lengths are in the ranges 10–99 and 100–999 seconds, and, notably, the payload sizes are both in the order of 10^8 bytes, significantly larger than any other. Figure 5.11(b) confirms that there are only two NetFlow records, occurred in Apr 6th and Apr 7th respectively.

The parallel coordinates in Figure 5.12(a) provide more detail. For these two events, the attacker is external host IP 10.7.5.5 and the victim is web server 172.10.0.40. The source ports for these two events are both 20, indicating they are using FTP. Port 21 on the external FTP server, which is used to establish the connection between two hosts. The red circles mark the payload: One is about 100 million bytes, and the other is 600

million bytes. The payload is actually from the external FTP server to the internal host, but here we still consider the two events to be data exfiltration.

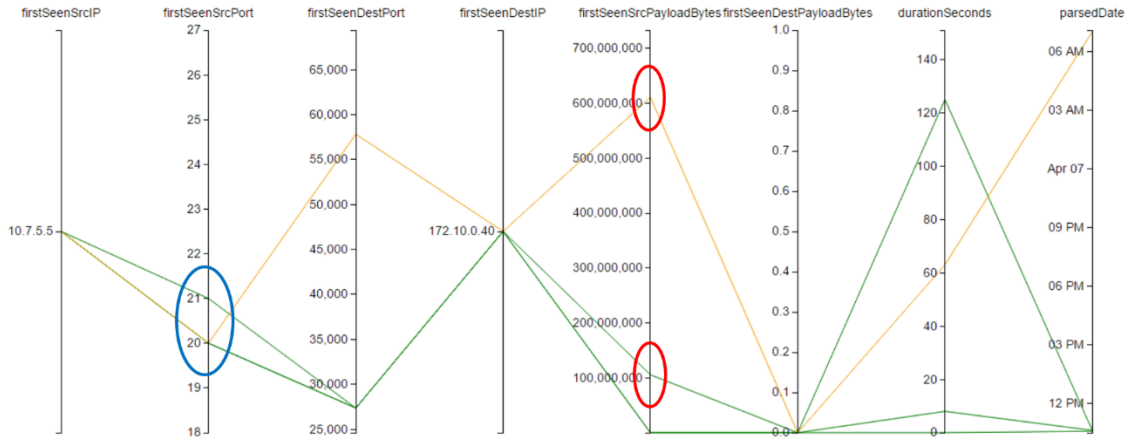

Figure 5.12(a) Host-Flow for the suspicious NetFlow record

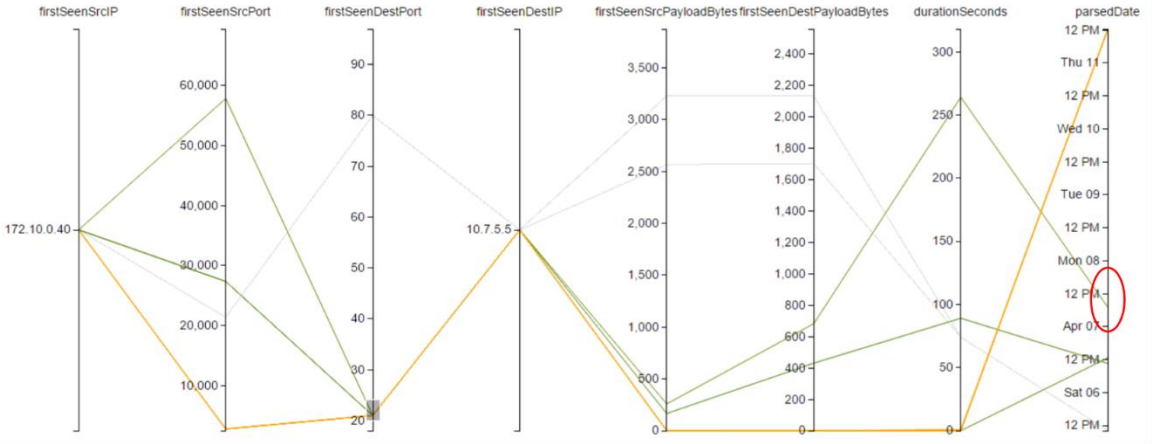

Figure 5.12(b) Host-flow between FTP server and internal host

In Figure 5.12(a), we see that the second data exfiltration event (orange line) uses only port 20, which is not supposed to be the case because port 21 is needed to

