ADVANCED VISUALIZATIONS FOR NETWORK SECURITY

A Dissertation
Presented to
The Academic Faculty

By

Troy Nunnally

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy
in
Electrical and Computer Engineering

School of Electrical and Computer Engineering
Georgia Institute of Technology
December 2014

ADVANCED VISUALIZATIONS FOR NETWORK SECURITY

Approved by:

Dr. Raheem Beyah, Committee Chair *Associate Professor, School of ECE* *Georgia Institute of Technology*	Dr. Yusun Chang *Asst. Professor, School of ECE* *Georgia Institute of Technology*
Dr. John Copeland *Professor, School of ECE* *Georgia Institute of Technology*	Dr. John Stasko *Professor, College of Computing* *Georgia Institute of Technology*
Dr. Henry Owen *Professor, School of ECE* *Georgia Institute of Technology*	

Date Approved: July 16, 2014

To my family, friends, and in loving memory of Maurice Nunnally

ACKNOWLEDGMENTS

First and foremost, I would like to dedicate this dissertation to my Lord and Savior, for without Him my work would not have been made possible. As a testament of my sincere gratitude, for all that You have done for me, I commit and dedicate this work to You (Proverbs16:3).

I would like to also express my deepest appreciation to my advisor, Dr. Raheem A. Beyah, and Dr. John A. Copeland. Both have been tremendous mentors and supporters of my research. It has been with their constant encouragement, priceless advice and patience that has helped propel my research and allowed me to grow within my role as a researcher. I am proud to be a member of the Communications Assurance and Performance (CAP) Group and Communications Systems Center (CSC) in the School of Electrical and Computer Engineering (ECE) at the Georgia Institute of Technology.

A truly thankful acknowledgement goes to my mentor, Dr. A. Selcuk Uluagac. Without his strong guidance and persistent help, this dissertation would not have been possible. Dr. Selcuk's useful comments, great mentorship, meticulous remarks, and enthusiastic engagement have been instrumental in the learning process and the successful completion of my thesis.

Also, I would like to express my cordial thanks to Jill Auerbach and Julie Riding for the opportunity to be a mentor in the Opportunity Research Scholar (ORS) program for the School of ECE; Dr. Kulsoom Abdullah and Penyen Chi for lending an extra hand of support; and the participants in my survey and user testing, who have willingly shared their precious time during the process.

Furthermore, I would like to extend my gratitude to my committee members, professor Dr. Henry Owen and Dr. Yusun Chang for generously serving as my committee members and giving their time. I also want to thank them for their brilliant comments and suggestions.

A special thanks goes to my family. Words cannot express how grateful I am to my mother Sheila Thomas, father Julius Thomas, brother Travis Nunnally, sisters Mia Nunnally and Farrah Brown, and other close family members (e.g., cousins, grandmother, aunts, and uncles) for all of the sacrifices that they have made on my behalf. Their prayer for me has sustained me, and their unconditional love, unwavering support, and encouragement has strengthened me throughout my Ph.D. Additionally, I would also like to thank all of my friends who supported me in writing and encouraged me to strive towards my goal.

Lastly, I would like to convey a sincere appreciation to my fiancé and love of my life, Tanysha Monique Stevenson, who spent sleepless nights with me writing this dissertation and who always provided me with support throughout my Ph.D. It has truly been with her love, kindness, and unwavering support during the past two years that has helped me to finalize this thesis.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

Increasingly, cyber security plays a critical role in the welfare of computer networks for every organization. Computer networks must be secure to ensure healthy operations and protection of valuable electronic assets (e.g., credit card numbers, account names, and passwords). However, to achieve this goal, the data within these computer networks must be monitored across multiple sources such as vulnerability scanners, Intrusion Detection Systems (IDSs), firewalls, and host systems. Often times, monitoring volumes of data across multiple sources can potentially be overwhelming. As a result, vital data is at a greater risk of being overlooked and the time span for analyzing it could be too lengthy.

One way to address this issue is to employ network security visualization techniques to evaluate security risks and identify malicious activity to help mitigate compromised nodes on a network. These visualization techniques convert textual network activity into meaningful two-dimensional (2D) or three-dimensional (3D) visual representations, which allow a user to explore and understand large amounts of information. If a visualization technique is well-designed, a user could quickly gain new insights and make more informed decisions about network datasets. Building upon this idea, the purpose of this thesis is to introduce a visualization framework to help reduce task-completion time, enhance situational awareness, decrease user error, and increase the learnability of complex visualizations for network security applications. From the developed framework, three techniques are suggested as contributions using visualization and interaction: (1) a *Stereoscopic visualization technique* aims to increase user awareness of vulnerabilities and malicious attacks, (2) a *recommender system* aims to ensure efficient navigation in complex 3D environments, and (3) an interaction system aims to assist in usability of visualization environments using *Natural User Interfaces (NUIs)*. To investigate the aforementioned techniques, the following tools were created: 3D Stereoscopic Vulnerability Assessment Tool (3DSVAT) [9], Parallel 3D Coordinate Visualization (P3D) [6], *NAVSEC* recommender system [10], and

Interaction System for Network Security (InterSec) [11].

CHAPTER 1

INTRODUCTION

Network administrators are often given tasks to evaluate security risks and malicious activity between local and remote hosts on a computer network. Most malicious activity is monitored via various network systems (e.g., data-sources) such as vulnerability scanners, firewalls, and Intrusion Detection Systems (IDSs). These systems produce datasets that are often represented in the form of textual logs and network traffic packet capture (pcap) files. However, as data volume continues to grow, textual representations of raw output from multiple data-sources may potentially become too overwhelming to efficiently evaluate malicious security risks in a timely manner. Historically, to overcome this challenge, researchers have investigated converting textual representations of network datasets into two-dimensional (2D) visual representations to enhance data analysis with visual aids for further investigation of larger datasets [12]. 2D visualizations produce representations on the x and y axes, which are used to identify, detect, and analyze malicious information. Yet as networks become more complex, depicting considerable amounts of information within 2D visualizations can be perceived as cluttered, limited, or misleading [13].

One way to address this design issue is to explore methods for expanding visualization techniques by incorporating the z-direction. The addition of the z-direction provides depth for three-dimensional (3D) visualizations that allow more information to be visualized and result in clearer representations. As a result, 3D visualizations are intuitively perceived to possess less clutter than its 2D and textual counterparts [13, 14, 15]. Furthermore, clearer 3D visualizations assist network security administrators in identifying a substantial amount of malicious information and gaining a more accurate global view of the data's structure. The work presented in this thesis extends the current research in 3D visualizations by producing a framework to leverage advances in 3D visualization techniques and stereoscopic (i.e., use of 3D glasses) 3D technologies. From this framework, various 3D visualization tools and techniques are developed to evaluate the effectiveness of 3D visualizations.

Although 3D visualizations could potentially be useful in network security applications, the advantages of 3D visualizations alone may not maximize user efficiency and reduce user error to identify and resolve critical network attacks. Network administrators must be acclimated with interacting in a network's 3D environment. Most users are familiar with basic interactions for 2D environments on traditional computer screens, keyboards, and mice. However, the third dimension adds its own complications and complexities in which a network administrator may not be aware of how to navigate within a 3D space to review and analyze vital network information. Thus, tools are needed to guide and recommend intuitive interactions for seamless navigation in 3D environment for quicker task completion and identification of malicious attacks. This thesis expounds upon a prototype system, the *NAVSEC* Recommender System [10], that can be implemented to assist network administrators in detection and analysis of network activity.

Additionally, 2D, 3D and stereoscopic 3D visualization techniques for network security applications are often employed on traditional desktop, mouse, and keyboard setup of WIMP (Windows, Icons, Menus, and a Pointer) interfaces [5]. These WIMP interfaces use an indirect pointing device (e.g., a mouse) where a user positions and tracks a digital cursor to target an object that represents network attributes (i.e., a node on a network). The benefit of WIMP interfaces is that they provide simple, easy-to-learn, and easy-to-use ”point-and-click” interactions [16]. Additionally, these interfaces are well-supported by current developers for network security applications. However, a single mouse cursor provides a maximum of two spatial degrees of freedom (e.g., cursor movement along the x and y axes). As a result, tasks that require manipulating more than two degrees of freedom (e.g., 3D interaction) must be broken up into multiple user actions. Furthermore, with a single cursor, users must make long traversals between spatially distant elements within an interface. The limited amount of degrees of freedom and long traversal cause difficulty scaling mouse interactions for more complex applications [17]. Often, WIMP interfaces require a user to perform many interactions while the user is navigating through a vast visualization environment. As a result, the time span to prioritize and analyze critical data becomes a lengthy process and vital data is at a greater risk of being overlooked. To alleviate these problems, researchers are investigating mouseless technologies (e.g., touch-enabled phones/monitors and Microsoft Kinect) that allow for more than two degrees of freedom [18].

New mouseless technologies are referred to as Natural User Interfaces (NUIs) [19]. NUIs utilize interactions that are ”natural” to the user and provide several key advantages from traditional mouse and keyboard input. One advantage is NUIs, such as multi-touch interfaces, provide interactions that can use up to all 10 fingers on both hands and provide up to 20 degrees of freedom. This allows users to perform more interactions and allow interactions to be performed in parallel. Research has shown that multi-touch interaction is about twice as fast as mouse interaction for tasks such as selecting objects that may represent nodes on a network [20]. In this thesis, the benefits of NUIs are investigated in complex 3D visual environments for network security applications. In an effort to leverage the advantages of NUI and stereoscopic 3D visualization, a Framework for Rendering Enhanced 3D Stereoscopic (FRE3DS) visualizations for network security is also designed and implemented. By employing this framework, three techniques are investigated: *stereoscopic visualizations technique* to increase user awareness of vulnerabilities and malicious attacks, *recommender systems* to ensure efficient navigation in complex 3D environments, and a *Natural User Interface (NUI)* system to assist in usability of visualization environments.

1.1 Research Objectives and Solutions

The objective of this thesis is to develop efficient interaction and visualization techniques for a network administrator to (a) enhance situational awareness using stereoscopic 3D technologies (b) reduce task completion time using recommendation systems (c) decrease user error by increasing the visualization space and (d) increase learnability of complex visualizations for network security applications by incorporating NUIs. In an effort to achieve these goals, the FRE3DS framework is developed to address three important aspects of usability in network security: visualization, user navigation, and user interaction. Each aspect is investigated using four prototypes designed from the FRE3DS framework:

1. 3D Stereoscopic Vulnerability Assessment Tool (3DSVAT) ensures increased user awareness of vulnerabilities and malicious attacks [9].

2. Parallel 3D Coordinate Visualization (P3D) prevents visualization attacks, specifically IP confusion and windshield wiper attacks [6].

3. *NAVSEC* Recommender System promotes rapid learning of complex tasks [10].

4. InterSec : Interaction System for Network Security assists in natural interaction of visualization environments [11].

In comparison to traditional 3D visualization interactions, these prototypes use state-of-the-art 3D, multi-touch, and motion sensing input devices (e.g., Microsoft Kinect and 3M multi-touch display) for enhanced interaction usability and quicker response times for potential users to navigate complex visualization environments within network security. More details for each specific prototype and the FRE3DS framework are provided in the following subsections.

1.1.1 Framework for Rendering Enhanced 3D Stereoscopic (FRE3DS)

As discussed later, FRE3DS framework was developed for producing rapid customized 3D visualizations for network administrators to easily and quickly develop various visualizations and efficiently investigate data. As previously mentioned, the four prototypes were developed to prevent occlusion attacks, increase situational awareness, reduce interactions using gesture sets, and reduce response times by using a recommender system.

1.1.2 3D Stereoscopic Vulnerability Assessment Tool (3DSVAT)

Stereoscopic 3D uses headgear or glasses to enhance the perception of depth for administrators to quickly detect vulnerable nodes on a network in a 3D space. Utilizing the FRE3DS framework, a 3D Stereoscopic Vulnerability Assessment Tool (3DSVAT) was developed to investigate the usage of both monocular cues such as perspective, size, and occlusion and binocular cues such as binocular disparity to enhance situational awareness. Through the introduction of 3DSVAT, users could potentially identify data more quickly and to accurately display complex information.

1.1.3 Parallel 3D Coordinate Visualization (P3D)

P3D assists in detection and increased awareness of distributed coordinate attacks. Moreover, by adding an extra dimension for a parallel 3D coordinate visualization, P3D can prevent information overload and certain types of occlusion-based attacks. Occlusion-based attacks occur when an attacker injects spoofed packets in a network so that legitimate data is partially or completely not rendered to a display. Using the enhanced perception of depth in a stereoscopic 3D visualizations, P3D includes a stereoscopic awareness region, which helps identify network scans of interest without further filtering techniques and reduces potential data loss.

1.1.4 *NAVSEC* Recommender System

3D visualization tools often require advanced knowledge in networking, visualization, and information security to operate, navigate, and successfully examine malicious attacks. Novice users, deficient in the required advanced knowledge, may find navigation within these visualization tools difficult. Thus, a visualization module was developed within the FRE3DS framework called NAVSEC. NAVSEC is a recommender system prototype for navigating in 3D network security visualization applications. NAVSEC recommends visualizations and interactions to novice users. Given visualization interaction input from a novice user and expert communities, NAVSEC can be used to reduce confusion for a novice user while navigating in a 3D visualization. NAVSEC illustrates with a use-case from an emulated stealthy scanning attack disguised as a file transfer with multiple concurrent connections. Through the use of NAVSEC, the use-case demonstrates a novice user's visualization converges towards a visualization used to identify or detect a suspected attack by an expert user. As a result, NAVSEC can successfully guide the novice user in differentiating between complex network attacks and benign legitimate traffic with step-by-step created visualizations and suggested user interactions.

1.1.5 InterSec : Interaction System for Network Security

A visualization module was developed, from the FRE3DS framework, called *InterSec*. InterSec is an *interaction system* prototype that interacts with 3D network security visualization tools. InterSec helps network administrators utilize gesture sets to coordinate multiple inputs across multiple interaction technologies. Using InterSec, a gesture set was adopted from GestureWorks [8], which combines multiple interactions into a single interaction to further reduce response times for accomplishing a task such as finding a set of scanned ports for a node. Through the implementation of the gesture set, users can combine the use of multiple network tools to evaluate network data more efficiently than its traditional WIMP counterparts. The FRE3DS framwork applies this gesture set to NUI interfaces such as multi-touch and hand gestures using 3M multi-touch monitor and Microsoft Kinect (Kinect for short), respectively. As mentioned earlier, multi-touch systems supports all ten fingers as input, providing many more input degrees of freedom than mouse inputs. As a result, network security users possess more interaction options to potentially manipulate data more quickly. InterSec takes advantage of this increased set of interactions to intuitively represent a series of smaller interactions or a commonly-used network security task (e.g., filter a packet capture) to reduce interaction time and quickly identify malicious attacks on the network.

1.2 Thesis Outline

The remainder of this thesis is organized as follows: Chapter 2 discusses origin and history of network security visualization and interaction tools and techniques. Chapter 3 presents details of the FRE3DS fremework, its general principals and design. Chapter 4 discusses the 3D Stereoscopic Vulnerability Assessment Tool (3DSVAT). Chapter 5 expounds upon Parallel 3D Coordinate Visualization (P3D) systems. Chapter 6 provides insight about the recommender system of NAVSEC. Next, Chapter 7 discusses the Interaction System for Network Security (InterSec). Lastly, Chapter 8 concludes the thesis and presents guidance for future work in the field.

CHAPTER 2
ORIGIN OF WORK

The background and historical concepts of network security and visualization techniques are further explored in this chapter. A substantial review of previous research and literature on network security interactions with 2D, 3D and stereoscopic visualization techniques are provided in great detail. Specifically, the first section presents a history of visualization. Next, the following sections include an overview of 2D, 3D, and stereoscopic 3D visualization within the field of network security. Lastly, a literature review of interaction techniques is discussed (e.g., recommender systems and multi-touch interaction) and their relationship to network security.

2.1 History of Information Visualization for Network Security

Information Visualization (InfoVis) is the study of visual representations and interaction techniques from abstract data. The first occurrence of InfoVis started in 1786 when William Playfair, a Scottish engineer, portrayed the line graph and bar chart of economic data [21]. Since the beginning of InfoVis, InfoVis has evolved significantly with the advancement of computer graphics to incorporate more complex 2D/3D techniques (3D scatterplot, 2D/3D parallel coordinates, 2D link graphs). These techniques take advantage of a user's massive visual bandwidth and their ability to process large amounts of visual data. Subsequently, these advantages allow users to explore, analyze, and understand useful trends and patterns within more complex and abstract data.

Since InfoVis techniques (e.g., scatter plots, histograms, and line charts) are fundamentally used in data analysis, the application of information visualization techniques is widely used in most fields of study that contain applied research and problem solving. Therefore, InfoVis is considered a critical component for many fields such as scientific research, business methods, data mining, financial data analysis, education, market studies, genetics, and drug discovery. At its core, InfoVis systems consist of two main components: *visual representation* and *user interaction*. The visual representation component maps textual data to graphical content that is rendered on a display. The user interaction component involves a user's manipulation of a system through a series of interactions as a user explores and analyzes datasets. However, depending on the characteristics and attributes of the dataset and the needs of the user, visualization type and interaction techniques may change as well. Therefore, various areas of research has created subsets of InfoVis to solve specific problems as it relates to a particular area of research. In this thesis, a specialized subset of Infovis will be investigated which is Visualization for Network Security (VizSec), with an emphasis on 3D visualizations.

VizSec, as the name suggests, is the study of visualization techniques for network security. VizSec has rapidly matured over the past several years. VizSec was first used in the visual representation of IDS logs [22]. However, advances in computational power has allowed the spread of many 2D/3D visualization techniques and tools to be used across many network security application (e.g., situational awareness, malware detection, SCADA system security, intrusion detection, stealthy port scanning). Primarily, VizSec explores malicious activities on a network using visual representation. Yet, historically, in a variety of fields such as statistics, pattern recognition, machine learning, and data mining, there are other traditional techniques to automatically detect, monitor and analyze network attacks. However, as network security attacks continue to become more complex and new algorithms are developed to prevent automatic detection by traditional methods, VizSec is needed as an aid to further identify patterns and anomalies that may go undetected.

2.2 2D Visualization Techniques

As discussed in Chapter 1, 2D visualizations produce representations on the horizontal (x) and vertical (y) axes to identify, detect, and analyze malicious information. 2D visualizations are used to visualize network scans, analyze attack patterns and graph routing behavior [23, 24] from data sources such as server logs, packet capture traces, NetFlow data, IDS logs [25, 15], firewall logs [26], and BGP traces. Although the type of visualization greatly depends on the data source, understanding the benefits and drawbacks of 2D visualization techniques may help researchers develop better user-centered approaches. Furthermore, although the majority of the research presented in this thesis focuses on 3D visualizations, comprehending the advantages and limitations of 2D visualizations techniques are important in developing 3D visualizations techniques. Therefore, a non-exhaustive list of commonly-used 2D visualization techniques are described.

2.2.1 Glyph-based Visualization

Glyph-based visualization perceptually links a marker (*glyph*) to an important characteristic in the information, thereby facilitating the rapid transfer of information to the user. A glyph uses an arrow that may signify the number of hosts on a network or a square, which may represent average packet size. The glyphs are commonly used to make decisions about the current or past snapshots of a network. Some glyph-based techniques map data parameters from a system's logs [27] or associated system statistics (e.g., system load, number of users, and consumed disk space). An early glyph-based visualization is Visual Information Security Utility for Administration Live (VISUAL) [1]. VISUAL is a network security visualization tool that allows users to view communication patterns between an internal network and an external host. In addition, the tool assists users in detecting abnormal traffic such as port scans or DoS attacks. As illustrated in Figure 1, VISUAL shows a representation of each internal host as a small square within a larger grid. The larger grid depicts a set of home hosts to a yellow square, which represents an external host. VISUAL provides insight for networks with up to 2,500 home hosts and 10,000 external hosts.

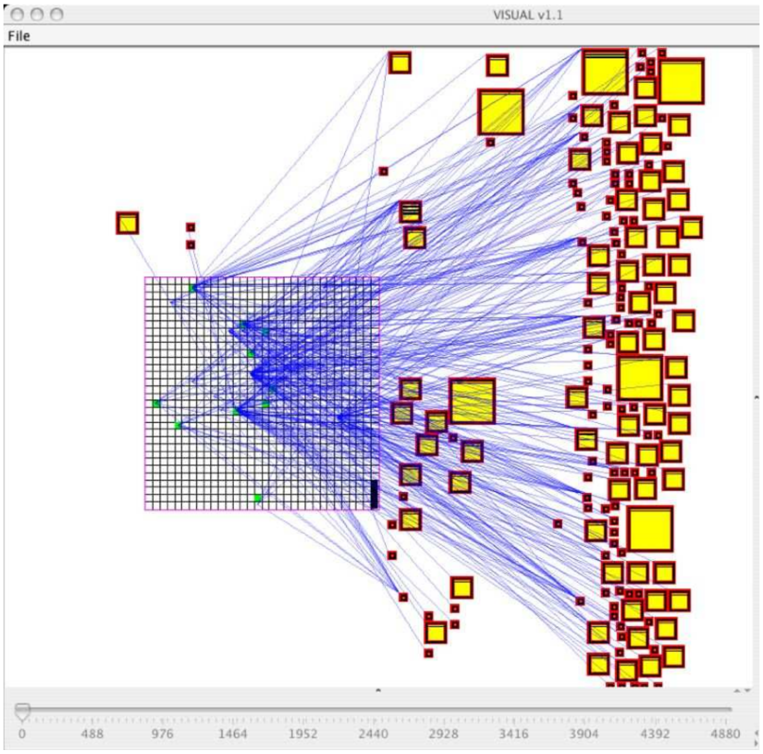


Figure 1: VISUAL [1] displaying 80 hours of network data on a network of 1,020 hosts.

In Figure 1, VISUAL displays 80 hours of network data on a network of 1,020 hosts. Although glyph-based systems are useful in portraying a large number of attributes for a node, displaying a number of attributes for multiple nodes becomes challenging, especially in a 2D visualization. Thus, by using a FRE3DS's 3D visualization framework, interaction and visualization space is expanded more than the 2D glyph-based visualization counterpart.

2.2.2 Parallel Coordinate Visualization

Another 2D visualization technique is *parallel coordinate visualization* [28]. This technique consists of *n* parallel lines (axis), typically vertical and equally spaced. In network security applications, each vertical axis represents a network attribute. For example, the first, second, and third axes may respectively represent source IP, source port, and destination port. Each axis is connected via a line. This line denotes the relationship between two vertical axes and the color of the line may represent a filter for specific hosts or another attribute (e.g., green stands for TCP connection). An early work on parallel coordinate systems is VisFlowConnect [29]. VisFlowConnect focuses on enhancing an administrator's situational awareness by providing an easy-to-use, intuitive view of NetFlow [30] data using link analysis. NetFlow records as links between two machines or domains while employing a variety of visual cues to assist the user. Other works include Rumint [31] and Parallel Coordinate Attack Visualization (PCAV) [28], which discuss 2D parallel coordinates for detecting unknown large-scale network attacks including internet worms, DDoS attacks, passive fingerprinting [32] and network scanning activities. PCAV uses hash algorithms to detect nine graphical signatures using a detection algorithm in addition to visual human monitoring. Some researchers use techniques such as brushing [33] to give some insight into the behavior of individual source IP addresses. Brushing selects a specific coordinate or group of coordinates that focus on specific behaviors. However, brushing may become tedious when trying to select the behavior of one coordinate out of 1000s of multiple coordinates. In addition, 2D representations may be susceptible to occlusion-based visualization attacks. The research presented in this thesis investigates the usage of depth in stereoscopic 3D visualizations using the FRE3DS framework to help prevent these attacks.