establish the connection. We can use a host-flow view to investigate further by setting 10.7.5.5 as the source IP and 172.10.0.40 as the destination IP. With NetFlow records, sometimes there is a mix-up between the source IP and the destination IP because the traffic collector does not catch the first packet (this is why they are called firstSeenSrcIP and firstSeenDestIP, and this is inevitable in NetFlow).

In Figure 5.12(b), NetFlow records related to port 21 are highlighted. On April 7, there is a record from internal host 172.10.0.40 to the external FTP server; the timestamp is marked in the red circle. This establishes the connection for the second data exfiltration, which is missing in Figure 2.11. We have the whole picture here for the two data exfiltration events.

April 11 is suspicious because the internal host seems to have been trying to connect to the FTP server again (orange line in Figure 2.12), but there is no significant payload this time (actually, it is zero payload). It is possible the attacker was trying to pull off a third data exfiltration but failed somehow. From this example, we can see that the host-flow view can be used to investigate any events of interest, providing the timeline for analysis.

5.5 Chapter Summary

In this chapter, we use the VA system to identify security events in 2 weeks of VAST data. First, Duration-Payload overview and Host-Maxconn overview can be used to highlight NetFlow traffic and hosts related to attacks. Zoomed-in timeline visualizations provide more insight. Finally, Host-Flow views are capable of displaying very detailed

information. The multiple-level VA system is able to not only discover security intrusions but also provide timelines for these events for further investigation. We have successfully identified denial of service, data exfiltration, server redirection, and port scanning through the VA system. In the next chapter, the VA system will be evaluated systematically, and results will be presented and discussed.

CHAPTER 6. SYSTEM EVALUATION

In the previous chapter, we described how the VA system can be used to detect network intrusions and abnormalities in detail. Furthermore, the VA system can provide situational awareness for the network by presenting and analyzing traffic patterns. To prove that the VA system is capable of detecting attacks effectively, however, a thorough evaluation is necessary.

In this chapter, we will present a systematic evaluation of the VA system. The ground truth for the Big Marketing Network data will be the primary criteria for the evaluation process, such as calculating false positives and false negatives in the VA system. It should be noted, however, that the ground truth lists only network intrusions but that the VA system can also highlight and identify abnormalities in the network that are not necessarily related to attacks.

Because overview is the most important feature in the VA system, the evaluation process will focus primarily on overviews. In other words, for the 2 weeks of data, we are primarily interested in how the overviews can help detect intrusions and what potential issues may arise.

6.1 Background and Attack Traffic Analysis

When evaluating intrusion detection systems, we need to have more insight into the size of normal and malicious network traffic. For example, if there are 100 NetFlow records and 10 of them are malicious records, it is relatively easy to target the records related to intrusions. When the traffic size increases to 10,000 records, however, finding the corresponding 10 records is rather difficult. Therefore, background and attack traffic analysis is presented here.

For data exfiltration, there are two intrusion records, and Table 6.1 displays the background network statistics for when data exfiltration happened. Here we can see that during the two data exfiltration events, the background traffic rate is rather high compared to the attack traffic rate. Moreover, the durations are only 125 seconds and 63 seconds, which is not significantly long. The only notable characteristic is that the data transfer size is very large compared to normal traffic.

Table 6.1 Background and attack traffic analysis for two data exfiltration events

Timestamp	Duration (seconds)	Data Exfiltration size (MB)	Background traffic rate (flow/s)	Attack traffic rate (flow/s)
4/6, 10:36	125	103	41.1	1.0
4/7, 7:00	63	596	79.2	1.0

Similarly, Table 6.2 provides scanning and background network traffic statistics related to two port-scan attacks, and the first one is obvious and second one is subtle. The “obvious” port scanning on April 6 has a very significant attack traffic rate, which is 270.18 flow/s. Comparing to background traffic rate, more than half of the traffic was associated to the port scan event. Moreover, the duration is also very long (100 minutes). All of these factors make this port scanning relatively easily to be identified.

In contrast, the subtle only lasted 5 minutes and attack traffic rate was 0.60 flow/s. As a result, filtering out the background traffic to detect port scanning in such cases is rather difficult and needs more effort.

Table 6.2 Background and attack traffic analysis for two port scanning intrusions

Timestamp	Duration (minutes)	Number of Scanned hosts	Background traffic rate (flow/s)	Attack traffic rate (flow/s)
4/6, 11:10	100	8	447.14	270.18
4/7, 7:00	5	6	20.75	0.60

Background and attack traffic analysis is not meaningful for denial-of-service attacks, however. In typical denial-of-service attacks, the attack traffic rate is much higher than the normal background traffic rate, making these attacks easily detectable from network traffic.

6.2 Metrics Evaluation

In the process of evaluating intrusion detection systems, true positives, true negatives, false positives, and false negatives are the most fundamental metrics. Because the ground truth lists all intrusions, we need to count the true positives that we can identify from the overview. For the Duration-Payload overview, we focus primarily on denial-of-service attacks and data exfiltration. Even though server redirection can be identified in this case, there is only one server redirection attack. For the Host-MaxConn overview, the main goal is to identify port-scan attacks.