2.2.3 Scatter Plots

A scatter plot displays a collection of points. Each point contains the value of two attributes determined by the position along the x axis and y axis. An earlier work of scatter plot is NVisionIP, which uses a scatterplot of a class-B network to allow analysts to quickly visualize the current state of their network [24]. Another work, Scanveiwer [34], combines scatterplots, parallel coordinates, histograms and color maps into a single tool. However occlusions, due to large volumes of datasets, result in cluttered visualizations and may cause data to be overlooked. In contrast to these techniques, P3D is introduced from the FRE3DS framework, which uses an awareness region mechanism to highlight important data and expand the visualization to help prevent occlusions.

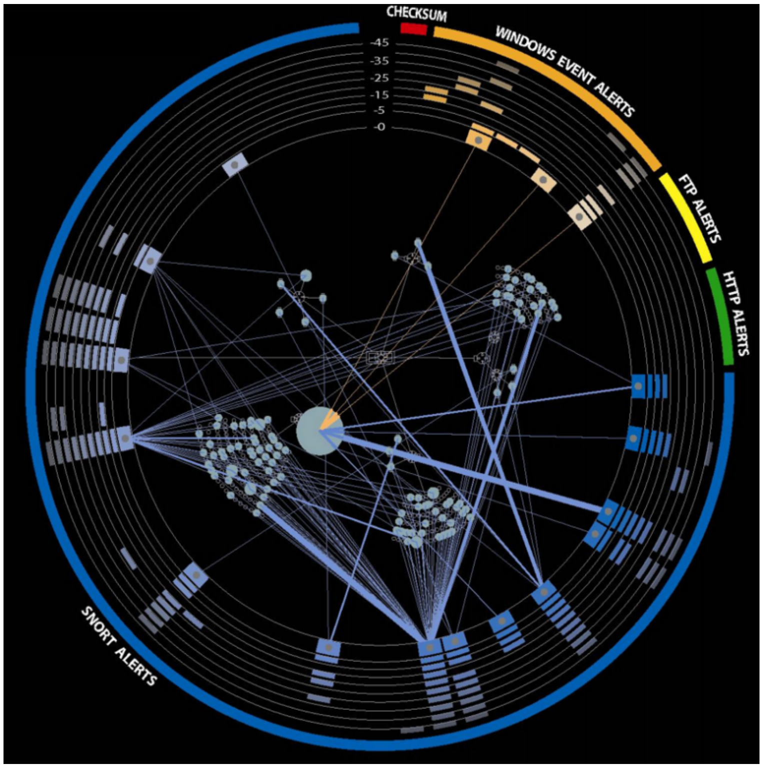


Figure 2: VisAlert [2]: A visualization paradigm for network intrusion detection.

2.2.4 Radial visualizations

Radial visualizations place visual elements along a circle, ellipse, or spiral on a screen. This layout allows for data to be encoded on both the outer and interior parts of a ring. The benefits of radial visualizations are its aesthetic appeal and compact layout for user interaction. Within radial visualizations, data is grouped using sections of a ring. In IDSs, these sections may represent categories of IDS alerts (Windows, FTP, HTTP, and Snort [35] alerts). Other past works include VisAlert [2] (as shown in Figure 2), NetSecRadar [36], and IDSRadar [37]. These works use radial visualization to show the global relationship between node topology and alert activity. While radial visualization provides more global insights of a network, more granular information is needed to investigate detailed network attributes and potential attacks. Through the FRE3DS framework network administrators could be able to extend global radial visualizations to further investigate the details of activity alerts and node topology.

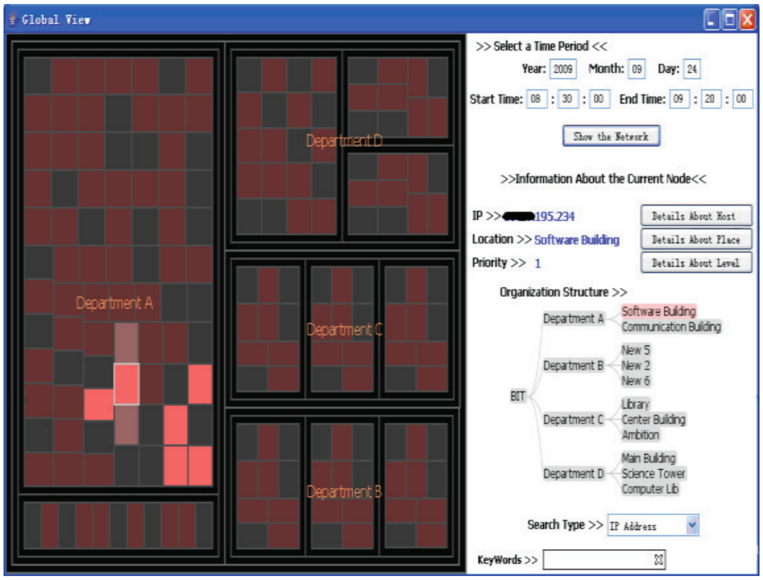
2.2.5 Treemaps

Treemaps use nested rectangles to portray hierarchical relationship. For example, NetVis [3] (Figure 3) uses a treemap visualization to combine the network security techniques and general network management in an integrated visualization. Within NetVis, a treemap expresses a global view of a network situation in an organization. The leaves of a treemap represent the hosts in the organization's network. The light-colored nodes show alert activity for this host in the network while dark-colored nodes illustrate a host without any alert. Treemaps are useful in displaying hierarchical relationship such as nodes within a network. However, treemaps are limited to hierarchical data such as IP address space and the relationship of high dimensional network attributes is lost. In this thesis, the FRE3DS framework contributes developing highly dimensional techniques that could be used to complement treemaps and allow for the highly dimensional attributes.

Although this is not a definitive list of 2D visualization techniques, it provides fundamental insights into the benefits and disadvantages of existing 2D visualizations. Although 2D visualizations are widely adopted and familiar to a user, 3D visualizations is emerging into network security and could be used to aid in preventing clutter and information overload to a user.

2.3 3D Visualization and Related Work

Recently, the gaming, television, computer-aided design, medical, and video graphics industries introduced stereoscopic 3D technologies to enhance the perception of depth. Stereoscopic refers to the use of 3D glasses to enhance the perception of depth by using *binocular disparity*. According to *MarketsandMarkets*, a marketing research firm, the global 3D technology-products and applications market is expected to reach $227.27 billion by 2016 [38]. Since the 3D market is growing rapidly in the upcoming years, 3D technologies are becoming more readily available and the latest monitors already are manufactured with 3D stereoscopic vision capabilities. As a result, security interface designers should consider



>> Select a Time Period <<

Year: 2009 Month: 09 Day: 24

Start Time: 08 : 30 : 00 End Time: 09 : 20 : 00

>>Information About the Current Node<<

IP >>[redacted]195.234

Location >> Software Building

Priority >> 1

Search Type >> IP Address

KeyWords >>

Figure 3: NetVis [3] visualizing hosts in the organization's network.

designing stereoscopic 3D security tools for complex tasks, large node sets, and important vulnerability data. It has been shown that stereoscopic 3D is superior to any monoscopic 3D viewing and to any shadow condition for enhancing accurate positioning and resizing tasks of objects located in 3D space [39]. Thus, network security could benefit from the creation of a stereoscopic tool that could potentially reduce error and enhance response rates. As will be explained later, binocular disparity is used in this thesis to enhance vulnerability awareness, decrease response times for detecting vulnerable nodes, and prevent *occlusion attacks* [40] that confuse and mislead network administrators.

An early work of 3D visualization is Tudumi [4] (as demonstrated in Figure 4). Tudumi monitors and audits user behavior on a server by visualizing connections using line patterns and system nodes. These system nodes are displayed with 3D glyphs on multi-layered concentric disks. Line patterns are used to encode different access methods such as coarse dashed lines to represent a terminal service and thin dashed lines to represent file transfer. The work in this thesis builds on Tudumi by including stereoscopic 3D technologies to perceptually increase the user space and enhance situational awareness.

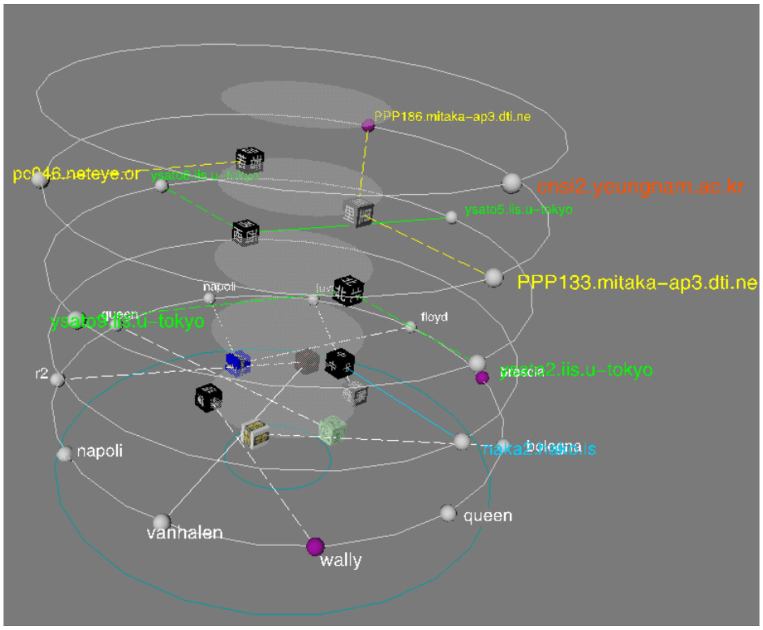


Figure 4: Tudumi [4].

Another example is Scapy's trace 3D function [41], which visualizes a 3D representation of traceroute data based on linked graphs. Likewise, Ipv6World [41] uses a similar method to visualize an IPv6 topology. Both Scapy 3D and Ipv6world portray a Python-based Real-Time 3D visualization of linked Graphs (RT3DG).

PortVis [42] is a visualization tool that aids in detecting large-scale network security events and port activity. NetBytes Viewer [43] visualizes the historical network flow data per port of an individual host machine or subnet on a network using a 3D impulse graph plot. These tools only consider the 4-tuple: source IP, destination IP, source port, and destination port. Thus, these security events show a small amount of detail and only display the counts of activities rather than the activities themselves. FRE3DS framework could enhance NetBytes Viewer by incorporating detailed packet header TCP fields such as RST, FIN, ACK, SYN and fragmentation bits. Various scanning events, such as stealthy intrusions at a firewall and IDS can also be detected and identified.

The Spinning Cube of Potential Doom [5] (Figure 5) uses 3D scatter plots to represent network activity on three axes: the destination IP of the local network on the x-axis, the

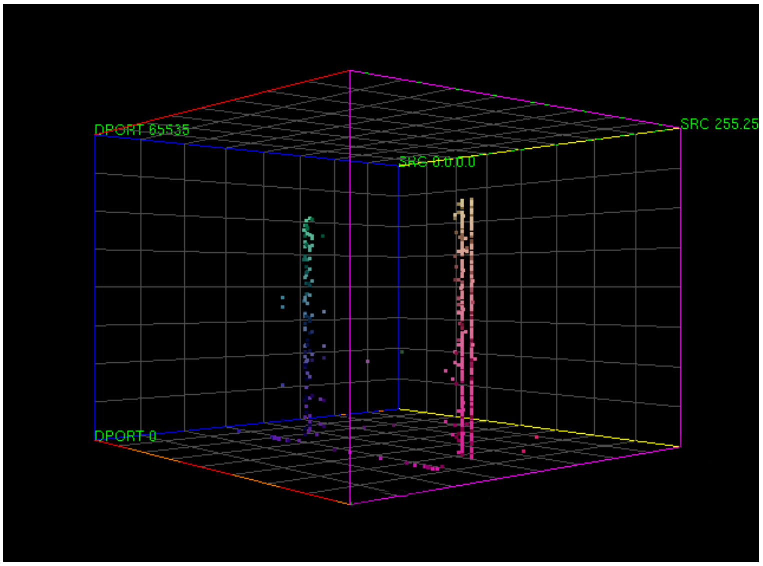


Figure 5: Spinning Cube of Potential Doom [5]

destination port on the y-axis, and the source IP on the z-axis. The color of the glyphs distinguishes the type of the connection (e.g., UDP or TCP). Their 3D scatter plots are useful in determining interesting patterns such as clusters or correlations for data using five parameters: source and destination IPs, source and destination ports, and connection type. Since the visualization is limited to five parameters, decoys cannot be detected without more parameters such as TCP flags and flow data. As a result, a deeper analysis of scanning behavior is not possible. The FRE3DS framework addresses these limitations by visualizing and incorporating more data to help uniquely characterize port scans and further understand scanning activity. Additionally, a module of the FRE3DS framework uses a stereoscopic region to increase awareness and reduce data overload.

Nessus 3D [44] is a node-based visualization that shows the number of vulnerabilities per node, TCP and UDP blacklisted connections, and patch updates. Although Nessus 3D depicts vulnerable nodes on a network, the FRE3DS framework differentiates itself from Nessus 3D by correlating vulnerabilities of multiple nodes across multiple exploits and categorizing the vulnerabilities for simpler evaluation. Thus, the FRE3DS framework provides tools and techniques for a better depiction of overall vulnerabilities on a network and provides the opportunity to prioritize patch updates. Furthermore, this framework groups nodes based on vulnerabilities and adds a stereoscopic 3D component for enhanced situational awareness of vulnerable nodes on a network.

Existing 3D visualizations have also been created to visualize data from IDSs [45] using techniques such as iconic tree structures, bar charts [46], and 3D scatter plots [5]. In addition, researchers have used various techniques to represent a larger number of attributes such as the size of a packet's payload in bytes, the number of packets, and inter-arrival time. The primary benefit of these visualizations is that they adequately portray generalizations of a network's behavior. However, they do not consider the error due to small subtleties in various attributes that could potentially be addressed using stereoscopic depth cues. For example, Visual Autonomous System Topology (VAST) [47] uses link graphs to extract information from Border Gateway Protocol (BGP) which route messages to assist in understanding topological properties of the Internet and Autonomous System's (AS) behavior. VAST uses quad-tree based visualizations to represent Autonomous System Numbers (ASNs) on a 3D plane. This tool successfully shows leaks from one AS to another AS. However, when a large number of ASes are present, it is a challenge to determine the depth (location) of an ASN in the VAST visualization due to a lack of visual cues.

As illustrated by these works of 3D visualization techniques, 3D visualizations within network security are still burgeoning due to the challenges presented by projecting depth on a 2D screen. Accurately depicting depth on 2D screens requires 3D interface designers to use various psychological and cognitive properties, also known as depth cues. In other words, since displays are physically constrained to 2D projections, these depth cues create a perception of 3D objects on a 2D plane. Commonly, to represent network security data, objects become 3D items such as spheres in 3D link graphs or points in 3D scatter plots. Additionally, depth cues assist users in easily locating, manipulating, and depicting spatial relationships between given 3D objects.

For 3D visualizations, depth cues are grouped into two categories: monocular and binocular. Monocular cues are depth cues that require only one eye to depict depth whereas binocular depth requires two eyes to depict depth. Some well-known monocular cues in network security are perspective, size, texture, occlusion, and shadows [48]. If these cues are used correctly, then obscurities and confusion in network security visualizations can be reduced. However, after performing a preliminary scan of past research and shown in Table 1, some 3D network security visualizations lack these depth cues and potentially result in higher error rates and slower decision times. For example, if IP addresses are represented as spheres, and the size of the spheres represent the amount of data entering the node, the node's information cannot be accurately portrayed without a visual cue such as shadowing to denote where the object is in respect to other objects. In the FRE3DS framework, both monocular cues are used such as perspective, size, and occlusion and binocular cues such as binocular disparity to reduce error in 3D network security visualizations. Thus, allowing users to identify network data more quickly and to accurately display complex information.

Table 1 shows a collection of network security tools and their associated visual cues. Below is an explanation of monocular cues.

Table 1: Visual cues for network security.

Utilization of Visual Cues in Network Security Tools	
Monocular Cues	***3D Visualization Tool***
Perspective	[49, 46, 50, 47, 51, 52]
Size	[49, 46, 51]
Texture	[49, 38]
Occlusion	[53, 49, 46, 50, 38, 47, 45, 51, 52]
Shadows	[49, 46, 38]
Motion Parallax	None
Binocular Cues	***3D Visualization Tool***
Binocular Disparity	[53]

- *Perspective* is the notion that parallel lines moving towards infinity converge to a point on a 2D plane. For example, parallel train track rails appear to meet at the horizon. Perspective is commonly used in network visualization to add more visualization data.

- *Size* refers to the relative position of the two objects of the same known size. If two objects are known to be the same size at the same distance and one object is positioned at a closer distance, the second object's size appears to be larger relative to the other object.

- *Texture* represents the level of detail used to represent an object. As objects move closer, the texture becomes clearer. However, as objects move away, the texture appears obscure.

- *Occlusion* is the partially or complete blocking of one object by another object. If one object completely blocks a second object, then the second object can potentially be overlooked.

- *Shadows* occur when the shadow of an object is visible on the object or on different objects. Shadowing could denote position of an object. For example, if an object is in front of another object, its shadow will also be in front of the second object's shadow.

- *Motion parallax* refers to the spatial properties within motion. The movement of the camera or the object can give spatial properties about the 3D location of the object. When an observer moves, absolute depth information of the distance can be determined from several stationary objects if the velocity and the direction are known. Closer objects appear to pass more quickly than objects further away.

In the next section, binocular cues that require two eyes to depict depth for stereoscopic 3D visualizations will be discussed.

2.4 Stereoscopic 3D Visualization Overview and Related Work

As shown in Table 1, many security tools use monocular cues. However, a small number of tools currently use binocular cues for steroscopic technology. A limited amount of works discuss stereoscopic technology in network security. Among a few are Papadopoulos' work which discusses CyberSeer [53], a desktop, interactive, immersive, auto-stereoscopic 3D environment. The environment is integrated with multi-channel immersive sound to enhance security awareness. It introduces a 3D auto-stereoscopic environment to analyze spatial information for intrusion detection [53]. In addition, Ipv6world, which presents a 3D link-node graph of the topology of IPv6 routers, [41] recently added a stereoscopic feature using red/cyan anaglyphs. CyberSeer is the closest work to the work presented in this thesis, however it is limited to only visualizing IDSs. Compared to this tool, the FRE3DS framework will allow better integration with future tools to produce a more holistic 3D stereoscopic tool set.

In addition, binocular disparity is used in the FRE3DS framework to enhance the perception of depth for stereoscopic 3D visualizations. As a synopsis, three types of binocular cues are discussed which are *binocular disparity*, *convergence*, and *accommodation*. *Binocular disparity*, also called binocular parallax, uses the notion that each eye within the visual system views two slightly different retinal images. Once the brain processes these images, it appears to give the illusion of depth. *Due to these depth illusion qualities, binocular disparity can be used for network security in situations where monocular depth cues do not adequately reveal enough information about the network's security posture.* Binocular disparity is a primary physiological characteristic that enables the stereoscopic viewing of objects, within a limited distance, and is widely used for portraying virtual objects (e.g., images on a computer screen) in real 3D space. *Accommodation* refers to the physical adjustment of the ciliary muscles in the eye when moving the focus on particular objects. When focusing on far objects, the lens in the eye decontracts and increases the focal length. *Convergence* is the inward movement of the eyes in an effort to maintain a single binocular vision of an object. In contrast, using accommodation and convergence to portray virtual objects in real 3D space is a challenge because both require a physical object to be present. However, through the use of binocular disparity, specifically in stereoscopic

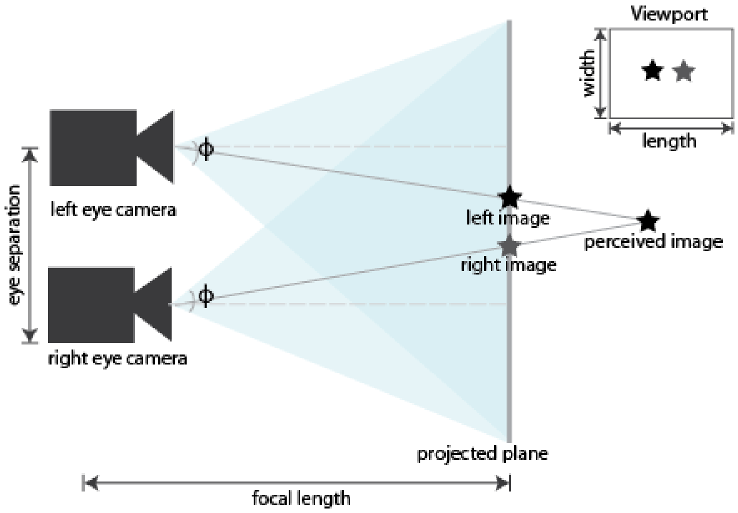


Figure 6: 3D Stereoscopic rendering for an image using left and right image.

3D, a physical object does not have to be present for a virtual rendering.

As shown in Figure 4, the stereoscopic rendering environment consists of two cameras: the right eye camera and the left eye camera. The usage of cameras is commonly used to create the environment in all 3D stereoscopic applications including 3D movies and 3D software. Each camera is separated with an average eye separation of 6.2 centimeters to mimic an average human eye separation. The cameras are positioned parallel to each other and perpendicular to the projection plane. In stereoscopic 3D, both the projection plane and the viewport are considered as the physical monitor. Likewise, the width and length of the viewport represents the length and width of the computer screen in pixels. Before each visualization is rendered, each camera creates an off-axis frustum with the projection plane. The viewing angle or *fovy* of the camera is denoted by the lowercase ϕ symbol as shown in Figure 6. The distance between the camera and the projection plane is the focal length. The left and right cameras produce a left and right image, respectively. As shown in Figure 4, the left and right images are two slightly unique perspectives of one image and this image is perceived to be behind the screen. This concept is used in network security tools when generating a stereoscopic 3D environment.

2.5 History of Interaction and Stereoscopic 3D Visualizations

Although the primary focus in VizSec is 2D/3D visual representation, user interaction also plays a significant role. If the usability of VizSec interfaces is increased, the response time to detect network incidents could potentially be decreased. Furthermore, user error could also be decreased. Thus, researchers introduced Human and Computer Interaction Security (HCI-Sec). HCI-Sec is the study of one or more human interactions between one or more computers as it pertains to VizSec. The aim of HCI-Sec is to improve the usability of security features in end-user applications [54]. Using HCI-Sec, researchers analyze and design Graphical User Interfaces (GUIs) to achieve a more secure and usable system [55]. For example, a user may inadvertently misconfigure the security features such as firewalls due to the design of the interfaces. Rather than focusing on the GUI design, the FRE3DS framework builds upon HCI-Sec by developing an interaction system to help network administrators reduce the number of interactions with network security 3D visualization tools. By focusing on the network administrator's interaction, the network administrator can detect suspicious network activity more efficiently than traditional methods, independent of the end user's configuration or design. This research could help network administrators reduce response times to mitigate security threats and discover more attacks. Furthermore, most work in HCI-Sec discusses how the design of 2D interfaces can be improved, but seldom evaluates the usability of an interface for natural user interactions. This thesis uniquely discusses how the interactions are used to increase productivity for 3D interfaces by using recommender systems and NUIs.

A substantial amount of stereoscopic work has been done in the areas of human computer interaction (HCI) [39] and robotics [56, 57]. Yet, currently, no research exists that uses stereoscopic 3D to provide situational awareness for network security applications. In the field of HCI, researchers shows that stereoscopic 3D is superior to monoscopic viewing, with or without shadow conditions, for enhancing positioning and resizing accuracy and response time [39]. The use of stereoscopic 3D attributed to a 22% reduction time compared to the use of non-stereoscopic visualizations while performing positioning tasks. With stereoscopic 3D, individuals can perceive large amounts of visual information, especially if 3D binocular senses are present. Thus, network administrators can manipulate multidimensional data and transform it into a simplified representation for easy analysis. This visualization is useful in network security tools that give an overall state of the network and allows the user to manipulate views of data and analyze network data at different levels of granularity. Other stereoscopy work in HCI domain focuses on examining the human error and response times for tracing link-node graphs [58]. The work in Ware [58] shows that 3D depth cues allowed participants to see paths in graphs containing 333 nodes with better than 92% accuracy. Also, it showed that stereoscopic conditions resulted in the shortest response times. Additionally, stereoscopic conditions possess substantially lower percent error rates than non-stereoscopic conditions for large node sets, particularly 333+ nodes for non-skilled users. Skilled observers could see up to a 1000-node graph with less than a 10% error rate. This is an order of magnitude better than the error rate of 2D visualizations. Also, tracing node-link graphs is commonly used in network security visualizations such as those that visualize IPv6 topologies [41, 46, 47]. Other 3D stereoscopic techniques have been used in robotics, construction, and teleoperation applications (for performing dexterous tasks to control machines in real-time from a remote location). Within teleoperation applications, the remote operators were asked to achieve the requested pick-and-place task swiftly, without any collisions with obstacles. The results showed that the operator saved more than 60% of his/her time when completing pick-and-place tasks with 3D stereoscopic visual feedback than its 3D monoscopic counterparts [56, 57]. Thus, 3D stereoscopic spatial positioning of objects can provide better efficiency using depth in 3D stereoscopic viewing in visual feedback systems.Within the FRE3DS framework, HCI and robotic research are applied to network security by adopting 3D stereoscopic viewing techniques to enhance situational awareness, help reduce human error, and increase response times of network security administrators.

2.5.1 Recommender Systems

Recently, recommender systems have been used in recommending products and services such as movies, books and music. Through the popular use of recommender systems, referrals are generated to a user by correlating identified user interest to similar interest of others to increase sales. However, these systems have yet to be adopted by network security applications. Therefore, a module within the FRE3DS framework called NAVSEC was developed. NAVSEC is a recommender system prototype for network security applications. NAVSEC combines the interaction behavior of both an expert user (i.e., skilled network administrator) and a novice user to assist in discovering network based attacks. Other systems (e.g., [59]) recommend a single interaction for software applications such as AutoCAD. On the contrary, NAVSEC recommends a sequence of interactions, which can be executed by a single advanced action. As a result, my technique is instrumental in reducing the number of interactions a novice user might use to render a visualization. NAVSEC can lead to attack discovery with fewer interactions and to more efficient utilization of resources (e.g., memory and CPU utilization).

2.5.2 Natural User Interfaces (NUIs)

Traditional GUIs adopt mouse and keyboard interactions, which use artificial elements like windows, menus, or buttons. On the other hand, NUIs adopt a direct manipulation style (e.g., touch, voice commands, and gestures). NUIs are useful because NUIs takes a user's pre-existing knowledge about manipulating objects in the real world for application in computer technologies. As a result, this technology makes NUIs easy-to-use and easy-to-remember [60]. Some researchers are beginning to deploy NUIs into visualizations and network security applications [61]. One researcher used multi-touch interaction for brushing in parallel coordinates [61]. My research adopts a gesture set and integrates these gestures into an interaction system for network security applications. As a result, the FRE3DS framework introduces new gestures into the gesture set and includes interactions from other devices (e.g., Kinect). Other researchers have attempted to develop NUI tools, such as using the Kinect device, to perform network attacks. For example, Kinectasploit [62] uses a 3D virtual environment to test security systems for vulnerabilities by interpreting Kinect's natural gestures into a series of Metasploit Framework [63] commands. The FRE3DS framework applies this concept more generally to the research field of network security.

CHAPTER 3

DESIGNING A FRAMEWORK FOR RENDERING ENHANCED 3D VISUALIZATIONS

As mentioned in Chapter 1, VizSec is rapidly maturing and will continue to mature in the upcoming years due to the emergence of new protocols and growth within the network security domain. Furthermore, researchers are applying many techniques and tools from VizSec to the problems of network security, specifically in network traffic analysis. However, while the design of network tools are founded on the problems of real world use cases, many tools are rarely tested empirically for usability. In addition, it is difficult for developers of network security tools to stay abreast, evaluate, and integrate the current advances in interaction and visualization technology (e.g., stereoscopic 3D and multi-touch technologies) into network security applications. In this section, I present the FRE3DS framework to assist developers in producing novel tools and techniques using state-of-the-art technologies and utilize this framework to study the effects of emerging stereoscopic and multi-touch technologies. In an effort to verify the aforementioned contributions of the FRE3DS framework, four prototypes were developed to prevent occlusion attacks (i.e., visual information is intentionally overwritten), increase situational awareness, reduce interactions using gesture sets, and reduce response times by using a recommender system. The four prototypes are 3DSVAT for analyzing vulnerability data on a local area network; P3D for identifying distributed scanning attacks intended to thwart network administrators; NAVSEC for providing a system to recommend interactions to novice users; and InterSec for decreasing interactions using NUIs. In the following section, the design, implementation and evaluation process for the FRE3DS framework is discussed.

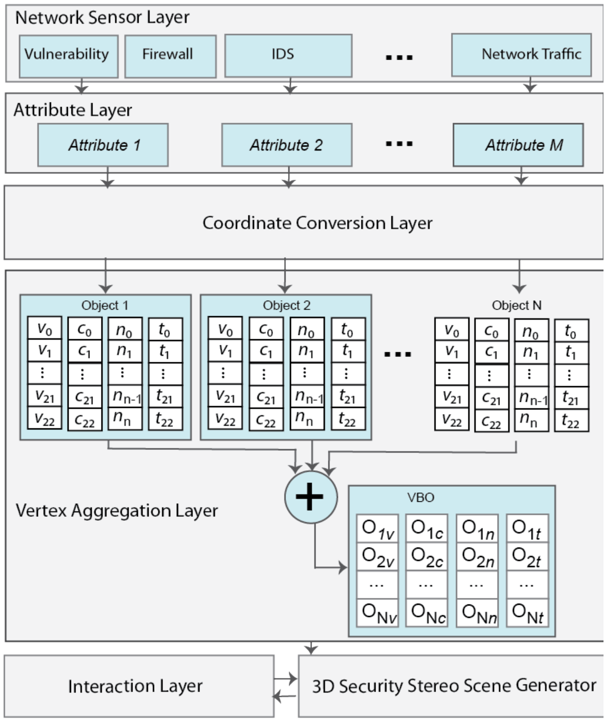

Network Sensor Layer
Vulnerability
Firewall
IDS
...
Network Traffic
Attribute Layer
Attribute 1
Attribute 2
...
Attribute M
Coordinate Conversion Layer
Object 1
Object 2
Object N
v_0 c_0 n_0 t_0
v_1 c_1 n_1 t_1
v_{21} c_{21} n_{n-1} t_{21}
v_{22} c_{22} n_n t_{22}
...
Vertex Aggregation Layer
+
VBO
O_{1v} O_{1c} O_{1n} O_{1t}
O_{2v} O_{2c} O_{2n} O_{2t}
...
O_{Nv} O_{Nc} O_{Nn} O_{Nt}
Interaction Layer
3D Security Stereo Scene Generator

Figure 7: FRE3DS layer model.

3.1 A Layer Model for the FRE3DS Framework

The FRE3DS framework uses rapid prototyping for 3D network security visualizations with stereoscopic and multi-touch support. FRE3DS framework is useful for producing rapid customized 3D visualizations for network administrators to easily and quickly develop various visualizations and efficiently investigate data. The framework uses a layered approach as shown in Figure 7. The primary concept of the framework is to receive raw data as input from a network and output a 3D visualization with stereoscopic support. Furthermore, a 3D visualization could be manipulated and evaluated using both traditional mouse/keyboard technology and non-traditional NUI technology.

The framework is divided into 5 layers: Network Sensor Layer, Attribute Layer, Coordinate Conversion Layer, Vertex Aggregation Layer, 3D Security Stereo Scene Generator, and an Interaction Layer. Each layer is described below.

3.1.1 Network Sensor Layer

The first layer of the framework, the Network Sensor Layer (NSL), is used to collect data from various systems including vulnerability scanners, firewalls, IDSs, keyloggers, and network traffic analyzers. The raw data is often represented in the form of pcap and log files. In real time scenarios, the sensors may produce packets and stream the packets to the NSL via a socket connection. Essentially, most network sensors can plug into the FRE3DS framework as modules for easy data visualization. Each module parses into a standardized readable format for the FRE3DS framework.

3.1.2 Attribute Layer

As portrayed in Figure 7, the NSL provides formatted data to the *Attribute Layer* (AL). The AL receives, filters, and stores relevant data such as an alert to a firewall log or port numbers as attributes into a storage location (e.g., MySQL database) for quick and easy retrieval. An attribute is defined as a characteristic that describes the network behavior. Although the current prototype stores the attribute in a MySQL database, the FRE3DS Framework could easily be expanded to include NoSQL technologies, such as MongoDB, to increase the capacity for concurrent users and data storage and retrieval.In addition, the storage of each attribute is tagged with a timestamp to ensure past network incidents may be examined at any time by users.

3.1.3 Coordinate Conversion Layer

After an attribute is produced and archived, it is sent to the *Coordinate Conversion Layer* (CCL). The CCL converts the security attributes into 3D environment coordinates. Each coordinate is determined by the type of visualization and depends on the visual representation for each attribute. After each attribute is converted into a coordinate, each coordinate is converted into objects. Objects are actually visual representations for a particular attribute such as cubes, lines, and planes. Each object contains a vertex array (v_0, v_1, $\ldots$,v_n) , a color array (c_0, c_1, $\ldots$, c_n), a texture array (t_0, t_1, $\ldots$, t_n), and a normal array (n_0, n_1, $\ldots$,

n_n) as seen in typical 3D graphics rendering. The vertex array consists of the vertices of objects being displayed. For example, if an IP source address is represented as a cube, then its vertex array contains 24 vertices created from one (x,y, z) coordinate. The color array contains the color coordinates of the object. For example, a high threat vulnerability object can possess color coordinates as red.

The texture coordinates are coordinates for applying a bitmap image to a surface of an object. Texture coordinates are beneficial when OS logos are textures on objects for OS fingerprinting. Normal coordinates are vectors that are perpendicular to the surfaces of the object and used to enhance lighting and shadowing depth cues. Each object is aggregated into a memory allocation array of vertex arrays and sent to the *3D Security Stereo Scene Generator* (SSG).

3.1.4 3D Security Stereo Scene Generator

The SSG adjusts the OpenGL rendering pipeline using quad-buffer technologies. SSG takes the object data and passes it through the OpenGL rendering pipeline. The SSG generates two separate rendering pipelines for each eye. In addition, the SSG coordinates works in tangent with the interaction layer to regenerate the screen based upon new user input. Once the two renderings are generated, the right and left visualizations are stored to right and left back buffers. When the object is rendered to the page, the right and left back buffers swap with the right and left front buffers. As a result, the security visualization is presented to the user. The pipeline creates two cameras, a left and right camera, each separated by a distance of 6.3 centimeters, with an off-axis frustum and positioned at the focal length of the screen. A camera's frustum is commonly used in computer graphics to describe a 3- dimensional region which is visible on a screen. The focal length of the camera refers to if an object is positioned with a positive z-axis value, the object is positioned within the focal length of the user's eye and appears to be in front of the 3D monitor. This visualization concept is used to display vulnerable nodes in front of the monitor for rapid vulnerability detection and to help prevent visualization occlusions for distributed network attacks. Moreover, the user can manipulate and interact with the 3D interface by zooming, panning, and selecting rendered objects by taking the newly generated vertex data and passing it back through the pipelines.

3.1.5 Interaction Layer

The Interaction Layer (IL) is used to manipulate an environment produced by the SSG. The IL provides traditional and alternative ways for a user to communicate with the visualization system, which include mouse/keyboard, multi-touch, and motion-sensing (i.e., Kinect) technologies. A variety of interaction techniques promote new alternative NUI designs as well as provide a platform for evaluation and allows a user to choose the appropriate technique for his/her tool. In the IL, a plugin is loaded to provide input from NUI sensor devices into FRE3DS framework. The input data is simply coordinates from actions performed by a user. For example, a LEAP Motion [64] plugin sense hands and fingers at close range (within 12 inches of a sensor) in 3D space and convert the coordinates from the fingers into gestures. The IL takes the gestures and maps it to a interaction. Next, the produced interaction is used to manipulate the visualization. After the interaction is executed, it is archived. If the NAVSEC module is enabled, the archived interaction could be utilized to recommend interactions by using the current user's historic interactions and interactions from a community of expert users. As discussed in greater detail later, the NAVSEC plugin within the IL could potentially assist confused novice users.

3.2 System Implementation and Testbed

The FRE3DS framework uses the C++ Object Oriented Model-View-Controller paradigm for higher modularity and extensibility. A custom OpenGL 3D widget is used within a QT framework for its cross-platform capability. Thus, the framework compiles and runs on Windows, GNU/Linux and Mac OS X operating systems. To render the content in stereoscopic 3D, Nvidia Quadro 2000, Nvidia RF 3D Vision Pro Shutter Glasses, and a 120 Hz Asus 3D monitor for 60Hz screen rendering per eye, as shown in Figure 8, were

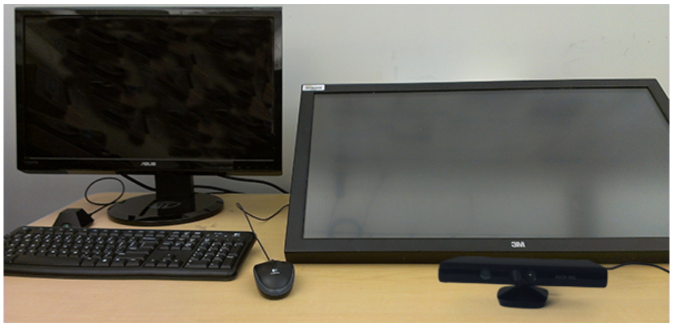


Figure 8: NUI visualization testbed for FRE3DS framework.

used for the 3D steroscopic rendering.

3.3 Evaluation Methods

This research introduces various stereoscopic and non-stereoscopic use cases to evaluate user error and task completion time. The evaluation investigates whether portraying network activity such as network scans or Denial of Service attacks (DoS) in three dimensions (3D) will reduce human error, decrease task completion time, and increase situational awareness. The evaluation hypotheses are tested using both use-case scenarios and empirical user testing. The purpose of this study is to provide empirically tested research that may help reduce error, enhance response rates, and increase awareness of peculiar network activity. In addition, a goal of this research is to provide methods for the network security research community and system administrators to interact with large amounts of data using stereoscopic technologies.

Although stereoscopic and interaction technologies has been evaluated in some HCI applications [39], this research has not been formally evaluated via user testing and analyzed for usage in network security applications. Thus, this research evaluates the benefits of stereoscopic 3D tools in network security applications utilizing user testing methods. With user testing, various network attack scenarios is analyzed using P3D and is determined if select framework's prototypes meet the intended purpose of reducing user error and response rates.

3.3.1 User Testing Methods

User testing is an evaluation method in user-centered interaction design to evaluate a Human Computer Interaction (HCI) technique by testing scenarios of the technique on users. User testing is commonly used for critiquing foods, evaluating new consumer products, and testing functionality of websites [65, 66]. My user testing method contained a group of network visualizations scenarios in which I asked users to analyze and identify malicious activity on the network. Each network visualization scenario ranged in difficulty from beginner to expert level. Some examples of scenarios were simple port scan, Windshield Wiper attack, Port Source Confusion attack, scans from 100 nodes on the network, and scans from 300 nodes on the network. A pre-survey was given (given in Appendix) to determine the expertise of the user. Each user was expected to possess at least a basic knowledge of networking. During each scenario, l recorded the time taken to successfully complete each scenario.

3.3.2 Testing Procedure

In this section, a brief overview is provided of the testing procedure for the selected participants. The testing procedure was approved by the Institutional Review Board (IRB). The testing procedure requires all participants to complete a consent form (given in Appendix). The consent form verifies that the user fully understands the purpose of the research, any risks associated with the experiment, and confirm their willingness to participate in the user testing session. Additionally, all information disseminated to the participants was scripted to ensure that the explanation of the experiment stays consistent and controlled to limit the occurrence of skewed results. Once the consent form was completed, a brief pre-survey was given to ensure the participants have an understanding of the subject matter and to assess the expertise level of the user. Using the pre-survey, the users were divided into the following categories:

- *Novice user* contains only basic knowledge of Internetworking and understands basic networking concepts such as IP address, MAC address, or port number. For example, if a person has taken a basic networking course, but has not pursued networking any further, then the user is considered a novice user.

- *Intermediate user* contains basic knowledge of Internetworking and may have completed projects within networking such as configured servers.
- *Advance user* fully understands the TCP/IP protocol stack and may have performed tasks such as built client/server programs and configured routers/switches. Within network security, an advanced user may have performed more advanced attacks such as Man-In-The-Middle (MITM) and ARP poisoning. An advanced user may have knowledge of multiple network security tools and many network attacks.
- *Expert user* possess extensive knowledge in network security and have performed research in the field. An expert may have published in the field or contain 5+ years of experience in a related field.

Within the pre-survey, a user was asked to list any related classes taken in the field of network communications and network security to further confirm the expertise of the user. At the conclusion of the pre-survey, the components of the user interface and visualization techniques were explained to the participants involved in the survey. In addition, various scenarios, attached in Appendix, were given to ensure that all participants understood the concept of the visualization techniques during the experiment. If, during the warm-up scenarios any participant made consistent inaccurate readings, the participant was considered as an ”inadequate user” and data for that user was discarded. Each scenario contained simulated or sample network traffic. For each scenario, users were expected to spend a maximum of 5 minutes. While each user performed each task, observation, note-taking, ”thinking- out- loud”, and other survey testing methods were used. Observation and note-taking allowed the observer to notice common mistakes of users, their sequence of logical choices and created a record of the session’s observations. The ”thinking- out- loud” method further enhanced my notes of the user's experience and led to possible layout reconstruction to decrease confusion during each experiment. Lastly, all participants were requested to fill out a post experiment survey. This survey served as a means to retrieve quality feedback such as information and comments from users that were not addressed during the experiment. The completion time for each user was 60 to 90 minutes and at least 15 participants with basic networking knowledge for networking courses were recruited from the Georgia Institute of Technology.

3.3.3 Evaluation Components

The evaluation plan contained both quantitative and qualitative components. each of the components are described below for the quantitative component:

- *Response time* refers to the amount of time to perform each task.
- *Percentage of task completed* is a metric measured by asking a participant to complete a list of questions after each of the tasks and evaluate the completeness of the answer. Each answer is given a score and all the scores for a user sums to 100 points.
- *User error* is measured by evaluating the user's sequence of interactions and comparing that to standard optimal routes. Also, verbal input from the user is used to measure the tasks.
- *Number of interactions* counts the sequence of interactions.
- *Sequence of interactions in conjunction with the "thinking- out- loud"* is used to understand the error and misconceptions of completing each task. Also, the sequence of user interactions is recorded to find which interactions are commonly completed in error.

For the quantitative component, each participant begins by pressing a start button, which starts the timer. As the participants perform their actions, they verbally explain them and press the finish button to stop the timer. Programmed hooks are used within each tool being evaluated to measure the quantitative component. For the qualitative component, each participant is requested to complete post-survey to evaluate user satisfaction with the interface used during the session. The user survey is given in the Appendix.

CHAPTER 4

3DSVAT: 3D STEREOSCOPIC VULNERABILITY ASSESSMENT TOOL

Currently, no visualization tool exists that focuses on enhanced network situational awareness for vulnerability correlations between nodes using stereoscopic 3D. Thus, a prototype was implemented from the FRE3DS framework called 3D Stereoscopic Vulnerability Assessment Tool (3DSVAT), which assists in rapid detection of vulnerable nodes using severity level as a function of depth. Furthermore, 3DSVAT uniquely visualizes the network topology based on a correlation between vulnerabilities and nodes on a local area network. As a result, 3DSVAT could provide it, among other capabilities, new insights for the deployment of patch updates.

4.1 3DSVAT System Design

3DSVAT uses the FRE3DS Framework to retrieve attribute input from the Qualys Guard Vulnerability [67] assessment tool, Common Exposure Vulnerability (CVE) list, and Exploit-db.com database, as shown in Figure 9. The CVE list is a dictionary of publicly known security vulnerabilities and exposures. It is published in formats such as XML or HTML for easy parsing of vulnerability scores and descriptions of the type of node vulnerability. The AL extracts the information from the NSL, parses the information, and filters the information into the following relevant attributes: IP, OS, Vulnerability Severity Score, port number, and the Common Exposure Vulnerability (CVE) identification number. Next, the CCL converts each attribute into 3D coordinates. The CCL converts severity scores into coordinates for bar graphs, OSes into textures, IP addresses into sets of cubes based on severity, and CVE identification numbers into scatter plots. Also, according to the severity scores, the CCL performs region conversion, which determines the coordinate for a region explained in later sections. After coordinates have been produced, the vertex aggregation

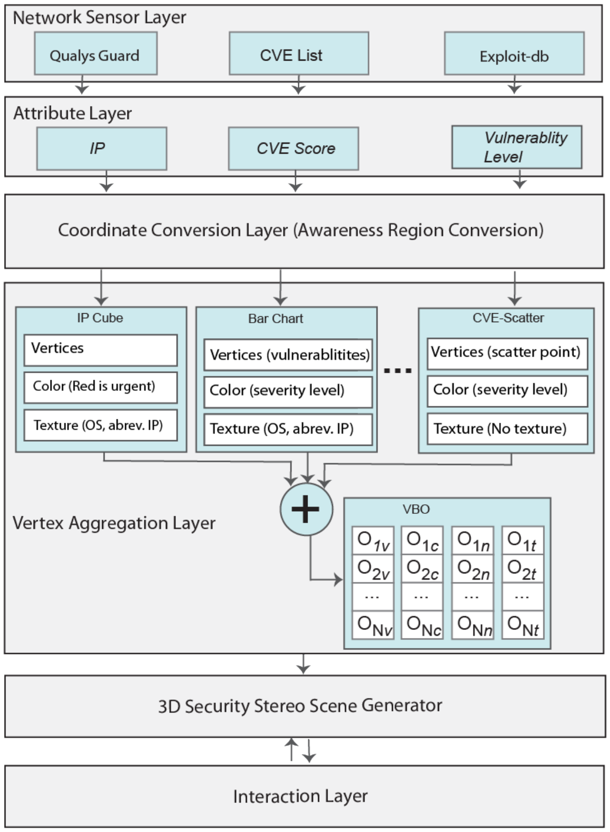


Figure 9: 3DSVAT, an implementation of the FRE3DS framework.

layer (VAL) combines these vertices and passes them to the SSG. The SSG creates the left and right cameras, off-axis frustum, and other components essential for rendering within an OpenGL environment.

3DSVAT distinctly shows a visualization from an aerial perspective based on x, y, and z coordinate systems. As shown in Figure 10, the 3D environment renders a stereoscopic visualization into three regions: the Grouped Vulnerability Region, Vulnerability Type Region, and Stereoscopic Region. Each region is described below.