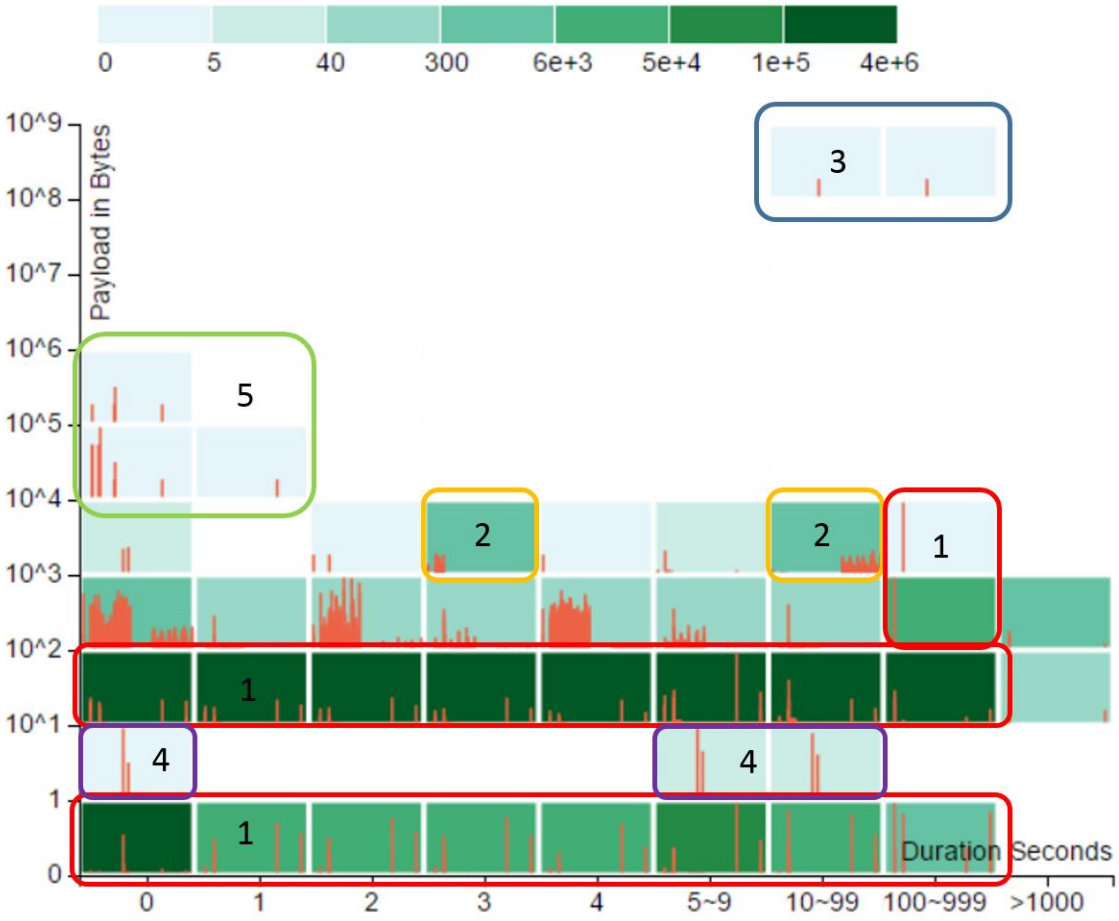

Figure 6.1 Duration-Payload Overview with multiple suspicious zones

In the Figure 6.1, we have highlighted the suspicious zones for different types of intrusions, and our primary goal is to identify denial of service attack, server redirection, and data exfiltration, although port scanning are also identified as we discussed in the previous section. The suspicious zones with different colors and number labels are discussed here:

1. Red Zone. The red zone identifies denial-of-service attacks. In the Duration-payload overview, quite a few cell graphs have peaks in similar positions, marked by the red rectangle. As discussed in the previous chapter, there are four significant peaks; the first two are indeed denial-of-service attacks, and the last two are port-scan attacks (these cannot be counted as false negatives because they are intrusions). So for the category of denial of service, the two attacks are both successfully identified, and there is no false negative or false positive.

2. Yellow Zone. The yellow zone identifies server redirections. Two cell graphs are selected in this case because they both have significant but inconsistent traffic records during the 2 weeks. Our investigation (discussed in Chapter 5) showed that the left one is server redirection, but the right one is not. Further analysis indicated reconfiguration on some servers, resulting in the traffic pattern shift in the right one, but it is not a server redirection attack. For the category of server redirection, the server-redirection attack is successfully identified, and there is one false positive.

3. Blue Zone. The blue zone identifies data exfiltration. The Duration-Payload overview reveals two cell graphs that contain NetFlow records potentially related to data exfiltration: They both have abnormally large payloads and are of relatively long

duration. Our study showed that these two records are between one internal server and one external FTP server, and ground truth confirms that these two events are data exfiltration. For data exfiltration, all the related attacks have been identified, and there are no false positives or false negatives.

4 and 5. Purple and Green Zones. These two zones are not specific for any type of attack, but they contain some isolated peaks. The two peaks in the purple zones are actually port-scan attacks, as seen in Figure 6.2, where a wide range of destination ports of internal servers are scanned, from 80 to over 55,000. Because we do not evaluate port-scan attacks in the Duration-Payload overview, we do not count these isolated peaks either as true positives or as false positives.

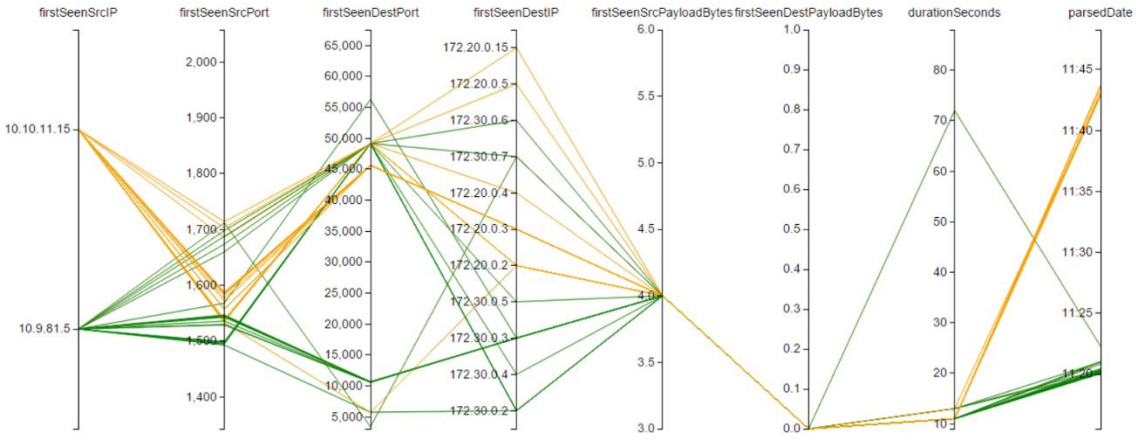

Figure 6.2 NetFlow records from purple zone

From an analysis of the different zones in Figure 3.1, we get the four metrics for the overview, as shown in Table 3.3, where TN is true negative, FP is false positive, FN is false negative, and TP is true positive.

Table 6.3 Standard metrics for Duration-Payload Overview evaluation

		Predicted Label	
		Normal	Intrusion
Actual Class	Normal	21(TN)	2(FP)
	Intrusion	0(FN)	12(TP)

From Table 6.3, we can see that there is even no false negatives, indicating all the intrusions (denial of service, server redirection and data exfiltration) are identified successfully. Next, we can calculate detection rate and false positive rate based on these metrics.

Detection rate (DR) is defined as the ratio between the amount of correctly identified attacks and the total amount of attacks, that is:

DR = TP/ (FN + TP) = 12/(0+12) = 1.0

False positive rate (FP) is defined as the ratio between the amount of normal NetFlow that are mistakenly identified as intrusions and the total amount of normal NetFlow, that is:

FP = FP/(TN + FP) = 2/(21+2) = 0.087

From the results of detection rate and false positive rate, we can see that all the denial of service attacks and data exfiltration were successfully identified and the false positive rate is only 0.087. Remember the null hypothesis for DoS attacks, FP <= 0.10 for detecting denial of service attacks (significance level 0.05). Thus here we accept the null

hypothesizes for denial of service attacks and data exfiltration. In other words, based on evaluation of VAST data set, Duration-Payload can effectively identify DoS attacks and data exfiltration.

On the other hand, for the Host-MaxConn Overview, we effectively identified all 9 external port-scan attackers, including 7 obvious attackers and 2 subtle attackers during the two weeks. There was only one external host, which was labeled as attacker incorrectly. Moreover, there are quite a few external hosts with significant inbound traffic, and they are possibly normal external servers. The data is summarized in Table 6.4.

Table 6.4 Standard metrics for Host-MaxConn Overview evaluation

		Predicted Label	
		Normal	Intrusion
Actual Class	Normal	20(TN)	1(FP)
	Intrusion	0(FN)	9(TP)

Similarly, in this case, detection rate is 1.0, and false positive rate is 0.048. Remember the null hypothesis for port scan attacks, FP <= 0.10 for detecting port scan attacks (significance level 0.05). Therefore, we accept the null hypothesizes for port scan attacks. In other words, based on evaluation of VAST data set, Host-MaxConn is capable of classifying port scan attacks.

6.3 Chapter Summary

In this chapter, we evaluate the two overviews systematically. A background and attack traffic analysis reveals that subtle attacks are indeed difficult to detect from massive network traffic. We investigate the two overviews, calculating detection rate, false-positive rate, and specificity. Detection rate and specificity can reach over 0.9, and the false-positive rate is below 0.1. Therefore, detection of the four types of attacks (denial of service, server redirection, data exfiltration, and port scan) in VAST data sets is successful.