4.1.1 Grouped Vulnerability Region

The Group Vulnerability Region portrays grouped nodes by highest vulnerability score similar to the Qualys Guard Vulnerability assessment tool [67]. These groups of vulnerabilities are arranged from severity level 1 to 4. The levels are described below:

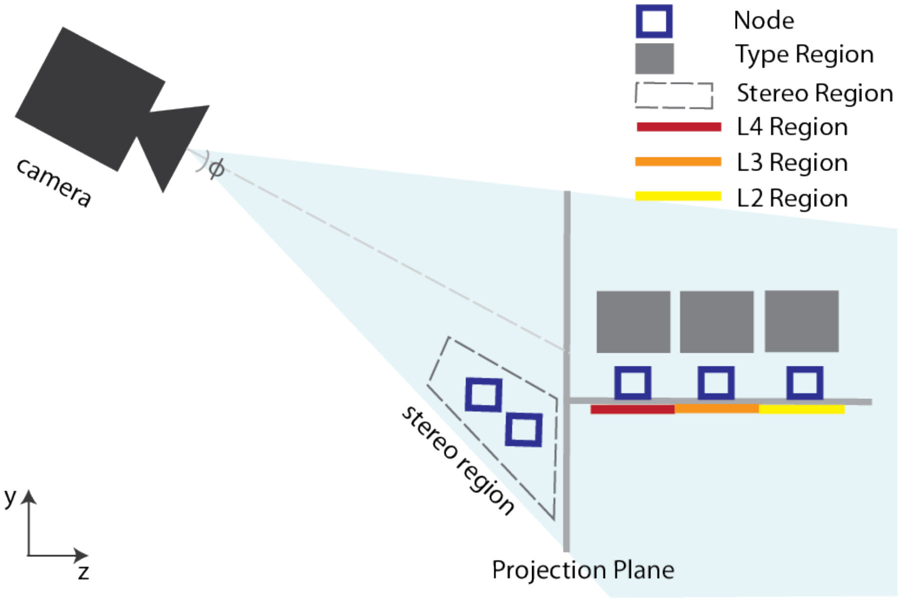


Figure 10: Side view of 3DSVAT visual layout.

- A system is labeled as critical (level 4), denoted by the color red, if its vulnerabilities allow the compromise of highly sensitive information on a system.
- A system is labeled as serious (level 3), denoted by the color orange, if its vulnerabilities enable intruders to gain access to specific information, potentially misuse information from the host, or allow unauthorized use of services such as access to certain files, Denial of Service (DoS) attacks, or mail relay.
- Medium and minimal levels (Level 2/1), denoted by the color yellow, are triggered if the nodes' vulnerabilities enable intruders to collect specific information about the hosts, such as version of software.

The group of correlated vulnerabilities allows the administrator to determine which nodes are the most vulnerable on a network and most common vulnerabilities between nodes. Furthermore, this allows administrators to prioritize which vulnerability to patch first. In addition to grouping, each node contains bar graphs showing the number of vulnerabilities of lower level grouped regions.

4.1.2 Vulnerability Type Region

The Vulnerability Type Region portrays how nodes in a grouped category correlate to specific vulnerabilities. A list of the common CVE identifiers is positioned horizontally along the y-axis and the number of nodes is positioned vertically along the z-axis. This region shows how specific vulnerabilities correlate to nodes on the network. For example, if there is only one point in the region, then all nodes share a single vulnerability. Furthermore, this section can be filtered based on types of vulnerabilities, such as buffer overflows or DoS attacks to introduce further details.

4.1.3 Stereoscopic Region

The Stereoscopic Region (denoted as Stereo Region in Figure 10) displays urgent vulnerabilities. The urgent vulnerabilities allow intruders to gain full control of hosts including full read and write access, remote code execution, and backdoors installations. Since the nodes are classified as urgent, the highest severity level,these vulnerabilities pose the most serious threats. As a result, stereoscopic technologies are used to enhance awareness of vulnerable nodes. These nodes are positioned within the focal length of the user. As a result, with stereoscopic technologies, the nodes within the Stereoscopic Region are perceived in front of the physical screen.

4.2 Use-Case Analysis

The following figure shows a visualization of an 18-node LAN network using 3DSVAT. The vulnerability data is a subset of data taken from a large production network and modified to portray important capabilities of the 3DSVAT prototype l in a 192.168.3.0/24 subnet. This visualization shows some interesting correlations between the nodes and their vulnerabilities by introducing the z-direction. In contrast to 2D visualizations, this tool promotes scalability by visualizing multiple grouped LAN networks. In Figure 11, each node is grouped by the highest vulnerability level it possesses and is positioned along the z-axis.

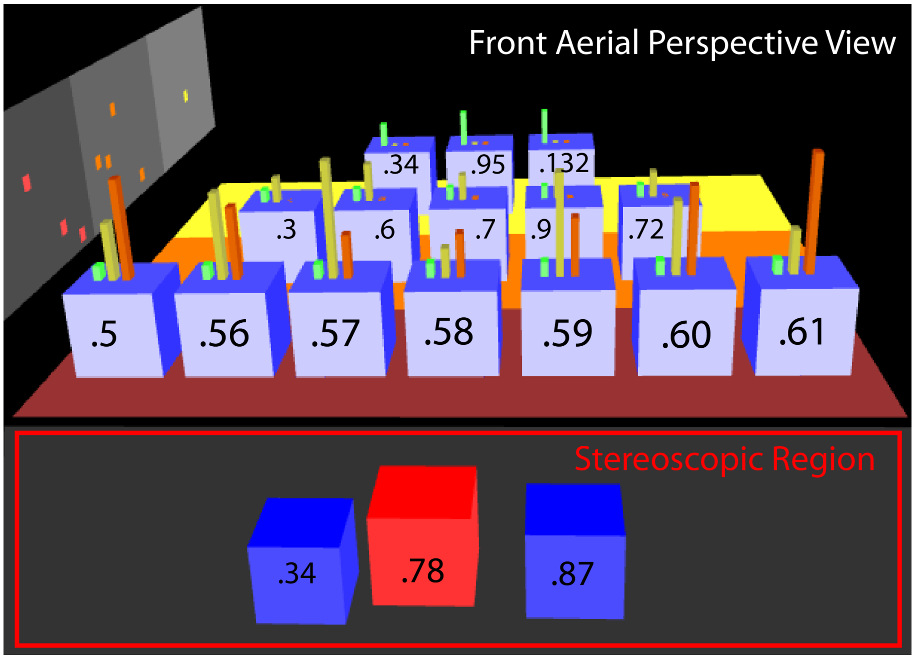


Figure 11: 3DSVAT aerial front-view visualization for 18-node LAN.

In this visualization, three nodes demonstrate level 5 vulnerabilities as the highest vulnerability level. Level 5 nodes are located in the stereoscopic region and when viewed with stereoscopic glasses, these three nodes (192.168.3.34, 192.168.3.78, and 192.168.3.84) are perceived to be in front of the monitor to increase the awareness of the network administrator.

A side view figure is given to better portray the location of the extremely vulnerable nodes within the visualization. Accordingly, they are perceived to be closer to the user. Also, as denoted in Figure 12, each node is categorized based on the highest vulnerability level it possesses. For example, 7 nodes possess level 4 vulnerabilities as the highest vulnerability level. Likewise, 5 nodes possess level 3 as the highest vulnerability level. If a level 5 node contains more than one vulnerability, the node is positioned closer to the user. The red node (192.168.3.78) represents a node where root access exploits can be quickly found by performing a simple Internet search. The red node list was compiled by comparing CVE data to well-known penetration testing sites.

In addition, the vulnerability type region demonstrates the specific vulnerabilities and

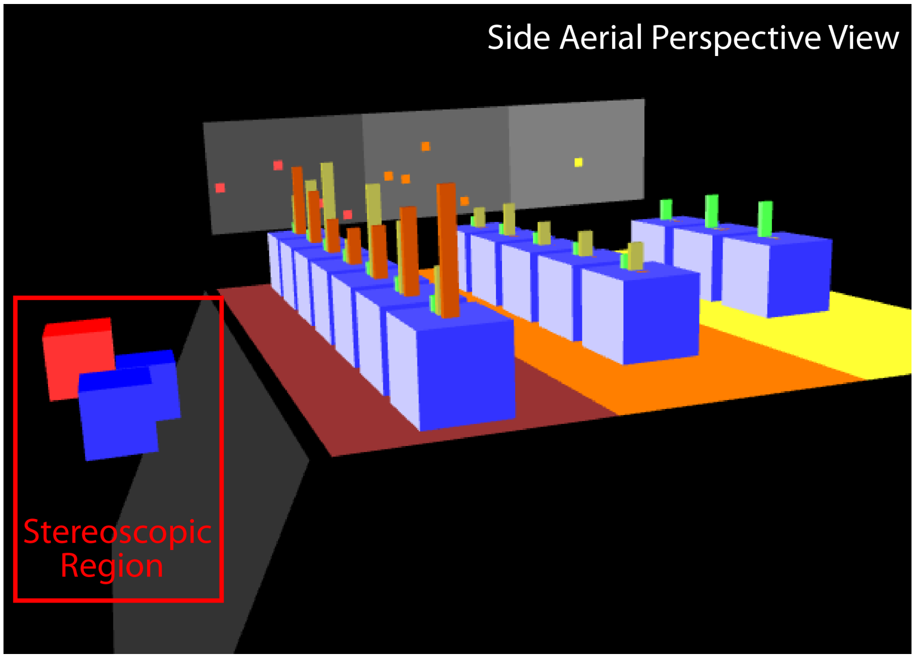


Figure 12: 3DSVAT aerial side-view visualization.

how they correlate to the nodes in a group. For example, the yellow level 2 nodes 192.168.3.34, 192.168.3.95, and 192.168.3.132 share the same CVE-2002-0510 vulnerability type where all hosts are transmitting UDP packets with a constant IP Identification field. As a result, an attacker can fingerprint the operating system version and approximate kernel version of the three vulnerable systems. Within the level 4 nodes, multiple nodes share multiple vulnerabilities across multiple levels. Three level 4 nodes have vulnerabilities that can allow an attacker to use the NetBIOS access to steal a remote user list of authenticated accounts on the node, including guest accounts. In addition, level 4 nodes 192.168.3.6, 192.168.3.6.7 and 192.168.3.9 are susceptible to man-in-the-middle attacks. Moreover, the level 4 nodes contain a large number of orange bars. This illustrates that they also contain a large number of level 3 vulnerabilities. However, since there are seven nodes with level 4 vulnerabilities, a network administrator may decide to address the level 4 vulnerabilities first or the administrator may patch the level 5 nodes first to prevent computers from compromising the entire network. Nevertheless, this visualization allows the administrator to identify the best strategy possible, which comply with the security policies of the organization.

CHAPTER 5

P3D: A PARALLEL 3D COORDINATE VISUALIZATION FOR ADVANCED NETWORK SCANS

Network administrators are making strong efforts to monitor and protect their network and protect against malicious attackers. Network attackers often use scanning and enumeration tools [68] to find more topology information and vulnerable services within a network. Moreover, these attackers attempt to deceive Intrusion Detection Systems (IDSs) and visualization tools by performing stealthy scans such as scanning from multiple hosts on a network, spoofing source and destination hosts, and adding noise (e.g., sending repetitive scans from dozens of spoofed IPs) to trigger false positives and generate misleading information [40]. If these scans are detected before a node is compromised, network administrators could use these scans as precautionary indicators for future attacks, reveal less information about their network, and prevent compromised networks. In this section, I discuss the benefits of using a stereoscopic *3D parallel* visualization technique for network scanning, in particular, when addressing occlusion-based visualization attacks intended to confuse network administrators. A prototype tool was developed from the FRE3DS framework called P3D to assist in detection and increased awareness of distributed coordinate attacks.

5.1 System Design

First, P3D utilizes the FRE3DS framework to capture input from a network traffic sensor. The network traffic sensor could be a network sniffer such as tcpdump or a custom solution using the libpcap library. The data from the NSL is passed to the AL in the form of IP packets and parses the packets into attributes (source IP and port, destination IP and port, TCP/UDP Protocol). Next, the CCL converts each attribute into 3D coordinates. Based on

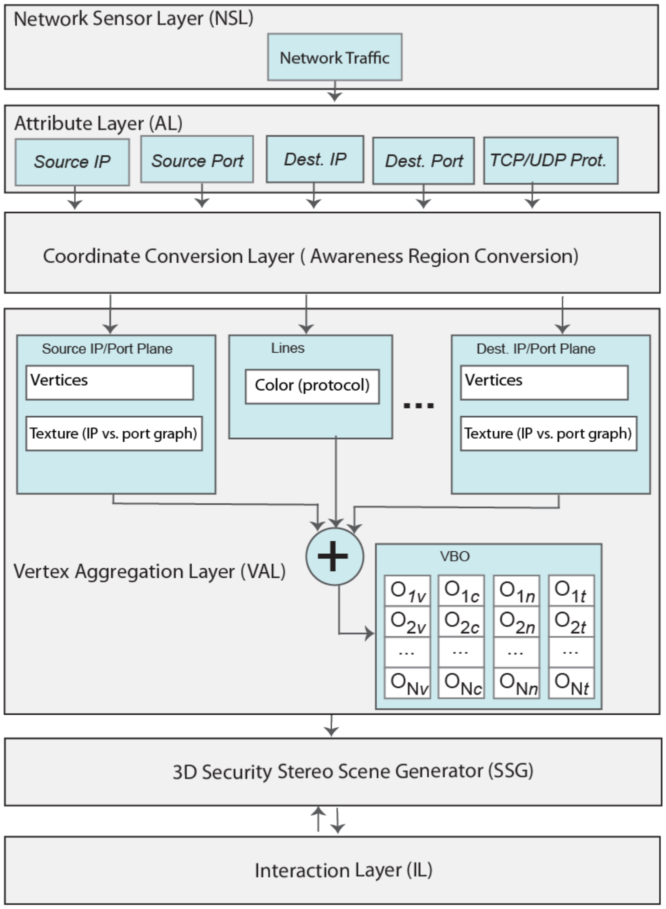


Figure 13: P3D implementation using FRE3DS framework.

the parsed attributes, each coordinate is placed in either the Coordinate Region or Awareness Region. These regions are explained in greater detail later in the chapter. As depicted in Figure 13, the VAL converts the coordinates and attributes to objects such as a line to depict a TCP/UDP connection. After the objects have been produced, the VAL combines vertices and textures and passes them to the SSG. Similar to 3DSVAT, the SSG creates components essential for rendering in an OpenGL environment (i.e., left/right cameras, frustums, clipping planes, focal length) to convert textual packet captures into a 3D visualization with stereoscopic 3D support and interactive techniques such as zooming and panning.

The P3D system consists of 5 components: *Parser*, *Converter*, *Detector*, *Database*,

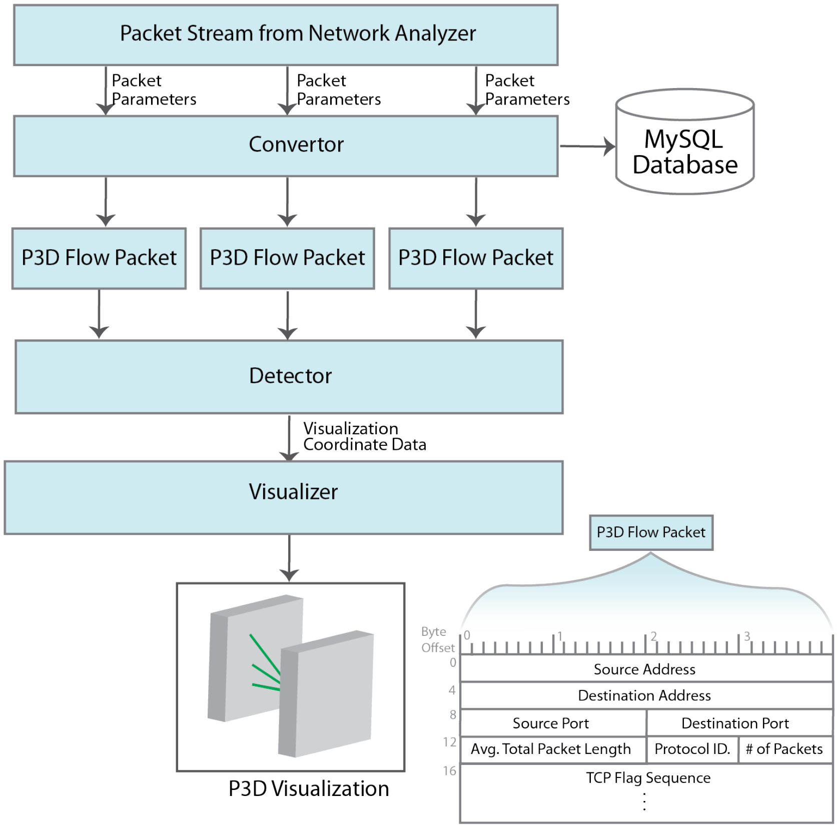


Figure 14: System design of P3D.

and *Visualizer* as illustrated in Figure 22. Network packets are sent to the Parser. Once the Parser recieves the packets it extracts and filters relevant parameters from the packets. Extraction of relevant data from network packets is important to reduce data size and enhance data management for network administrators. The Parser filters and extracts data as follows: IP, Average Packet Size, Source and Destination Port, IP ID, Fragmentation bit, Timestamp, and TCP header flags such as SYN, FIN, URG, and PSH, and ACK. Once the Parser has filtered and extracted all relevant data, it sends this data to the Converter.The Converter converts the parameters from P3D Flow packets into MySQL format and inserts them into a MySQL database.

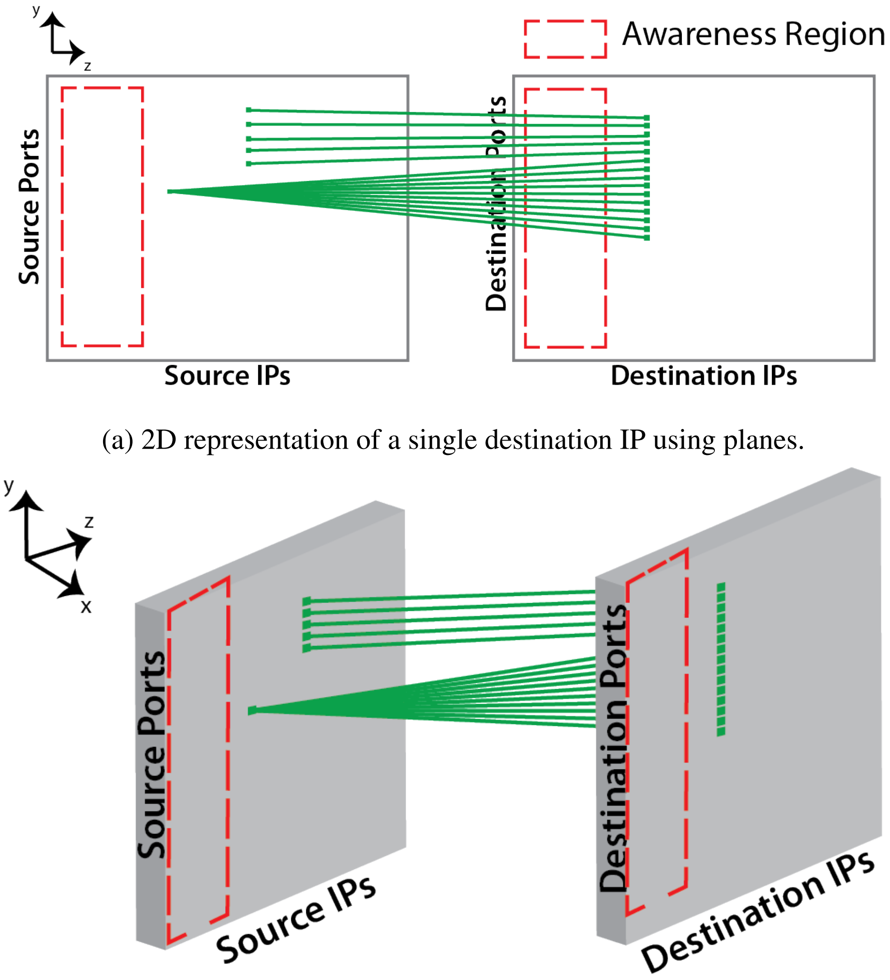
A P3D flow *packet* is a compacted data format that contains information about a flow. A *P3D flow* is defined as a network connection between two nodes or a set of packets with the same source IP, destination IP, source port, and destination port. Within a P3D flow packet, a sequential record of TCP flags is recorded between the source and destination to help determine the type of scan or connection. Next, the P3D flow data is sent to the *Detector*. The Detector examines each flow packet and categorizes the connection as various scans such as FIN, ACK, SYN, and Ping scans. These scans are commonly used to bypass firewalls and subvert IDSs. Next, the Detector performs fixed-time detection and categorizes the scan by examining the flow packets between the two hosts. For example, if there are at least 15 destination ports scanned in 15 seconds, the Detector categorizes the connection as an aggressive port scan. [1]

P3D uses the C++ Object Oriented Model-View-Controller paradigm for higher modularity and extensibility in the *Visualizer*. A custom OpenGL C++ class was developed within a native Windows 7 operating system. Therefore, P3D can easily extend to include multi-touch interactions. Similar to 3DSVAT, the FRE3DS framework [9] was used to produce rapid customized 3D visualizations with stereoscopic support for network administrators to easily and quickly develop various visualizations and efficiently investigate data. As previously discussed, to render the content in stereoscopic 3D, a Nvidia Quadro 2000, Nvidia RF 3D Vision Pro Shutter Glasses, and a 120 Hz Asus 3D monitor for 60Hz screen rendering per eye was used (shown in Figure 8).

5.2 Visualization Design

Currently, no 3D or 2D visualization tool exists that prevents occlusion-based visualization attacks. Using 2D planes in P3D, instead of 1D axes, allows administrators to understand the relationship between source IP and source port. Figure 15 shows a 2D and 3D representation of P3D. Figure 15a shows two adjacent planes to portray the relationship between source port, source IP, destination port, and destination IP.

[1] This rate is used commonly in IDS configurations such as Snort [35].



(a) 2D representation of a single destination IP using planes.

(b) P3D scan to a single destination IP.

Figure 15: P3D visualization design.