CHAPTER 7. CONCLUSIONS AND FUTURE WORK

7.1 Conclusions

In the previous chapters, we briefly stated reasons to apply visual analytics to network security and intrusion detection. Related literature was presented and discussed, indicating that many visual analytics systems do not provide an appropriate overview to detect subtle intrusions effectively. Research methodology was presented in Chapter 3, including research framework, system components, primary data sources, and system evaluation criteria. In Chapter 4, the VA system was described in detail, including two overviews (Duration-Payload and Host-MaxConn) and several zoomed-in visualizations. The functions of these approaches were discussed by presenting how to use them to highlight various characteristics of network intrusions. Therefore, our approach is a characteristics-based VA approach. In Chapter 5, three primary attacks types (denial of service, data exfiltration, and port scan) were discussed, showing how to use the VA system to identify these attacks in the VAST data sets. Interestingly, server redirection can also be identified from Duration-Payload overview. Finally, we evaluated the VA approach systematically with the data set's ground truth, showing that it is capable of identifying intrusion events with very few false positives.

Based on the VA system and evaluation results in the previous chapter, we can draw the following conclusions:

(1) The Duration-Payload overview is capable of detecting denial of service and data exfiltration effectively. For the VAST data sets, false positive rate is only 0.087, which is significantly less than our pre-defined criteria 0.10 with significant level 0.05.

(2) The Host-MaxConn overview is capable of detecting port scanning effectively. For the VAST data sets, false positive rate is only 0.048, which is significantly less than our pre-defined criteria 0.10 with significant level 0.05.

(3) Duration-Payload overview can be used to detect server redirection based on the patterns of network traffic.

(4) Zoomed-in visualizations can facilitate analyst investigations of security events and provide situational awareness.

(5) Characteristic-based visual analytics approaches have been proven to be effective and practical in detecting subtle attacks from network traffic data. The approaches also provide some degree of scalability considering the size of Big Marketing network.

7.2 Future Work

The visual analytics approach developed in this project has been proven capable of identifying both obvious and subtle intrusions from massive network traffic. Future work will focus on the following:

(1) Evaluate the system with another data set and optimize the system accordingly. Because we evaluated the VA system only with the VAST data set, it would be better to test and evaluate the system with other data sets.

(2) Incorporate a more highly automated process into the VA system so that identifying suspicious network traffic can be done more intelligently.

(3) Extend the system to detect other network intrusions, such as botnets, malware infection and other application-layer attacks. Currently, the VA system focuses only on three common types of network intrusion.

(4) Test the VA system in a real-world network to see how it behaves in a practical scenario.

LIST OF REFERENCES

LIST OF REFERENCES

Abdullah, K., Lee, C. P., Conti, G., & Copeland, J. A. (2005).Visualizing network data for intrusion detection. In *Information Assurance Workshop, 2005.IAW'05.Proceedings from the Sixth Annual IEEE SMC* (pp. 100–108). IEEE.

Ardito, D., Byreddy, A., Orchier, J., Salvaterra, L., & Soriano, R. (2000). *Computer network security management system*. Google Patents.

Ball, R., Fink, G. A., & North, C. (2004). Home-centric visualization of network traffic for security administration. In *Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security* (pp. 55–64). ACM.

Bhuyan, M. H., Bhattacharyya, D. K., & Kalita, J. K. (2011). Surveying port scans and their detection methodologies. *The Computer Journal*, bxr035.

Camacho, J., Macia-Fernandez, G., Diaz-Verdejo, J., & Garcia-Teodoro, P. (2014). Tackling the Big Data 4 vs for anomaly detection. In *Computer Communications Workshops (INFOCOM WKSHPS), 2014 IEEE Conference on* (pp. 500–505). IEEE.

Card, S. K., Mackinlay, J. D., & Shneiderman, B. (1999).*Readings in information visualization: Using vision to think*. Morgan Kaufmann.