As shown in Figure 15b, an aerial perspective of P3D is based on the x, y, and z coordinate systems, which consists of two planes and colored links based on connections (e.g., green denotes TCP connect() procedure call) between the planes. The aerial perspective allows users to select features such as pan, rotate, and translate for faster identification of anomalies on a network than its 2D counterparts. One plane represents a range of source IPs along the z-axis and a range of destination ports along the y-axis, and the other plane represents a range of destination IPs and Ports. The ports range from 0 to 65535 and the IP range depends on the network. The Awareness Region is the stereoscopic portion of the visualization that appear in front of the screen. The colored line denotes the type of TCP

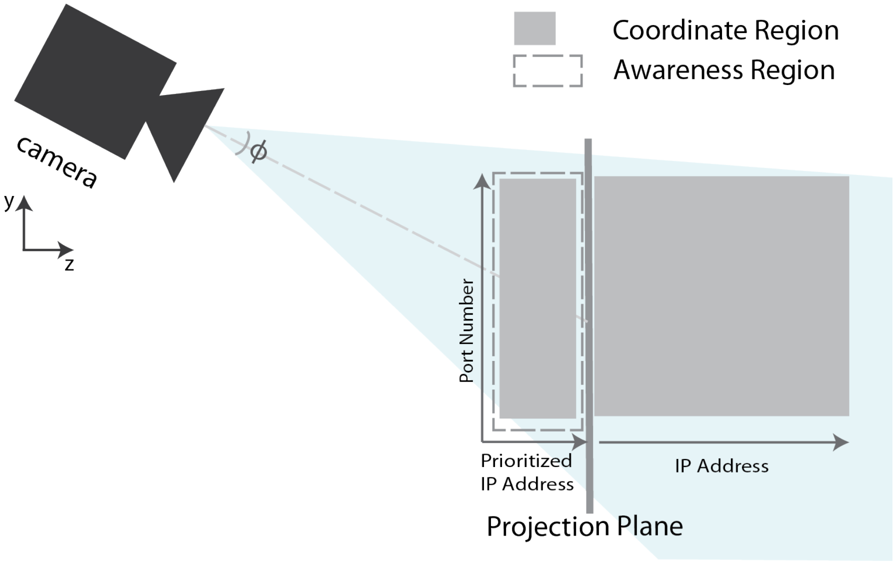


Figure 16: Sideview of P3D.

connection in a flow. For example, the color green may mean the source is attempting to perform a TCP 3-way handshake. Figure 15b clearly shows that one source IP address is scanning from 5 source ports to 5 destination ports on a single destination IP, as portrayed by consecutive horizontal lines. Yet another source IP is scanning from one port to 5 destination ports, as portrayed by a fan pattern. Such a scan goes undetected on most traditional visual IDS systems. In 2D Parallel coordinate systems, the relationship between source IPs, source ports, destination IP, and destination ports is lost. The 3D coordinate system design allows for administrators to uniquely distinguish the scans from different hosts scanning from the same port and detect more advanced techniques such as occlusion attacks.

The 3D Visualizer creates the left and right cameras, off-axis frustum, and other components essential for rendering in a stereoscopic OpenGL environment. The 3D environment renders a stereoscopic visualization into two regions: the Coordinate Region (CR) and the Awareness Region (AR).

5.2.1 Coordinate Region

The Coordinate Region (CR) is used to detect stealthy scans, bogus scans, distributed scan and scans meant to bypass firewalls including SYN, ACK, and FIN scans by coloring the connection link. Using stereoscopic 3D, P3D is especially designed to handle more data than the traditional 2D plots.

5.2.2 Awareness Region

The Awareness Region (AR) is a stereoscopic area that shows a subset of IPs with the highest priority. P3D uses a detection mechanism to determine interesting scans and prioritized IPs by analyzing the TCP/IP attributes in the flow. The detection mechanism visually groups nodes based on various categories: stealth SYN scanning, ACK scanning, and FIN scanning. Another option is grouping the nodes into prioritized IPs. Since AR contains IPs and scanning categorization with the highest priority, stereoscopic technologies are used to enhance awareness of vulnerable nodes. The nodes' awareness is enhanced by positioning the nodes within the focal length of the AR. As a result, with stereoscopic technologies, the nodes within the AR are perceived in front of the physical screen. These nodes allow the administrator to determine which nodes are scanning and being scanned within a network and distinguishes which nodes are potentially compromised.

5.3 Use-Case Analysis

In this section, P3D is evaluated based on use case scenarios for occlusion-based visualization attacks. Next, similarities and differences between Rumint's 2D parallel coordinate visualization technique [31] and P3D are discussed. Rumint's 2D parallel coordinate technique is used for comparison because this technique, like P3D, has no theoretical limit in the number of network parameters that can be visualized. Additionally, the 2D parallel coordinate visualization has, until P3D, led to a quicker understanding and a more informational graph over that of a 2D/3D scattered plot matrix [28].

5.3.1 Source Port Confusion Attack

One occlusion attack is source port confusion. 2D Parallel coordinate visualizations become confusing when multiple source IPs share the same port [40]. This attack is important because it prevents users from understanding how each individual node is behaving on the network in comparison to other nodes.

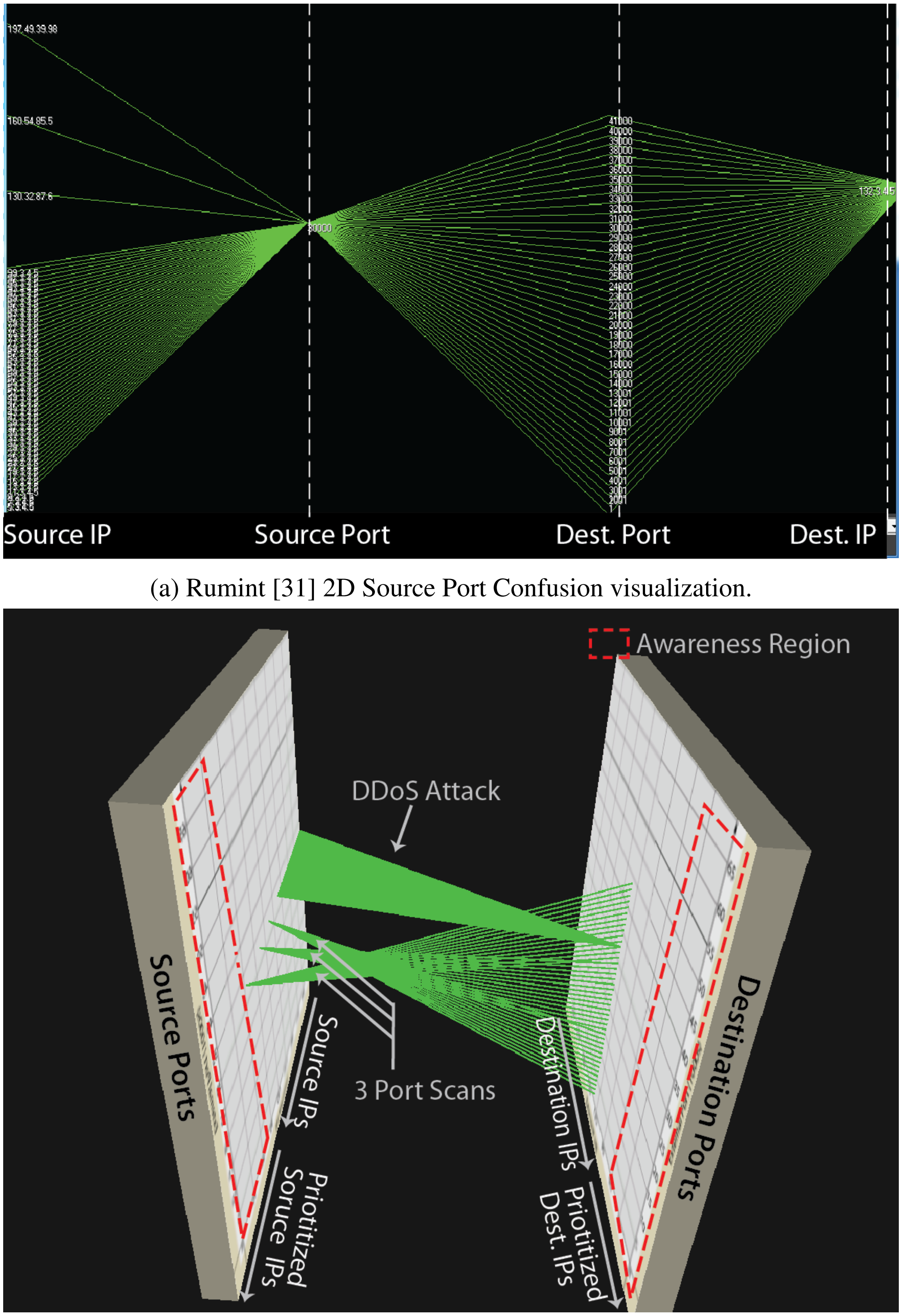
Most network scanner tools contain the ability to forge various packets (e.g., RST packets) from spoofed sources and destination IP addresses as though they are coming from protected hosts behind the firewall. Although visualization IDSs detect most script kiddies scanning activity, more advanced attackers can use a source port confusion attack to run distributed scans from botnets to subvert IDSs and fool most visualization systems by sending a mixture of bogus and real TCP connections.

P3D allows users to distinguish the source and destination relationship between 4-tuple connections (source IP, source port, destination IP, destination port). This distinction allows administrators to quickly determine which host is sending malicious or benign data and prioritize the IP to prevent occlusion. For example, Figure 17 shows a simulated coordinated botnet attack of 100 nodes, ranging from 1.1.0.0 to 200.254.254.254, attacking destination IP address 132.3.4.5 on port 30000.

In the 2D case [31], as shown in Figure 17a, the visualization causes confusion and can be misleading because it is difficult to distinguish whether the scan is coming from one source host, multiple source hosts, or all source hosts. P3D (Figure 17b) clearly shows that 3 IP addresses, 160.54.85.5 and 197.49.39.98, and 130.32.87.6 are performing scans while the other IPs are performing a Distributed Denial of Service (DDoS) attack on port 30000. As a result, P3D can better pinpoint the behavior of individual targeted source IPs in the network than its 2D counterpart.

5.3.2 Windshield Wiper Occlusion Attack

Another example of an occlusion attack is Windshield Wiper attack [40], which attempts to completely obscure a visualization system's output by manipulating packet header fields in a coordinated way. Figure 18 is presented to better portray the detection of the extremely stealthy nodes within the visualization. This attack is created using a packet crafting tool Hping [69] and each packet is generated using the equivalent source and destination ports



(a) Rumint [31] 2D Source Port Confusion visualization.

(b) P3D Source Port Confusion visualization.

Figure 17: Use Case 1: Source Port Confusion.

from 40,000-60,000 on a 10.0.0.0 network. Figure 18b shows a visualization of 10.0.0.0 LAN network using P3D. Within the P3D visualization in Figure 18b, the Windshield

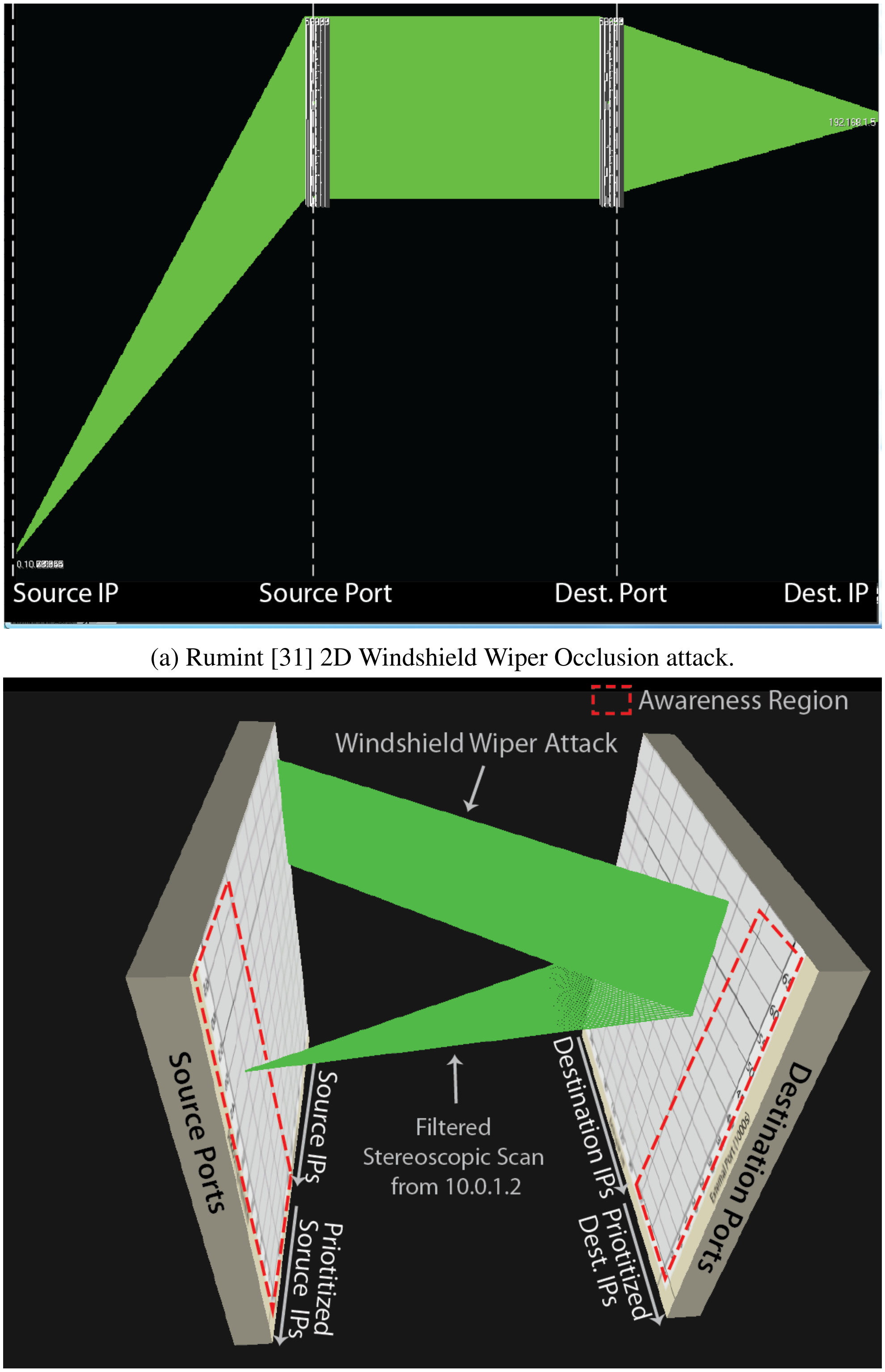
Wiper attack is detected by a resulting diagonal rectangular pattern in the CR.

Unlike in the port source confusion scenario (Figure 18a), the Windshield Wiper attack can obscure a range of ports rather than one port. To perform such an attack, attackers send much more data onto the network than a source port confusion scenario on a network. For this reason, the Windshield Wiper attack is considered more data intensive. As a result, the Detector applies a filtering algorithm to prioritize scans into the stereoscopic AR.The algorithm helps prevent data intensive occlusion attacks such as the Windshield Wiper attack while still maintaining other network activity without removing data. This region uses 3D technologies in the Visualizer to enhance the awareness of the scan by perceiving the scan in front of the computer monitor. As shown in Figure 18b, it clearly shows a scan in the stereoscopic AR. On the other hand, in Figure 18a, the same filtering algorithm is applied to detect the prioritized IP and use brushing to portray the scan in blue. However, in the 2D case [31], *occlusion occurs and scan from IP 10.0.3.34 in the 2D visualization is completely hidden.*

5.4 User Study on Efficacy of Stereoscopy in P3D

In this section, the efficacy of P3D under non-stereoscopic conditions and P3D under stereoscopic conditions are evaluated utilizing user testing methods. Each scenario contains P3D with stereoscopic viewing disabled *P3D* or P3D with stereoscopic viewing enabled *P3D-S*. With user testing, various network attack scenarios are analyzed using P3D and determined if P3D-S validates the hypothesis of reducing user error and increasing response rates. Each task is evaluated using the timespan to complete each task. Also, both the number of user interactions and the sequence of interactions to complete each task are measured. This data collection, in conjunction with ”thinking out loud” technique, provides valuable insights to find the points where a user may have performed an error.

Both quantitative and qualitative data were collected. Quantitative data was recorded from participant interactions with the user interface. To verify that all attempts were



(a) Rumint [31] 2D Windshield Wiper Occlusion attack.

(b) P3D Windshield Wiper Occlusion attack.

Figure 18: Use Case 2: Windshield Wiper Occlusion Attack.

counted correctly in the post-hoc analysis, for each detection attempt, we plotted interactions. In total, we individually validated 135 attacks from 15 different subjects. Questionnaire responses were transcribed to a spreadsheet. Qualitative data was gathered from observations during the study, open questions in the questionnaire, and a post-survey using 5-point Likert scales following the study. This data was transcribed and important themes identified.

5.4.1 Testing Procedure

For our lab-based evaluation, 15 different subjects were recruited (13 male and 2 female), aged between 22 and 32 years (mean = 25, sd = 4.7) to take part in our experiments. To explore 3D stereoscopic conditions, the subjects were asked to detect and identify 9 network scenarios. When each participant arrived at the lab, the tasks were explained to the participants and each participant was asked to sign a consent form. Then, participants were instructed on how to use P3D, in particular how to detect common attacks such as port scans and DoS attacks. A user interface (UI) was developed to automatically guide each participant through a sequence of screens, each prompting them to enter an explanation of each attack. Each sequence consisted of the following 9 attacks:

- **Port scan** is a scan of various services from a single host.
- **DoS** occurs when one IP attempts to conned to a range of ports on a destination IP by sending spoofed packets on a network from multiple source ports.
- **Port source confusion attack** occurs when multiple source IPs share a common source port [40].
- **Port Confusion Attack with DDoS and Scan** is defined as a set of source IPs that sends packets to a a single port on a destination IP. In parallel, 3 source IPs attempts to scan the network under the rate is used commonly in IDS configurations such as Snort [35].

- **Distributed Denial of Service (DDoS) using SYN flood** is used to block services by sending special crafted packets with the SYN flag enabled by multiple hosts.
- **2 port scans with large noise** occurs when a large amount of background noise is injected into the network while 2 port scans are occurring.
- **DDoS with background noise** performs a DDoS attack with a large amount of legitimate traffic from remote hosts.
- **FTP disguised attack** is an attack disguised as a concurrent FTP transfer.
- **Legitimate traffic with no attacks** represents benign legitimate traffic on a network.

Furthermore, each participant was asked to verbally mention his or her actions. We recorded response time, recognition error, and interaction error rate during each session. The response time is defined as the time span for recognizing a visualization goal (i.e., successfully detect an attack). The recognition error refers to either a false negative (negligence) or a false positive (mis-identification) occurrence of network activity. The interaction error rate is defined as the ratio between the user's number of interactions that does not contribute to identification[2] of an attack and total number of interactions that is used to determine an attack.

5.4.2 Response Time Analysis

Regarding response time, no statistical differences existed among P3D and P3D-S under non-noisy conditions. A non-noisy condition refers to network dataset that contains traffic with a low variation of source and destination IPs. However, under conditions where large amounts of legitimate traffic across a wide variety of IPs (e.g., noisy conditions), P3D-S allowed for faster user response than P3D. This phenomena occurs in the visualization because noisy datasets cause visualization occlusions that require the user to perform more

[2] Identifying the interactions that do not contribute to the identification of an attack is somewhat subjective. In this work, the interactions perceived to be conflicting with goals stated by the user while performing the ”thinking out loud” method are considered as not contributing to the identification of the attack.

interactions to understand the visualization. As a result, these visualization occlusions produce a dense visual representation. Table 5 lists the mean response times in seconds, respectively, for each of the nine scenarios. For P3D-S, the results show a 11.38 % decrease in response time for 2 port scans with noise and a 10.88 % decrease in response time for DoS with noise.

Table 2: Mean user response time (s), broken down by scenario.

Scenario	***P3D-S* (s)**	***P3D (s)***	***Percentage (%)***
Port scan	49.01	41.58	15.15
DoS	46.49	43.10	7.29
Port Confusion with DDoS and Scan	115.18	105.43	8.46
Port source confusion attack	125.67	115.46	8.12
DDoS using SYN Flood	126.95	107.39	15.41
DoS with background noise	182.86	162.96	10.88
2 Port Scans with large noise	180.69	160.136	11.38
FTP disguised attack	191.74	170.99	10.82
Legitimate traffic with no attacks	230.34	196.05	14.89
AVG	138.77	122.57	11.38

5.4.3 Recognition Error Analysis

We analyze recognition error using P3D. The results show no statistical differences existed either among P3D and P3D-S for recognition error. The thinking out loud technique (i.e., verbally explaining actions) is used to assist in understanding error in recognizing an attack. From this technique, 1.48 % of scenarios of legitimate traffic were misdiagnosed (false positive) as an attack. Another 1.48 % of attacks were not detected (false negative). This result notably shows that although no difference exists under stereoscopic conditions, the P3D tool contains low detection recognition error rate under both stereoscopic and non-stereoscopic conditions.

5.4.4 Interaction Error Analysis

Similar to the response time, the results for the interaction error show that if the network contains more noisy traffic, the interaction error rate increases to 22.70 % as shown in Table 5. As discussed earlier, the interaction error rate is defined as the ratio of a user's number of interactions that does not contribute to identification of an attack and the number of interactions that is used to to determine an attack. To calculate the interaction error, each interaction is recorded using programmable hooks and an optimal path is manually constructed for the recorded interaction set. The analysis shows that 1123 interaction errors occurred out of 7232 interactions, which approximates to 18.29 percent. The majority of the error stemmed from the exploration of the visualization and investigation other IPs and port numbers rather than the IPs and ports of interest.

Table 3: Interaction error rate, broken down by scenario.

Scenario	*P3D (%)*	*P3D-S (%)*	*Percentage Increase (%)*
Port Scan	13.34	12.83	3.82
DoS	12.32	12.40	-0.65
Port Confusion with DDoS and Scan	17.50	16.70	4.57
Port source confusion attack	18.00	16.00	11.11
DDoS using SYN Flood	19.30	16.20	16.06
DoS with background noise	23.20	20.30	12.50
2 Port Scans with large noise	25.50	22.70	10.98
FTP disguised attack	30.50	28.20	7.54
AVG	19.96	18.17	8.24

5.4.5 Subjective Feedback

Participants filled out a post-experiment questionnaire rating their experience with stereoscopic 3D on a 5 point Likert scale (1 being most negative and 5 being most positive). They were asked to comment on effectiveness of stereoscopic 3D visualizations. Furthermore, the qualitative feedback from the study revealed that some participants developed strategies for detecting attacks. One participant constructed stories and rhymes around the visualization to make them more memorable (i.e., upright square is scanning a port, not fair).

Participants from the main study found stereoscopic to be easier to detect attacks, due to the perceived depth. A couple of participants reported some eye fatigue. This occurrence is due to the eye's natural ability to focus on the monitor instead of objects perceived to be in front of the screen. A proper analysis of memorability requires a longterm study and was therefore beyond the scope of this work. However, when we asked two participants to

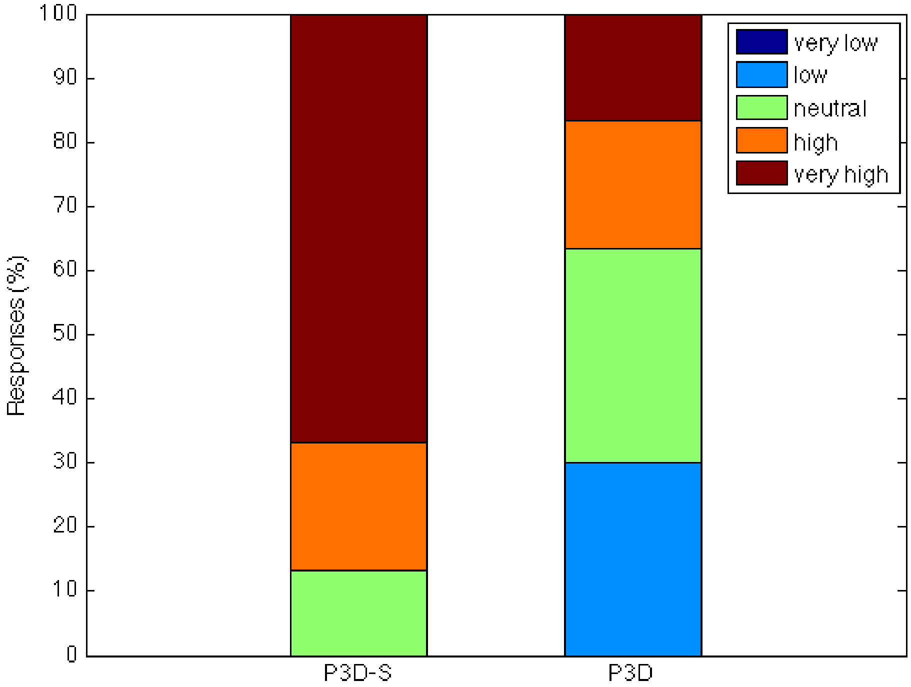


Figure 19: Distribution of responses to question: ”How do you rate the ease-of-use of the system?”.

detect attacks two days later and we observed that the participants were still able to detect the attacks at a rapid rate.

CHAPTER 6

NAVSEC : A RECOMMENDER SYSTEM FOR 3D NETWORK SECURITY VISUALIZATIONS

3D network security applications [45, 46] use visualization techniques to extend the visualization space and show more network activity from more than a single vantage point. Since displays are physically constrained to 2D devices (e.g., computer monitor), these 3D environments use visual cues such as shadowing and size to adequately represent depth. However, most 3D visualization techniques for the network security field encompass a long range of network data attributes such as IP address, port number, and TCP/IP protocol that can not be adequately viewed from a single vantage point. Thus, some visualization designers employ a *top-down* approach [26] for netowrk administrators to explore 3D visualization environments. A top-down approach occurs when network administrators start at an initial visualization that reveals a single overview of the network and manipulate the visualization environment to discover more specific details to potentially detect network attacks. Other researchers use a *bottom-up* approach [70], which starts from an initial node on a network and expands to show the relationship between other nodes. This approach is useful where the dataset is too large for a top-down approach. In both top-down and bottom-up approaches, obtaining different views of the network requires a user to perform interactions that manipulate and navigate within these visualization environments. However, in many network security visualization tools, there could be hundreds of possible interaction combinations for a user to navigate [71]. For a novice user possessing little or no expertise, executing a complex task using a visualization tool, such as identifying a network attack, becomes difficult. Thus, a novice user may need assistance in performing the correct sequence of interactions. To solve this problem, a system was developed that could be used as a module within the FRE3DS framework for 3D visualization tools to recommend both visualizations and interactions for a network-based attack using a recommendation algorithm. This system recognizes similar patterns found in current and historical interaction data from a group of expert users and recommends a set of interactions to a novice user. NAVSEC uses the nearest-neighbor machine learning algorithm to identify unknown attacks. Current user's and a group of expert users' past behavior are studied to assist users in determining a visualization to use for identifying an attack and to aid a novice user in navigating and manipulating 3D visualizations for network security applications. This assistance in navigation leads to the reduction of interactions used to detect various network activity and guides a novice user to certain objectives such as detecting a stealthy network-based attack.

As discussed in chapter 3, NAVSEC takes advantage of FRE3DS' IL (portrayed in Figure 20). Within the IL, coordinates (C_{1i} ... C_{1N}) are generated from n different inputs (i). The input could be multiple points of contact from a user's fingers on a multi-touch device. The input is stored in a buffer and the InterSec Module pulls data from the input using First-In-First-Out (FIFO) queuing method. Next, based on the type of input, InterSec translates the coordinates into interactions or gestures and executes the interactions or gestures within SSG. In parallel, the NAVSEC receives the interaction from the InterSec Module and applies the recommender system to the FRE3DS framework. Finally, the NAVSEC module sends a recommended interaction or set of interactions to the SSG.

6.0.6 An Introduction to Nearest-neighbor Approach for NAVSEC

Navigating within a 3D visualization for network security applications can be difficult because a large amount of network data is portrayed in a screen-size visualization. Moreover, visualizations offer a limited amount of navigational information [59]. There are hundreds of details to track and one mistake could result in a misinterpretation of an attack that is being visualized. In addition, different interactions could be utilized to navigate through a 3D environment to extend the visualization space. For instance, possible interactions include zooming in and out of the visualization, rotating about a pivot point, panning, or shifting

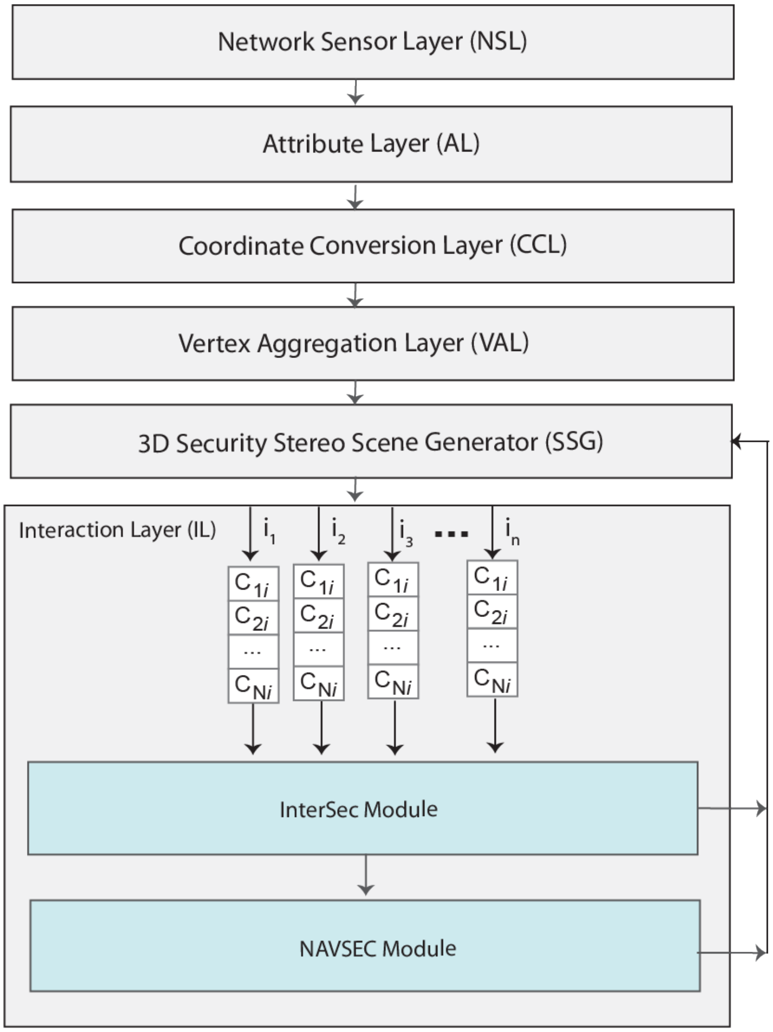

Network Sensor Layer (NSL)
Attribute Layer (AL)
Coordinate Conversion Layer (CCL)
Vertex Aggregation Layer (VAL)
3D Security Stereo Scene Generator (SSG)
Interaction Layer (IL)
i_1
i_2
i_3
...
i_n
C_{1i}
C_{2i}
...
C_{Ni}
InterSec Module
NAVSEC Module

Figure 20: NAVSEC and InterSec modules within the FRE3DS framework.

the camera view of the visualization environment. Another interaction used to extend the 3D environment is the addition of a 2-axes plane that plots other network attributes such as average packet size vs. total number of packets. Within FRE3DS, NAVSEC, collects sequences of interactions from a group of expert users and stores them into a database. The active user's current and past interactions are also collected. The nearest-neighbor approach, using cosine similarity, generates both suggested visualizations and next interaction for a novice user based on the collected interactions. Through the usage of NAVSEC, users (specifically novice users who are exposed to unfamiliar visualizations) are recommended possible interactions to extend 3D visualizations and become aware of interaction options to detect attacks more effectively in their future use.

As mentioned previously and shown in Figure 21, NAVSEC applies the nearest-neighbor approach [59] to an active (current) user's recent set of interactions at instance *t* (state of the active user) for an attack (e.g., DoS or an advanced stealthy port scan meant to bypass a firewall or subvert an IDS) as denoted by the square. Instance *t* is calculated using the active user's current set of interactions. At instance *t*=0, an initial visualization is presented to an active user. This visualization could be a representation of real-time network data or an offline network dataset from a packet capture file. From this initial visualization, the user can speculate an attack, but the system needs more information to help the user investigate the attack further. By choosing an attack type portrayed as a button located in the tools palette on the visualization interface, the user provides NAVSEC with a starting point. As the active user interacts with the visualization tool, the active user's set of interactions at instance *t* is compared to sets of interactions from expert users. In my work, I assume expert users are knowledgable and successful in producing visualizations from a set of interactions to identify an attack. Each set of interactions is a sequence of interactions for the same type of attack speculated by the active user as denoted by triangles in Figure 21. The system picks the nearest-neighbor and recommends this set of interactions as denoted by the arrows until a visualization goal is reached. The recommendation system uses the number of occurrences of each interaction type between a novice user and group of expert users and computes the similarity to a set of novice user's interactions using a cosine similarity function. The visualization goal in Figure 21 is a visualization used by expert users to identify or detect a speculated attack.

Furthermore, if an active user does not perform any interactions (e.g., instance *t*=0), but wishes to be recommended a set of interactions, the most commonly-used sequence of interactions produced by the group of expert users for an attack are used.

Depending on the active user's behavior, NAVSEC dynamically generates the most

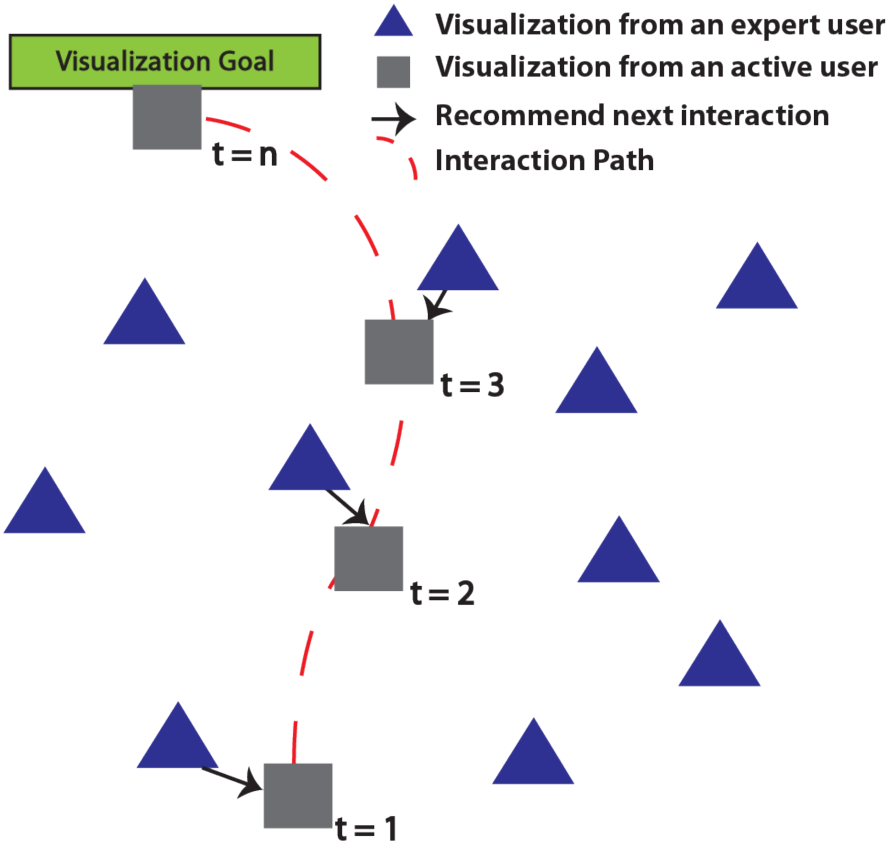

Visualization from an expert user
Visualization from an active user
Recommend next interaction
Interaction Path
Visualization Goal
t = n
t = 3
t = 2
t = 1

Figure 21: Nearest-neighbor approach for a network attack.

similar set of interactions or visualization from a group of expert users. My intention for this system is to recommend a set of interactions used to extend or manipulate the visualization space for an active user to be guided along a path to reach a visualization goal at instance *n*. NAVSEC only recommends how a novice user can reach a speculated attack based on expert users' interactions and does not promise to find any attack.

6.0.7 3D Network Security and the Application of NAVSEC

Existing 3D visualizations have been created to visualize data using techniques such as iconic tree structures, bar charts [45, 46], and 3D scatter plots [5]. For example, PortVis [42] is a visualization tool that aids in detecting large-scale network security events and port activity. Also, NetBytes Viewer [43] visualizes the historical network flow data per port of an individual host machine or subnet on a network using a 3D impulse graph plot.

These tools only consider the 4-tuple: source IP, destination IP, source port, and destination port. Thus, these tools show a small amount of detail and only display the counts of activities rather than the activities themselves. This lack of detail could lead to a misinterpretation of network attacks. Also, NetBytes Viewer is static; hence, new visualizations cannot be derived from NetBytes Viewer to detect interesting attacks. However, with integration of NAVSEC and NetBytes Viewer, we can provide a foundation to recommend new visualizations and help novice users of NetBytes to detect more types of attacks.

Furthermore, research has been performed in detecting unknown large-scale Internet attacks including internet worms, DDoS attacks and network scanning activities using a parallel coordinate system [28]. Parallel coordinate attack visualization (PCAV) [28] uses hash algorithms to detect nine graphical signatures, and a parallel 3D coordinate system for network security (P3D) [6] extends the visualization space by introducing a stereoscopic awareness region mechanism using 3D glasses to highlight important data and expand the visualization to help prevent occlusions. NAVSEC can be uniquely integrated as a module to reduce the amount of interactions used in these visualizations without any guidance. Moreover, NAVSEC focuses on aiding novice users by examining the interaction space rather than the visualization space. NAVSEC addresses these limitations by visualizing and incorporating more data, allowing it to help uniquely characterize port scans and further understand scanning activity. In addition, NAVSEC uses a knowledge base of expert users so that as different attacks occur, the most similar visualization could be dynamically generated. As more expert users utilize the system, thereby increasing the expert user interaction database, the most similar visualization to the active user's behavior may evolve over time.

The only ”intelligent” visualization system within network security applications that has been proposed is NIMBLE [72]. In [72] , Rasmussen introduces NIMBLE, an incident management system through visualization. NIMBLE calculates the similarity for given

IDS alerts and historical alerts classified by network administrators. His results show improved network analyst's accuracy in defending the network with the tool's visual display and the given recommendations. His primary visualization focuses on correlated IDS output rather than network traffic. In contrast, NAVSEC is more comprehensive as it also accounts for understanding of attacks meant to mislead the user or subvert an IDS. NAVSEC can also ”walk” users through a network attack analysis.

6.1 System Design

As mentioned in the Introduction, the goal of this research is to present a novice user with recommendations of a set of user interface (UI) interactions that produce visualizations that aid in effectively navigating within 3D visualizations. This set of interactions is shown as a textual list or as a visualization produced by automatically performing the listed interactions. NAVSEC provides the textual interactions as an option for the user to iterate tasks while referencing the textual list for more detail on the expert's reasoning in creating a visualization. This feature could also be conducive in education/training situations for allowing instructors to show practical examples of various expert users' identification of attacks. However, showing a visualization rather than textual interactions allows the active user to view a quick and accessible visualization from an expert user by simply clicking a button. This feature reduces the amount of the interactions needed from an active user.

The recommender system was designed as a modular web-based system, which can incorporate various machine learning techniques (e.g., nearest neighbors, support vector machines, neural networks) as modules. A developer can extend the recommender system by developing a custom module to meet their specific requirements. As a result, this modular approach increases the flexibility of the system. Furthermore, the system allows for multiple expert and active users to connect to a centralized web-server. This system could potentially be expanded to accommodate load balancing and redundancy, common techniques used for reliability and scalability. Moreover, NAVSEC can support different expert

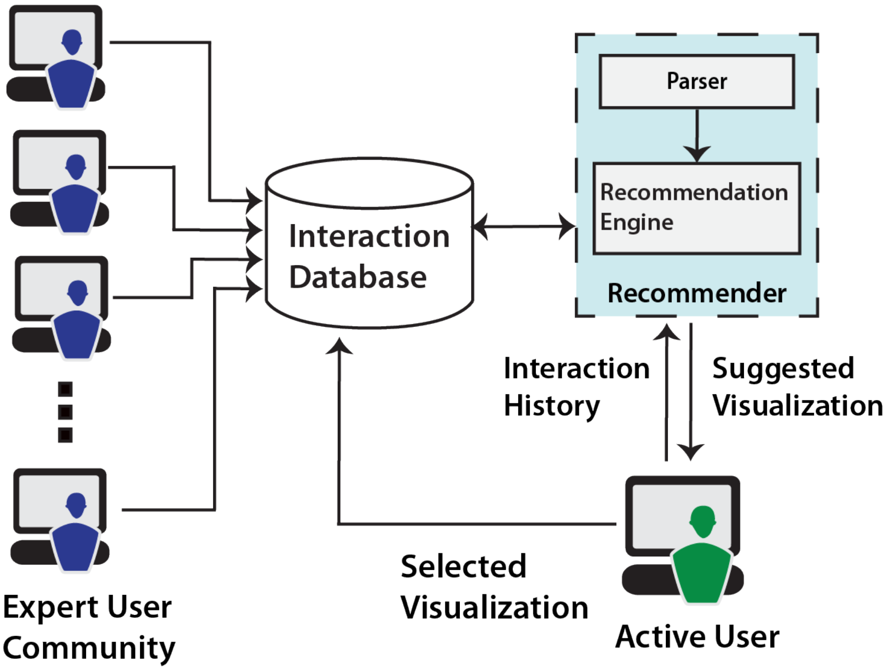


Figure 22: System design of NAVSEC.

user communities such as colleagues in a small security organization where resources are limited or training inexperienced military personnel. Additionally, NAVSEC user interface is seemingly unobtrusive; it does not force the user to respond to recommendations before continuing their work. To implement this system, a distributed system was deployed for both expert users and novice users. A light-weight visualization tool is deployed to each user with the NAVSEC module installed and the module connects to a centralized server for computation. As illustrated in Figure 22, the NAVSEC system consists of 4 components: *Active User*, *Expert User Community*, *Interaction Database*, and *Recommender*. Each of these components are expounded upon in the following sub-sections.

6.1.1 Active User

The active user (i.e., current user) is defined as the one navigating the visualization tool with the NAVSEC module installed. In my work, the active user is a novice that may be confused regarding the next set of interactions (and resulting visualization) to be performed to successfully identify an attack.

6.1.2 Expert User Community

The Expert User Community is a set of users with significant experience in the network security and visualization fields. An expert user could be colleagues in a security organization or IT department. These users interact with a visualization tool (e.g., P3D [6]), to create visualizations used to detect or identify an attack on a network. Expert users categorize the visualizations by clicking on an attack category button located in the tools palette. In this work, an expert is assumed as someone who has sufficient knowledge to perform the correct sequence of interactions to detect an attack. The sequence of interactions (i.e., zoom, pan, translate, add plane) for each expert user and categorized attack is recorded and sent to the centralized Interaction Database prior to any active user activity.

6.1.3 Interaction Database

The Interaction Database is a collection of interaction sequences. Also, the Interaction Database stores the amount of interactions input by both expert and active users for each attack. The stored amount of interactions are used by the Recommender component to compute the popular set of visualizations and interactions for specific network attacks. The Recommender also receives interactions as input from an active user via HTTP requests and stores them into the Interaction Database. The data from the Interaction Database contains the interaction identification number which is a number that maps to the type of interaction performed by the active user. This identification number is used to convert the sequence of interactions into readable text. The Recommender performs a *cosine similarity analysis* on the active user data and suggests a set of interactions to find a particular type of attack. The next sub-section further discusses how cosine similarity analysis is used to compare an active user's set of interactions to a group of expert users' interactions and then finds the most similar set of interactions from the group of expert users. This set of interactions can be used to generate visualizations for the active user. After the active user performs the suggested sequence of interactions, the sequence is sent and stored in the Interaction Database for future use.

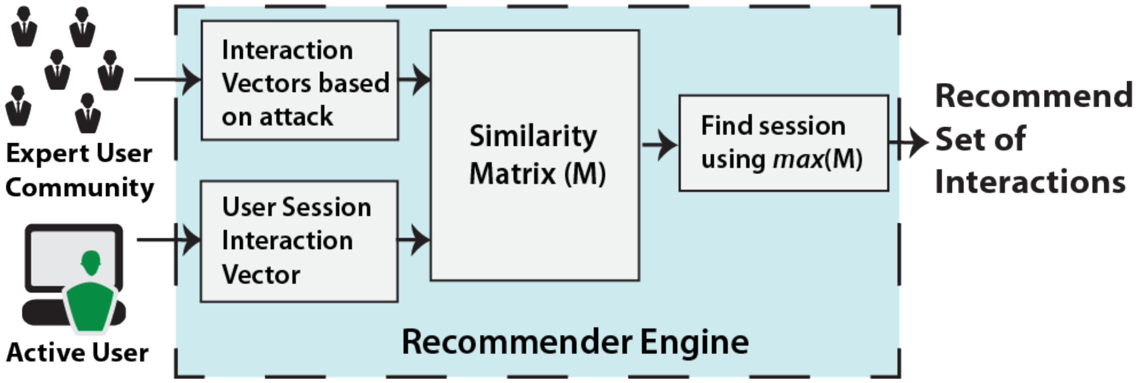

Expert User Community
Active User
Interaction Vectors based on attack
User Session Interaction Vector
Similarity Matrix (M)
Find session using *max*(M)
Recommender Engine
Recommend Set of Interactions

Figure 23: Design of recommendation engine.

6.1.4 Recommender

The Recommender parses the data contained in the Interaction Database and computes a set of interactions for recommendation to an active user in real-time as listed in Algorithm 1. This computation is performed by the Recommendation Engine (Figure 23).

The Recommendation Engine uses two inputs: the interaction history of expert users and the current history of interactions of an active user. Once the two inputs are generated, the Algorithm 1 *ComputeSetOfRecommendations( )* is used to calculate a recommended set of interactions from the two inputs. The algorithm for recommending a set of interactions is divided into three parts: formulate an interaction vector, compute a similarity matrix, and recommend a set of interactions.

The recommender system formulates an interaction vector (lines 2-8) by using the total number of occurrences n_j of the interaction type j for the kth attack session s_k. The interaction type is the categorized interaction performed by the user to manipulate a visualization such as *add left plane, zoom, rotate, add right plane*. An attack session is a sequence of interactions to visualize an attack. Furthermore, a user can perform multiple sessions for a type of specific attack a. For example, an expert user could visualize 10 port scan attacks (K=10) where k is the session identifier and K is the total number of sessions for an attack. As given in Equation 1, each user's number of occurrences n_j is tabulated into an interaction vector $\vec{v_k}$ for the kth session where k is the unique identifier for each session. In Equation 1, n_j is the summation of each unique occurrence i of interaction type j for attack session s_k. Each interaction type j corresponds to the position of interaction vector $\vec{v_k}$ where J is the total number of interaction types. Thus, the total number of dimensions within an interaction vector $\vec{v_k}$ is equivalent to the total number of interaction types J. Each interaction vector $\vec{v_k}$ for all expert users is stored in the Interaction Database.

$$\begin{aligned} \vec{v_k} &= [n_0\; n_1\; \cdots\; n_{J-1}] \\ where\; n_j \;&=\; \sum_{i=0}^{I-1} i_j\; for\; attack\; session\; s_k. \end{aligned} \tag{1}$$

Next, the cosine similarity function was used to produce a similarity matrix (Equation 2). This similarity matrix is derived from comparing the active user's interaction vector $\vec{v_h}$ to each expert user's interaction vector $\vec{v_k}$ for all sessions.