Conti, G., & Abdullah, K. (2004). Passive visual fingerprinting of network attack tools. In *Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security* (pp. 45–54). ACM.

Cook, K., Grinstein, G., Whiting, M., Cooper, M., Havig, P., Liggett, K., … Paul, C. L. (2012). VAST Challenge 2012: Visual analytics for big data. In *Visual Analytics Science and Technology (VAST), 2012 IEEE Conference on* (pp. 251–255). IEEE.

Debar, H., & Wespi, A. (2001). Aggregation and correlation of intrusion-detection alerts. In *Recent Advances in Intrusion Detection* (pp. 85–103). Springer.

Fink, G. A., Muessig, P., & North, C. (2005). Visual correlation of host processes and network traffic. In *Visualization for Computer Security, 2005. (VizSEC 05). IEEE Workshop on* (pp. 11–19). IEEE.

Fischer, F., & Keim, D. (2013).Vacs: Visual analytics suite for cyber security-visual exploration of cyber security datasets.

Fischer, F., Mansmann, F., Keim, D. A., Pietzko, S., & Waldvogel, M. (2008). *Large-scale network monitoring for visual analysis of attacks*. Springer.

Foresti, S., Agutter, J., Livnat, Y., Moon, S., & Erbacher, R. (2006). Visual correlation of network alerts. *Computer Graphics and Applications, IEEE*, *26*(2), 48–59.

Gates, C. (2006). Co-ordinated port scans: a model, a detector and an evaluation methodology.

Giani, A., Berk, V. H., & Cybenko, G. V. (2006). Data exfiltration and covert channels. In *Defense and Security Symposium* (pp. 620103–620103–11). International Society for Optics and Photonics.

Goodall, J. R. (2008). Introduction to visualization for computer security. In *VizSEC 2007* (pp. 1–17). Springer.

Goodall, J. R., Lutters, W. G., Rheingans, P., & Komlodi, A. (2005). Preserving the big picture: Visual network traffic analysis with TNV. In *Visualization for Computer Security, 2005.(VizSEC 05). IEEE Workshop on* (pp. 47–54). IEEE.

Goodall, J. R., Lutters, W. G., Rheingans, P., & Komlodi, A. (2006). Focusing on context in network traffic analysis. *Computer Graphics and Applications, IEEE*, *26*(2), 72–80.

Goodall, J. R., & Sowul, M. (2009). VIAssist: Visual analytics for cyber defense. In *Technologies for Homeland Security, 2009.HST'09. IEEE Conference on* (pp. 143–150). IEEE.

Itoh, T., Takakura, H., Sawada, A., & Koyamada, K. (2006).Hierarchical visualization of network intrusion detection data.*Computer Graphics and Applications, IEEE*, *26*(2), 40–47.

Jiawan, Z., Liang, L., Liangfu, L., & Ning, Z. (2008). A novel visualization approach for efficient network scans detection. In *Security Technology, 2008. SECTECH'08. International Conference on* (pp. 23–26). IEEE.

Jung, J., Paxson, V., Berger, A. W., & Balakrishnan, H. (2004). Fast portscan detection using sequential hypothesis testing. In *Security and Privacy, 2004. Proceedings. 2004 IEEE Symposium on* (pp. 211–225). IEEE.

Keim, D. A. (2002).Information visualization and visual data mining. *IEEE Transactions on Visualization and Computer Graphics*, *8*(1), 1–8. doi:10.1109/2945.981847

Keim, D. A., Mansmann, F., Schneidewind, J., & Schreck, T. (2006).Monitoring network traffic with radial traffic analyzer. In *Visual Analytics Science And Technology, 2006 IEEE Symposium On* (pp. 123–128). IEEE.

Keim, D., Andrienko, G., Fekete, J.-D., Görg, C., Kohlhammer, J., & Melançon, G. (2008). *Visual analytics: Definition, process, and challenges*. Springer.