NAVSEC's algorithm loops through (lines 9-13) a set of interaction vectors for each session within the attack. Each interaction vector $\vec{v_k}$ of an expert user performs a similar distance function such as cosine similarity to an active user current session $\vec{v_h}$. The resulting value M_k is stored into a user similarity matrix $\vec{M}$. The resulting similarity values stored in $\vec{M}$ ranges from 0-1. The value 0 means that vectors are orthogonal to each other and the set of interactions is not related. The value 1 means the vectors are collinear or the set of interactions is similar. Thus, values close to 1 indicate $\vec{v_h}$ is similar to $\vec{v_k}$.

$$M_k = cos(\vec{v_k}, \vec{v_h}) = \frac{\vec{v_i} \cdot \vec{v_h}}{||\vec{v_i}|| * ||\vec{v_h}||} \tag{2}$$

Finally, a set of interactions (lines 14-20) were recommended by computing the closest score to 1 after taking the maximum value of similarity matrix $\vec{M}$. Each position in the similarity matrix $\vec{M}$ is mapped to a session identification number k and sequence of actions can be determined by selecting the sequence of interactions from the session identification number. Then, the sequence of interactions from the expert user session can be sent back

Algorithm 1 ComputeSetOfRecommendations( )

```
begin
% Formulate an interaction vector $v_k$.
  $\vec{v_k} = [n_0\ n_1 \cdots\ n_{J-1}]$
  for ($i = 0$ $to\ total\_Number\_Interactions_{s_k}$)
    if ($typeOfInteration(i) == j$)
        return $n_j + +$
      end if
  end for
% Compute a similarity matrix $M$.
  $\vec{M} = [M_0\ M_1\ \cdots\ M_{K-1}]$
  for ($k = 0$ $to$ $K - 1$)
    $M_k \leftarrow cos(\vec{v_k}, \vec{v_h})$
      return $M_k$
% Recommend a set of interactions.
  for ($k = 0$ $to$ $K - 1$)
    if $M_k > max(\vec{M})$
      return $session_k$
      end if
  end for
  return $Interaction\_sequence_{s_k}$
end
```

to the active user as the recommended set of interactions.

6.2 System Implementation

NAVSEC was implemented as a module for a Parallel 3D coordinate system (P3D) [6] and illustrate the functionality of NAVSEC with a use-case scenario for advanced stealthy port scans. P3D is used because unlike most 3D counterparts this tool has no theoretical limit in the number of network parameters that can be visualized. Therefore it is better able to detect visualization attacks [40] unlike a 2D/3D scattered plot matrix [28], which uses stereoscopic 3D support and interactive techniques such as zooming and panning. P3D uses 2D YZ planes of network attributes positioned along the x-axis and shows the relationship of the attributes using colored lines. Each colored line represents the type of connection (e.g., TCP, UDP). To incorporate NAVSEC into P3D, I extended P3D by modifying the interaction layer of FRE3DS [9] for both expert and active users. As discussed, NAVSEC produces a recommended set of interactions or visualizations. This set of recommended interactions are displayed in a web-based interface that the user can refer to when convenient. NAVSEC can also be displayed as a peripheral tool palette located within the P3D user interface. This tools palette contains a list of network attacks in the form of buttons. If a user clicks on an attack, the suggested visualization is presented to the user.

NAVSEC contains a client-side C++ component that is integrated as a module to send GET HTTP request of interactions to the server-side application. The server-side uses an Application Programming Interface (API) to receive HTTP requests. The NAVSEC API is developed using Codeigniter, a PHP framework used for rapid web development [73]. Codeigniter uses a Model-View-Controller architecture design to assist in code reusability so that the NAVSEC module code can be easily integrated with other visualization tools.

6.3 Use Case Analysis

In this section, NAVSEC is evaluated with a use-case scenario for stealthy port scanning attacks. Stealthy port scanning attacks were used because these attacks are commonly used by attackers to bypass firewalls, subvert IDSs, and could often be misinterpreted by network administrators. Next, discussion is presented about a novice user's confusion between visualizing a stealthy port scan disguised as a FTP scan and a file transfer using multiple concurrent TCP connections. A concurrent FTP transfer occurs when a client and server creates multiple TCP connections to increase total throughput of the file transfers. This network activity is commonly implemented by clicking the ”Enable Concurrent/Multiple Connections for transfers” in FTP clients (e.g, Filezilla [74]). Through the FRE3DS framework, NAVSEC demonstrates that it can help avoid this confusion. The number of interactions performed when NAVSEC is enabled vs. disabled is also evaluated. Finally, the convergence of interactions to a high similarity score is shown, which denotes that an active user set of interactions approaches an expert user's set of interactions.

6.3.1 Stealthy Port Scanning Use-Case

Stealthy port scanning is a network attack used to discover exploitable communication channels by probing for vulnerable services in a form that goes undetected by traditional intrusion detection systems. Most network scanner tools contain the ability to forge various packets (e.g., RST packets) from a spoofed source and destination IP addresses as though they were coming from protected hosts behind the firewall. Also, a highly skilled attacker can perform scans that emulate legitimate network traffic. Although current visualization IDSs detect most scanning activity, more advanced attackers can perform stealthy port scanning attacks to subvert IDSs. Therefore, network administrators must distinguish stealthy attacks from legitimate network traffic.

As shown in Figure 24, a concurrent FTP file transfer of ten 20 Megabyte files using Filezilla [74] on the Windows Operation System (OS) is simulated. The source IP 57.25.6.30 is attempting a TCP connection from 10 ephemeral ports (50332-50341) to 10 ephemeral ports (53829-53837, 53850) on a destination IP 57.25.6.100. The source IP is also attempting to connect from 18 ephemeral source ports to port 21 (default port for FTP service) on 57.25.6.100. These connections occur because two connections are used to initiate a FTP connection and send FTP commands for each data transfer. By examining this initial visualization, both a novice and expert user could suspect that an attacker is performing a stealthy scan from 10 multiple consecutive source ports to 10 destination ports rapidly, but below most IDSs scanning rate thresholds.

However, since an expert is familiar with P3D and network security, the expert can examine the network data further. The expert extends the visualization by performing interactions with the P3D interface such as color coding the lines to specific network protocol to keep the consistency of the visualization and adding a left plane with the z-axis as total size of packets in a TCP flow and the y-axis as number of packets in a TCP flow. By examining this TCP flow, the expert can determine the number of packets and size of packets in a TCP connection to evaluate if the connection is actually a port scan. Figure 25 shows an

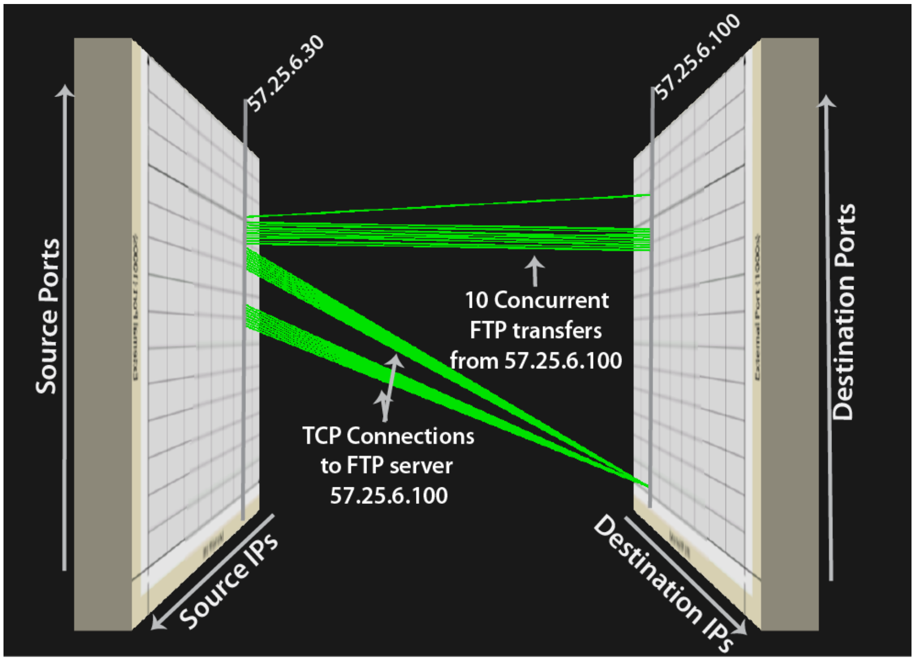


Figure 24: A potential stealthy port scan.

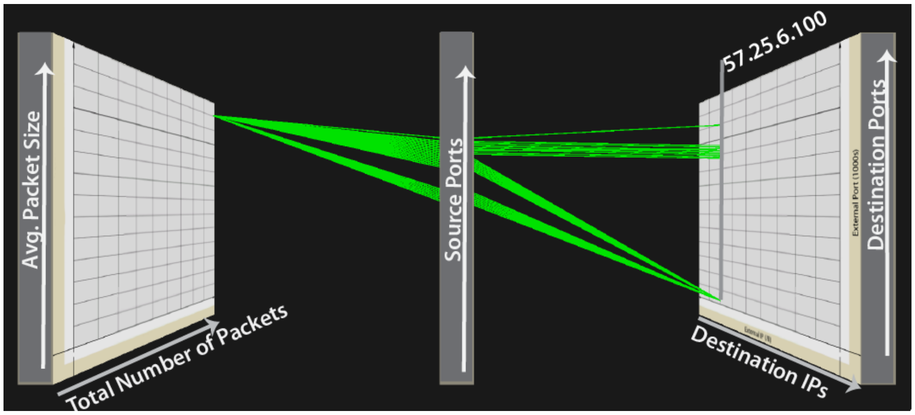


Figure 25: P3D [6] Multiple concurrent FTP file transfer.

extension of P3D (i.e., an expert added an additional plane) where the average size of the packets and the number of packets sent by the source ephemeral ports are high (1438 bytes and 147304 packets, respectively). Since a large amount of data is sent, the source node can be interpreted as a potential FTP connection in passive mode with multiple concurrent connections to increase FTP server's throughput rather than a port scan.

On the contrary, if the total number of packets in a TCP connection and an average

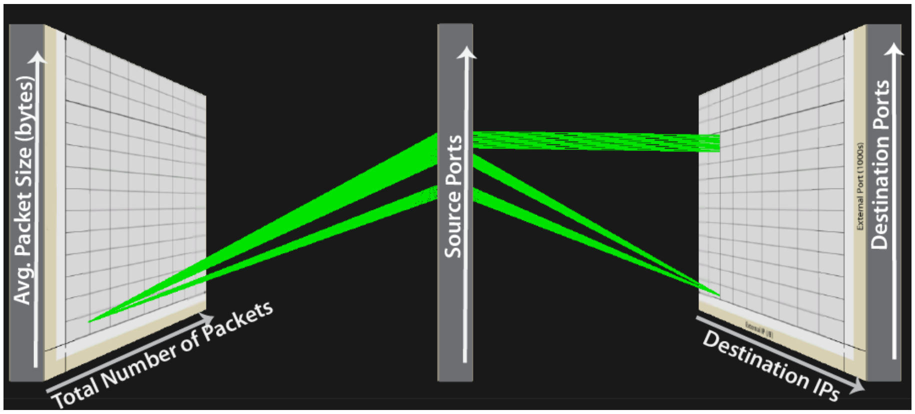


Figure 26: P3D [6] steathy scan.

packet size from the source are low, (one packet) as illustrated in Figure 26, the visualization is potentially a stealthy port scan. An attacker can send packets with data in the payloads to further confuse the administrator. However, an expert user could further evaluate the scenario by adding a plane to show the number of ACK packets versus RST packets. If no ACK packets are being sent from the server, then a legitimate connection has not been established.

An expert user can successfully distinguish attacks from legitimate traffic by expanding and manipulating the P3D environment. However, the ability to successfully and efficiently expand this visualization requires the user to possess an advanced background in network security and experience with the P3D visualization system. From the initial analysis in Figure 24, the novice may view multiple connections to multiple destination ports and mistakenly assume the network activity is a port scan. Moreover, due to the novice user's basic visualization knowledge, the novice may not have the knowledge to extend the visualization. Using the NAVSEC system as explained in Section 7.2, the novice can use the expert user's suggested visualization by clicking the ”port scan” button in the attack tools palette. By clicking on this button, P3D uses the recommender algorithm and input from a group of expert users via the NAVSEC module to produce the most recommended visualization based on the cosine similarity function.

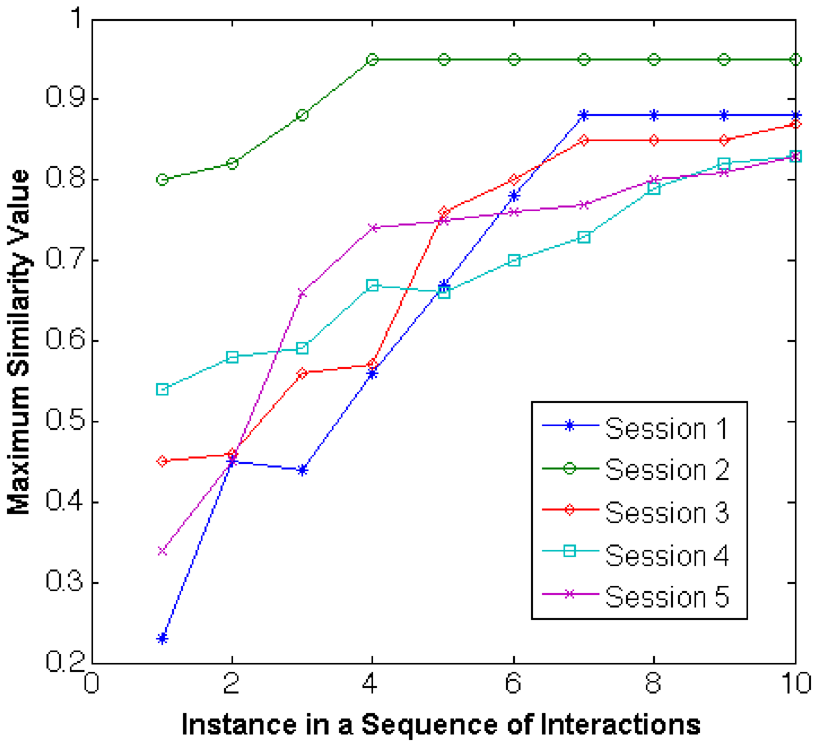


Figure 27: Convergence of interactions.

6.4 User Interaction Convergence Evaluation

In this section, the convergence of five sets of arbitrary instances of an active user session is shown (Figure 27). As discussed in Section 1 ”Nearest Neighbor Approach for NAVSEC”, an instance is denoted by the current set of interactions that was performed by the active user. For each instance, the recommended next interaction is performed using NAVSEC. Next, the maximum similarity score is computed after each recommended instance and plotted as shown in Figure 27. If the maximum similarity score continues to converge towards the value 1 after each consecutive step in the set of performed interactions, then the visualization produced by the active user is becoming similar to a visualization used to identify or detect a suspected attack by an expert user. Thus, as illustrated in Figure 21, NAVSEC is guiding an active user to a visualization completed by an expert user.

As shown in Figure 27, five sessions from active users are arbitrarily chosen from a group of active sessions. These sessions are compared to 40 emulated interaction vectors performed by expert users. Each interaction vector contains a set of 30 types of interactions (e.g., zoom out, zoom in, rotate, add left plane, add line glyphs). The line graph shows that as the set of interaction increases, each active session converges towards a value of 1 for the similarity score. The highest value is .95. This result suggests that with the use of NAVSEC, visualizations for the P3D tool converges towards an expert user's interaction set which can ultimately help novice users detect attacks.

6.5 User Study on Efficacy of NAVSEC

Both quantitative and qualitative data were collected from 15 users (13 males and 2 female) using NAVSEC. These users completed a pre-survey used to understand each user's expertise. From the survey, 10 users were denoted as novice users (basic knowledge in networking concepts or less than one year of experience), and 5 users were expert users (advanced knowledge with 3+ years of experience in a network security field). With each user, a total of 18 trials (9 trials with NAVSEC enabled and 9 trials with NAVSEC disabled) were recorded using emulated networking scenarios (e.g., DoS, DDoS, port scans, ftp transfer, disguised FTP transfer). Before the trial period, each participant was asked to perform a warm-up session to verify their expertise. In the warm-up session, the participants were asked to identify specific network attacks using P3D. For each trial of the experiment, the sequence of interactions used to complete this task was recorded. In total, we individually recorded 135 trials for 9 different scenarios (depicted in Table 4). For qualitative analysis, a post-survey using 5-point Likert scales were gathered.

The goal of our experiment was to compare the performance of NAVSEC vs. non-NAVSEC (NAVSEC module disabled). The performance was measured by examining the average number of interactions and similarity values between novice and expert users. The similarity is determined by computing the maximum cosine similarity between a vector for a set of interactions performed by a novice user and the vector performed by an expert user.

6.5.1 Interaction Analysis

As shown in Table 4, NAVSEC outperforms non-NAVSEC by showing a 24.8 % (average) reduction in the number of interactions. This reduction occurs because novice users could verify the next interaction by looking at an expert user's next interaction or next sequence of interactions. Furthermore, with NAVEC enabled, the user can view the sequence of interactions performed by an expert user, which helps prevent the novice user from performing interactions in error. Under conditions with background noise, the result show as much as a 33.7 % percentage increase (DoS with background noise) in the number of interactions. However, although the number of interactions increased, the response time also increased. This result is partly due to the fact that NAVSEC sends updates to the user by querying the NAVSEC recommendation engine in 5 second increments. Also, since the NAVSEC prototype was located on a separate monitor while user testing, the response time was also affected by the user looking away to another screen. The issue could be addressed by implementing a near realtime system where NAVSEC feedback is embedded within the same interface as the visualization tool.

Table 4: Average number of interactions for NAVSEC and non-NAVSEC.

Scenario	***NAVSEC***	***non-NAVSEC***	***Percentage increase***
Port scan	19.5	16.2	16.92
DoS	25.02	20.4	18.47
Port Confusion with DDoS and Scan	48.64	38.2	21.46
Port source confusion attack	50.4	38.3	24.01
DDoS using SYN Flood	60.2	47.9	20.43
DoS with background noise	63.4	42.8	32.49
2 port scans with large noise	67	44.4	33.73
FTP disguised attack	65.02	50.6	22.18
Legitimate traffic with no attacks	83	63.6	23.37
AVG	53.58	40.27	24.84

6.5.2 Similarity Analysis

Comparisons of the similarity of expert and novice user vectors reveal significant differences between NAVSEC and non-NAVSEC. As mentioned previously, a value of 0 denotes that the set of interactions is not related between an expert user and novice user. The value 1 means the set of interactions is similar. As shown in Table 5, the average similarity of novice users using NAVSEC shows a 21.04 % increase in similarity to the expert users compared to that of the novice users using non-NAVSEC. This result implies that with NAVSEC enabled, novice users perform more interactions similar to an expert than performing interactions with non-NAVSEC. However, although NAVSEC can guide a novice user to a visualization of an expert user, the qualitative analysis shows that some novice users could not understand the purpose of the next interaction. In these scenarios, users began blindly performing steps, which does not contribute to the comprehension of the the visualization system.

Table 5: Average similarity to expert users for NAVSEC and non-NAVSEC.

Scenario	*NAVSEC*	*non-NAVSEC*	*Percentage decrease*
Port scan	0.75	0.81	7.52
DoS	0.80	0.86	6.98
Port Confusion with DDoS and Scan	0.68	0.87	21.84
Port source confusion attack	0.73	0.85	14.12
DDoS using SYN Flood	0.79	0.90	12.22
DoS with background noise	0.56	0.86	34.88
2 port scans with large noise	0.59	0.82	28.05
FTP disguised attack	0.64	0.87	26.44
Legitimate traffic with no attacks	0.42	0.67	37.31
AVG	0.66	0.83	21.04

6.5.3 Qualitative Feedback

Participants filled out a post-experiment questionnaire using 5-point Likert scale to rate the effectiveness of the NAVSEC system. Furthermore, they were asked to comment on the improvements and enhancements for the NAVSEC system. These participants suggested that a description explaining why the interactions were performed be provided. Using this feedback, the NAVSEC system could be improved by attaching a comment to a sequence of interactions. Our findings show that 60 percent of novice users rated NAVSEC as very high easy-to-use and 40 % of the users rated NAVSEC as high ease-to-use. It is also noted

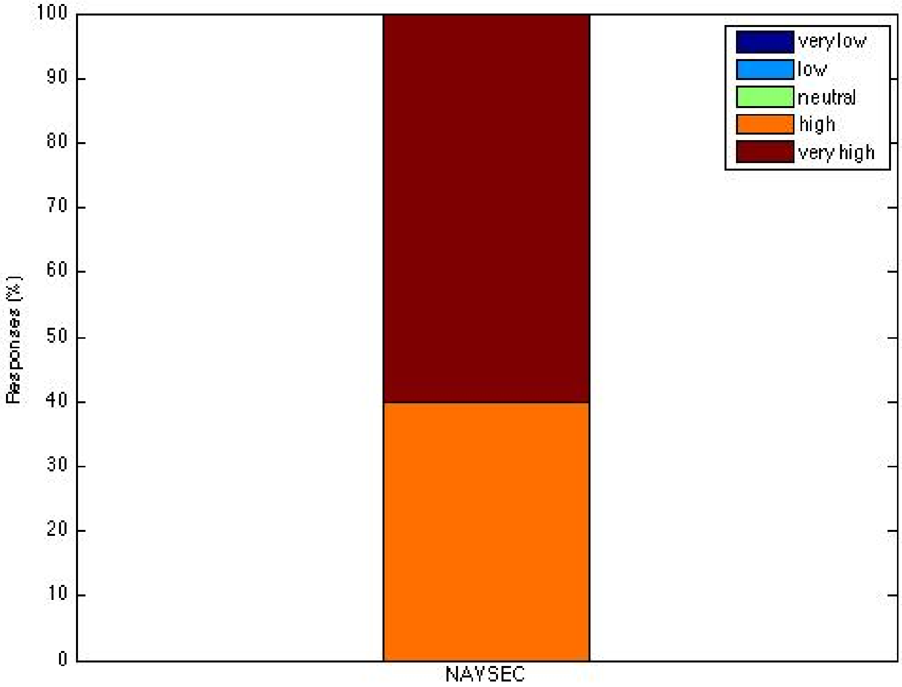


Figure 28: Distribution of responses to question: ”How do you rate the ease-of-use of the NAVSEC system?”.

that a post-survey’s open response mentioned that the NAVSEC could be distracting for scenarios where a decision for a network could be determined without NAVSEC enabled. This issue could be potentially solved by developing a button that allows the user to easily enable/disable NAVSEC.

CHAPTER 7

INTERSEC: AN INTERACTION SYSTEM FOR NETWORK SECURITY APPLICATIONS

As mentioned in Chapter 6, navigating within a 3D visualization for network security applications can be difficult because a large amount of network data is portrayed in a screen-size visualization. There are hundreds of details to track and different interactions must be utilized to navigate through a 3D environment. Thus, a new module for the FRE3DS framework, InterSec, was developed to provide a system for managing input gestures, tailored specifically to network security, that would enable the reduction of interactions to detect and identify a network attack. Furthermore, using Natural User Interface (NUI), InterSec allows a single gesture to serve as multiple serial gestures to achieve faster response times.

7.1 Examining the Use of NUIs

A Natural User Interface (NUI) is an interface that enables a user to interact with a computer in the same way the user interacts with the physical world, through using voice, hands, and body. For example, instead of using a keyboard and a mouse, a NUI allows users to speak to machines, stroke their surfaces using multi-touch systems, or perform gestures in the air. One NUI technique uses gestures in 3D space to manipulate a visualization environment. A gesture is a form of non-verbal communication in which visible bodily actions communicate particular messages.