Kim, J., & Lee, J.-H. (2008). A slow port scan attack detection mechanism based on fuzzy logic and a stepwise policy.

Koike, H., Ohno, K., & Koizumi, K. (2005). Visualizing cyber attacks using IP matrix. In *Visualization for Computer Security, 2005. (VizSEC 05). IEEE Workshop on* (pp. 91–98). IEEE.

Lakkaraju, K., Yurcik, W., & Lee, A. J. (2004). NVisionIP: Netflow visualizations of system state for security situational awareness. In *Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security* (pp. 65–72). ACM.

Liao, H.-J., Richard Lin, C.-H., Lin, Y.-C., & Tung, K.-Y. (2013). Intrusion detection system: A comprehensive review. *Journal of Network and Computer Applications*, *36*(1), 16–24.

Livnat, Y., Agutter, J., Moon, S., Erbacher, R. F., & Foresti, S. (2005). A visualization paradigm for network intrusion detection. In *Information Assurance Workshop, 2005.IAW'05.Proceedings from the Sixth Annual IEEE SMC* (pp. 92–99). IEEE.

Marty, R. (2009). *Applied security visualization*. Addison-Wesley.

Nyarko, K., Capers, T., Scott, C., & Ladeji-Osias, K. (2002). Network intrusion visualization with NIVA, an intrusion detection visual analyzer with haptic integration. In *Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2002. HAPTICS 2002. Proceedings. 10th Symposium on* (pp. 277–284). IEEE.

Onut, I.-V., & Ghorbani, A. A. (2007). SVision: A novel visual network-anomaly identification technique. *Computers & Security*, *26*(3), 201–212.

Paxson, V. (1999).Bro: a system for detecting network intruders in real-time. *Computer Networks*, *31*(23), 2435–2463.

Phan, D., Gerth, J., Lee, M., Paepcke, A., & Winograd, T. (2008). Visual analysis of network flow data with timelines and event plots. In *VizSEC 2007* (pp. 85–99). Springer.

Shiravi, H., Shiravi, A., & Ghorbani, A. A. (2012). A survey of visualization systems for network security. *Visualization and Computer Graphics, IEEE Transactions on*, *18*(8), 1313–1329.

Staniford, S., Hoagland, J. A., & McAlerney, J. M. (2002).Practical automated detection of stealthy portscans. *Journal of Computer Security*, *10*(1), 105–136.

Taylor, T., Brooks, S., & McHugh, J. (2008). NetBytes viewer: an entity-based NetFlow visualization utility for identifying intrusive behavior. In *VizSEC 2007* (pp. 101–114). Springer.

VAST. (2013). Challenge 2013: Mini-Challenge 3. Retrieved Dec 5, 2014, from http://vacommunity.org/VAST+Challenge+2013%3A+Mini-Challenge+3.

Vivo, M. de, Carrasco, E., Isern, G., & Vivo, G. O. de. (1999). A review of port scanning techniques. *SIGCOMM Comput. Commun. Rev.*, *29*(2), 41–48.

Yen, T.-F., Oprea, A., Onarlioglu, K., Leetham, T., Robertson, W., Juels, A., & Kirda, E. (2013). Beehive: large-scale log analysis for detecting suspicious activity in enterprise networks. In *Proceedings of the 29th Annual Computer Security Applications Conference* (pp. 199–208). ACM.

Yin, X., Yurcik, W., Treaster, M., Li, Y., & Lakkaraju, K. (2004). VisFlowConnect: netflow visualizations of link relationships for security situational awareness. In *Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security* (pp. 26–34). ACM.

Zadeh, L. A. (1994). Fuzzy logic, neural networks, and soft computing. *Communications of the ACM*, *37*(3), 77–84.

Zhao, Y., Liang, X., Fan, X., Wang, Y., Yang, M., & Zhou, F. (2014). MVSec: multi-perspective and deductive visual analytics on heterogeneous network security data. *Journal of Visualization*, *17*(3), 181–196.