Within the FRE3DS framework, NUI interfaces such as multi-touch and hand gestures are implemented using the Microsoft Kinect and 3M multi-touch monitor. The Kinect has proven to be an effective NUI device for interacting with applications using hand gestures, object recognition [75], and 3D scene generation [76]. The Kinect's infrared projector, infrared camera, and RGB camera are used to detect gestures in 3D space. Kinect's interactions are natural to the user and could be beneficial in assisting in the learnability of the network security visualizations. On the other hand, multi-touch systems supports all ten fingers as input, providing many more input degrees of freedom than mouse inputs. Research has shown that multi-touch interaction is about twice as fast as the mouse for selection [20]. Research has also shown the direct-touch nature of multi-touch interfaces accounts for 83% of a reduction in selection time and with additional training, bimanual strokes (i.e., using two hands) outperform making strokes serially by 10-15%, which further enhances the user's efficiency [20]. InterSec adopts the usage of NUIs, commonly-used in applications to minimize a user's response time in identifying harmful network security threats.

As shown in Figure 29, InterSec applies a gesture set from the NUIs to enable a user to perform multiple interactions quickly and to discover a network attack (e.g., DoS or an advanced stealthy port scan aiming to bypass a firewall or subvert an IDS). For example, two users, User 1 and User 2 use a NUI, but User 2 uses the traditional WIMP interface. By using a gesture set within InterSec, User 1 could use one gesture J_1 to implement a set of simultaneous interactions (I_1, I_2, I_3) portrayed by User 2. In this scenario, our intention is to reduce the number of individual interactions to assist in reducing the complexity of analyzing network activity in 3D environments.

In addition, due to the nature of multi-touch interactions, the time required to perform a gesture J_1 could be less than a single interaction, I_1, especially in the selection of tasks. For example, assume a network administrator is attempting to discover problematic nodes communicating via an Internet Relay Chat (IRC) channel within a botnet and decides to send his findings to a colleague using a 3D visualization tool. Using InterSec, the network administrator can use a NUI to perform a zoom/rotate interaction (a combination of zoom and rotate) rather than rotate and zoom interaction in series on a WIMP interface. Thus, the NUI would take one command instead of two separate commands and reduce the number of interactions. In addition, once the network administrator finds the command and control node, the network administrator may use a collaborative sharing gesture (e.g., four finger

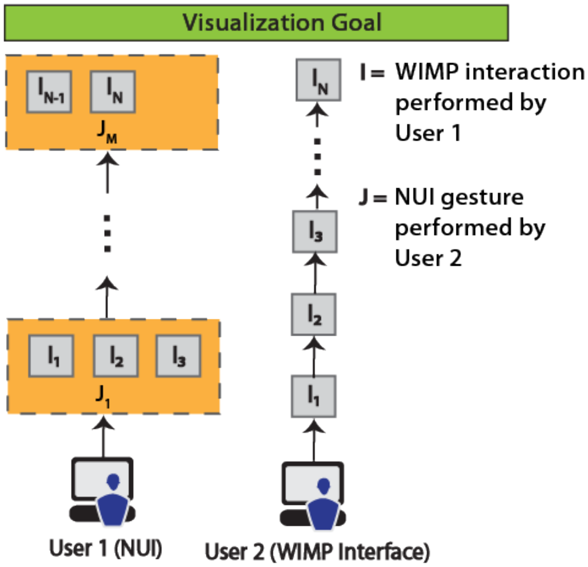


Figure 29: InterSec's application on a NUI vs. WIMP interface.

swipe) to notify a colleague by sending a filtered packet capture attached to an email for further investigation.

7.2 System Design and Implementation

As a recap, the InterSec module, as previously discussed in Chapter 6 and shown in Figure 20, is located in the IL of the FRE3DS framework. Within the IL, coordinates (C_{1i} ... C_{1N}) from touch, mouse, or hand inputs are generated from input devices. The input is stored in a buffer and the InterSec Module pulls the data from the input and converts the coordinates into interactions or gestures, depending on the type of input device. The interactions are passed through the InterSec process (as discussed later in the chapter), which could produce a set of commands or interactions based-on the gesture. Finally, these commands are then executed by the SSG and passed to the NAVSEC module.

InterSec takes advantage of direct-touch and bimanual input from state-of-the-art NUI technologies to aid in effectively navigating within 3D visualizations for network security applications. InterSec could be integrated into existing visualizations tools or used to promote new alternative NUI designs. Furthermore, a NUI system is presented, which helps developers of network security tools to build and manage gestures that require the coordination of multiple fingers and body limbs. The system uses NUI sensor devices (e.g., Kinect and touch monitor) so that the network administrator could use the advantages of devices with a high degree of freedom unlike traditional WIMP technologies.

Additionally, in order to allow for efficient monitoring and detection of network traffic and to evaluate its efficacy, InterSec has been designed and implemented. InterSec uses the FRE3DS framework [9] to convert gestures into a series of interactions for visualizing network attacks. The InterSec system consists of 4 stages: *Sensor*, *Gesture Recognition*, *Gesture Mapping*, *Visualization Manipulation* as illustrated in Figure 30. First, the raw data is sent from the NUI sensors (e.g., Kinect) to the Gesture-based detection system as input. Our detection system is adopted from GestureWorks [8], an HCI engine that contains pre-selected gestures, which produce the gesture input for gesture mapping. Each input is loaded as a plugin to the system in which there are a number of input devices that can be used with InterSec to produce the input data. For example, a LEAP Motion [64] plugin could be installed and interact with other 3D visualization tools in a physical 3D space. In our research, Kinect was used as the NUI sensor to input and to send the raw data as a series of gestures performed by larger body limbs (e.g., hands, arms, head). Through the use of the NUI sensor from *Kinect 3-point skeleton tracking* tool (Figure 31), Kinect tracks the motions of body limbs from a remote space to navigate through a 3D environment.

Kinect enables the user to perform several gestures simultaneously in order to convey multiple tasks at one time with fewer interactions. Unlike traditional WIMP technologies, which possess a single mouse cursor with only two spatial degrees of freedom, Kinect allows the user to employ body-motion gestures to perform more complex tasks in less time. During the *Gesture Recognition* stage, the input from the NUI sensor is converted into a gesture. After the gesture is recognized within the Gesture Recognition stage, the Task Mapper compares the gesture to a group of gestures in a relational database management system (RDMS) during the *Gesture Mapping* stage. If a match exists, the gesture is mapped

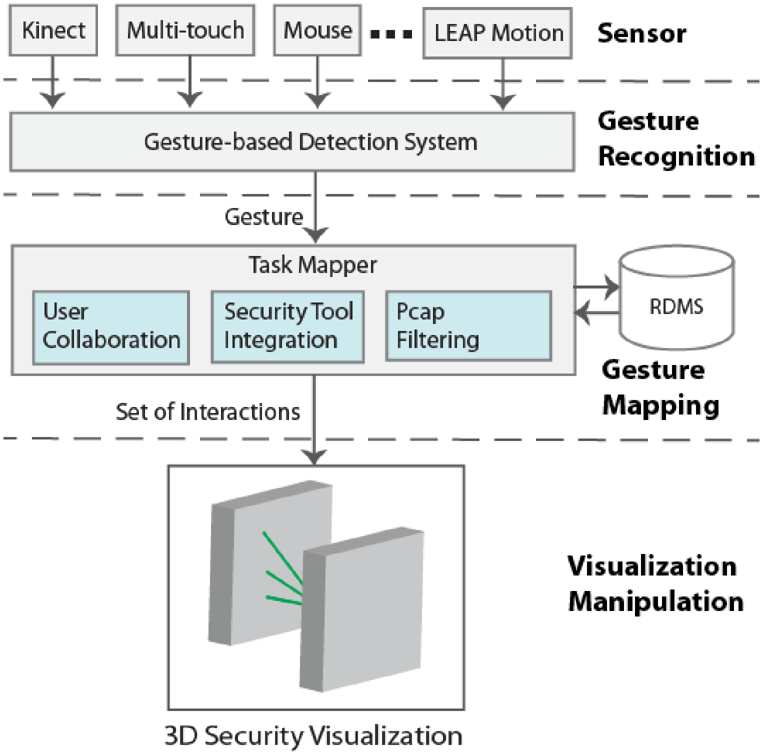
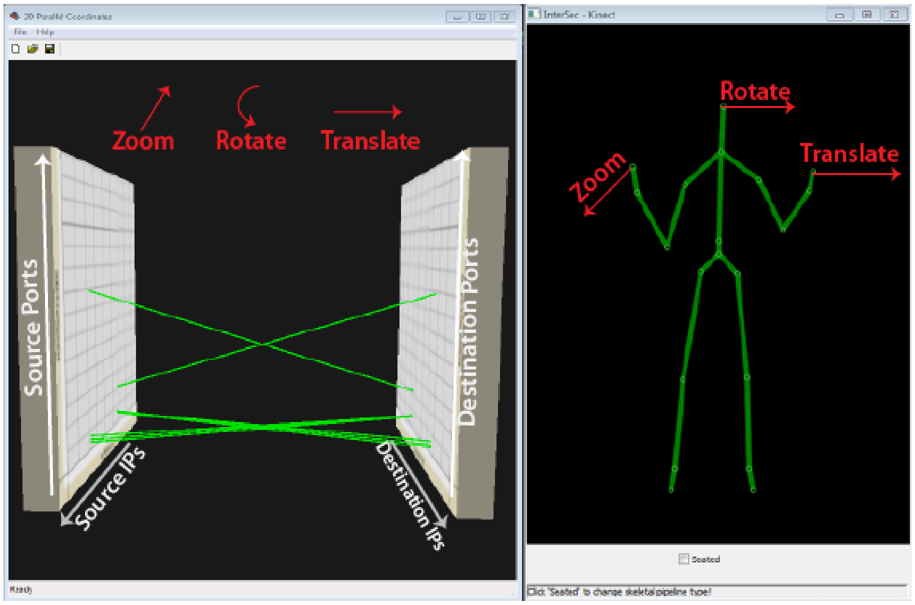


Figure 30: System design of InterSec.

to a task such as performing packet capture filtering. Once the gesture is recognized, the task is performed by filtering the captured data as follows: IP, Average Packet Size, Source and Destination Port, IP ID, Fragmentation bit, Timestamp, and TCP header flags such as SYN, FIN, URG, and PSH, and ACK. Common tasks for an administrator such as extracting relevant data through packet capture filtering, as opposed to storing entire network packets, is vital to the reduction of data sizes and results in more efficient data management. Finally, the *Visualization Manipulation* stage extracts and filters relevant parameters from the packets and performs a sequence of interactions. In some cases, if the interactions are independent, the interactions could be dispatched onto multiple processing threads to increase efficiency and further reduce response time.



Click 'Seated' to change skeletal pipeline type!

Figure 31: Kinect's [7] 3-point skeleton tracking tool

7.3 Gesture Set

In this sub-section, a sample gesture set from the GestureWorks [8] system is introduced and used in InterSec (Figure 32). The sample gesture set is used specifically for a network security analyst to reduce the number of interactions performed. As a result, response time of the user could be reduced. Currently, no tool exists, for network security administrators that proposes NUI gestures to reduce response times in 3D environments. A sample gesture set is as follows:

7.3.1 Five finger flick

Five finger flick employs five fingers of a single hand to be placed on the visualization rendering of the network and the motion of the fingers accelerates immediately before the fingers are released from the interface. This gesture denotes collaborative sharing between a user and a colleague. This gesture is important when a new attack has been detected because the *Five Finger Flick* can be used to quickly send a filtered packet capture file via email to a colleague or supervisor for further investigation.

7.3.2 Two finger rotate/zoom

Two finger rotate/zoom combines both the rotate and zoom gestures in order to quickly manipulate the visualization environment to produce fewer interactions than two separate gestures.

7.3.3 Four finger hold

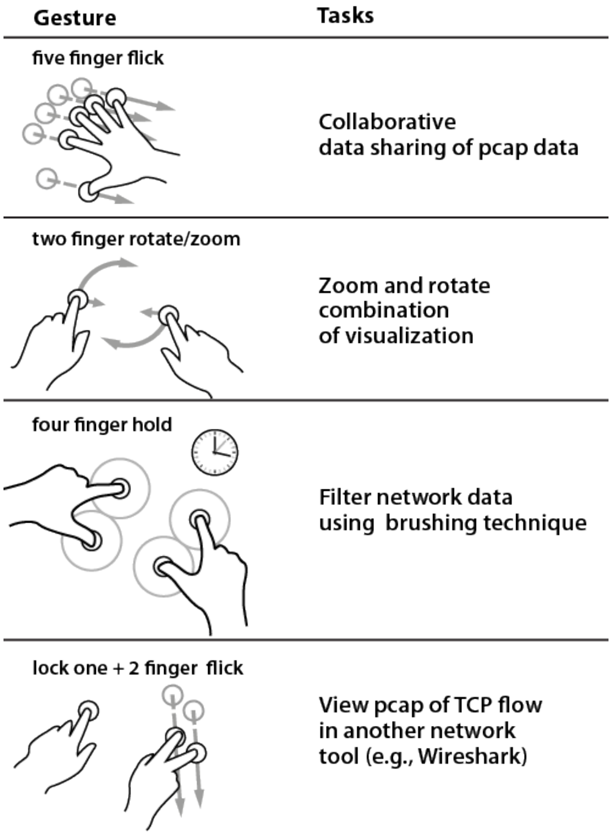
Four finger hold denotes four fingers touching the interface for a set period of time. These four touch points created on a multi-touch screen produces a visualization window. This visualization can be used as a filter mechanism to show only the packets that are visually shown in the selected visualization window. This allows users to quickly display pcap data of interest and filter unwanted pcap files in one gesture.

7.3.4 Lock one and 2 finger flick

Lock one and 2 finger flick refers to the selection of a visual data set using a single finger and flicking downward with two fingers. Once the user flicks down, InterSec selects the TCP flow and runs a companion tool, such as Wireshark, for the TCP flow data in order to investigate the textual data in greater detail.

InterSec uses the C++ Object Oriented Model-View-Controller paradigm for higher modularity and extensibility in the 3D visualization tools. We used a custom class using the Microsoft SDK. To render the raw data, we used an interaction testbed on the Kinect device and 3M's multi-touch monitor as shown in Figure 8. While using Kinect, we converted the data to gesture data using Kinect for Windows SDK. Kinect sensor data contains data from a four-element linear microphone array, IR emitter, and two cameras that deliver depth information, color images, and skeleton tracking data. For the multi-touch workstation, we used a 3M 32-inch monitor, which supports Capacitive Touch Technology (CTT) of up to 20 fingers. The multi-touch workstation is useful for collaboration of multiple network administrators on the same interface.

InterSec was implemented as a module of the FRE3DS framework and the functionality



Gesture	Tasks
five finger flick	Collaborative data sharing of pcap data
two finger rotate/zoom	Zoom and rotate combination of visualization
four finger hold	Filter network data using brushing technique
lock one + 2 finger flick	View pcap of TCP flow in another network tool (e.g., Wireshark)

Figure 32: NUI Gesture Set for Network Security Applications [8]

of InterSec is illustrated with use-case scenarios for analyzing a compromised host on the network using the Parallel 3D coordinate system (P3D) [6]. As discussed earlier, P3D is used because unlike most 3D counterparts, this tool has no theoretical limit in the number of network parameters that can be visualized. Therefore, P3D is able to better detect visualization attacks [40] in comparison to a 2D/3D scattered plot matrix [28], which uses stereoscopic 3D support and interactive techniques such as zooming and panning. To incorporate InterSec into P3D, P3D was extended by the modification of the interaction layer of the Framework for Rendering Enhanced 3D Stereoscopic Visualization (FRE3DS) [9]. Using FRE3DS, network administrators can easily and quickly develop various visualizations to efficiently investigate data.

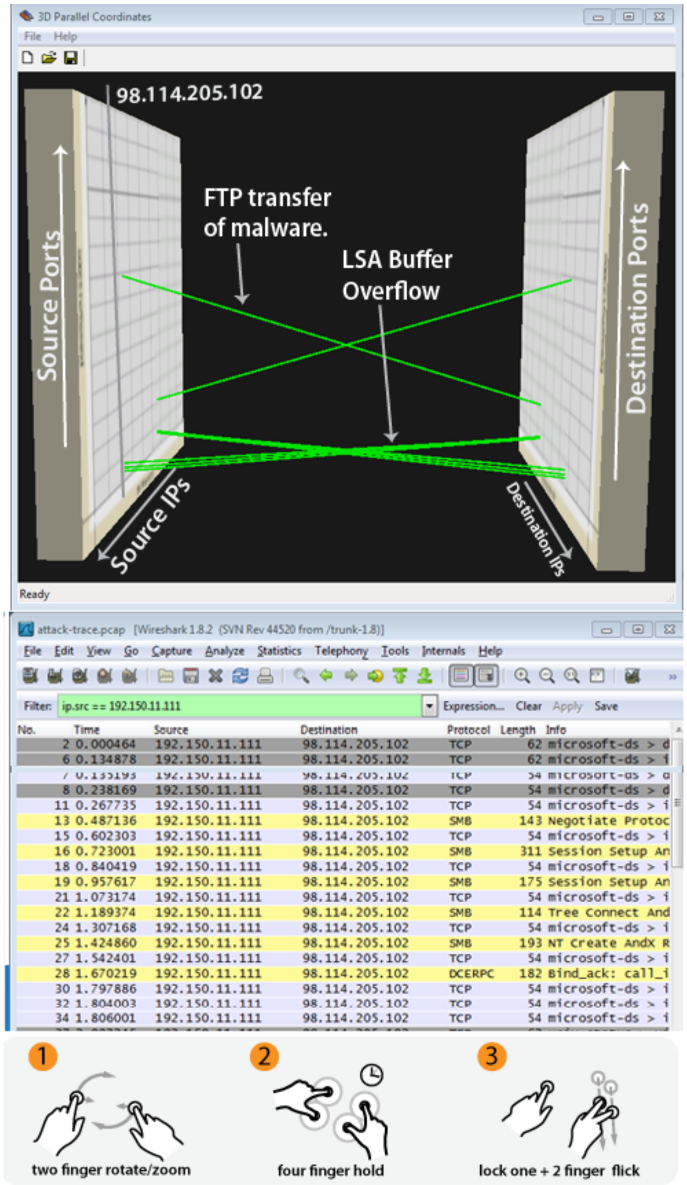
7.4 Use Case Analysis

In this section, InterSec is evaluated based on a real-world network attack data collected from the honeynet project. Specifically, the number of interactions is examined to perform tasks using mouse-keyboard vs. multi-touch interactions for P3D [6]. Also, the evaluation method assumes the model for determining the number of interactions by an error-free expert user. Although the user is practically error-proned, this model can be used as a preliminary indicator for determining how long it takes to perform a task, specifically a network attack. An expert is a user that knows the network task domain well and knows how to perform all the tasks that need to be completed. For evaluation, a live network data adopted from the Honeynet project's 2010 Forensic Challenge is used. The honeynet pcap portrayed a ”LSASS buffer overflow”, which caused a vulnerability (CVE-2003-0533), exploited by the Sasser worm. The attacker (source IP 98.114.205.102) established a TCP connection with the victim or honeypot (192.150.11.111) on Microsoft-ds port 445 and exploits the victim using the Windows Local Security Authority (LSA) Remote Procedure Call (RPC) service. From this exploit, the attacker opens a new port on the socket listening on port 1957 with a command shell bound to it. Finally, the victim initiated an FTP connection to the attacker and the attacker sent commands to the victim to download the malware. This scenario is beneficial because it is commonly used by attackers. Although the exploit commonly exists, we believe the methods for analysis could be applied more generally to new attacks and discoveries. Note that our intent of this work is to analyze the method for discovering a new attack rather than analyzing the attack itself.

InterSec is analyzed based on 4 common tasks: 1) Discover peculiar network activity (e.g., port scans). 2) Filter TCP flows. 3) Analyze IP/TCP header and TCP trace using a Wireshark filter. 4) Report to a colleague or upper management for verification and further investigation. These tasks are commonly used when visualizing network traffic. First, the user uses Kinect's 3-point skeleton tracking tool to configure and show the preferred visualization while away from the monitor. Next, the user employs a sequence of interactions to discover peculiar network activity such as port scans or denial of service attacks. In addition to these interactions, a user can also use InterSec to discover peculiar traffic by analyzing the source IP, source port, destination IP, and destination port by using a translation interaction. Next, the administrator can use two finger rotate/zoom interactions to view the ports from a source IP. After the administrator finds a peculiar port, the user filters the TCP data from a source IP using a four finger hold interaction. After the user filters the network data, the network administrator performs a lock one and 2 finger flick to investigate the TCP flow data and packet payloads in a supplement tool such as Wireshark (Figure 33). If the user finds a binded shell or a malicious executable transfer, then the user performs a five finger flick to send a filtered pcap file of the shell and/or executable to a colleague for further analysis.

In Figure 34, the number of interactions for NUI vs. mouse/keyboard interactions performed by an error-free expert user for each of the four tasks previously discussed are shown. This number is examine by exploring the number of minimum combinations of error-free interactions to successfully accomplish tasks. Each number along the x-axis in the figure represents each task.

With InterSec, at least a 50 % reduction in the number of interactions compared to traditional mouse/keyboards interactions for analyzing a Windows LSA RPC buffer overflow is shown. As a result, this reduction in interactions will significantly reduce a user's response time. A user's response time refers to the time taken by a user to react to a given visualization. In task 3 (Analyze IP/TCP header and TCP trace using a Wireshark filter), InterSec shows a substantial interaction reduction from 32 mouse/keyboard interactions in comparison to 5 NUI gestures using InterSec. This is partly due to the large amount of interactions to open Wireshark, scroll to an interesting packet, and investigate the packet payload. Using InterSec, these interactions could be replace with a few gestures. The analysis shows the number of interactions of a Windows LSA RPC buffer overflow analysis will


3D Parallel Coordinates
File Help
98.114.205.102
FTP transfer of malware.
LSA Buffer Overflow
Source Ports
Destination Ports
Source IPs
Destination IPs
Ready
attack-trace.pcap [Wireshark 1.8.2 (SVN Rev 44520 from /trunk-1.8)]
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter: ip.src == 192.150.11.111
Expression... Clear Apply Save
1
two finger rotate/zoom
2
four finger hold
3
lock one + 2 finger flick

```
No.    Time       Source           Destination       Protocol Length Info
     2 0.000464   192.150.11.111   98.114.205.102    TCP        62 microsoft-ds > d
     6 0.134878   192.150.11.111   98.114.205.102    TCP        62 microsoft-ds > i
     7 0.135193   192.150.11.111   98.114.205.102    TCP        54 microsoft-ds > d
     8 0.238169   192.150.11.111   98.114.205.102    TCP        54 microsoft-ds > d
    11 0.267735   192.150.11.111   98.114.205.102    TCP        54 microsoft-ds > i
    13 0.487136   192.150.11.111   98.114.205.102    SMB       143 Negotiate Protoc
    15 0.602303   192.150.11.111   98.114.205.102    TCP        54 microsoft-ds > i
    16 0.723001   192.150.11.111   98.114.205.102    SMB       311 Session Setup An
    18 0.840419   192.150.11.111   98.114.205.102    TCP        54 microsoft-ds > i
    19 0.957617   192.150.11.111   98.114.205.102    SMB       175 Session Setup An
    21 1.073174   192.150.11.111   98.114.205.102    TCP        54 microsoft-ds > i
    22 1.189374   192.150.11.111   98.114.205.102    SMB       114 Tree Connect And
    24 1.307168   192.150.11.111   98.114.205.102    TCP        54 microsoft-ds > i
    25 1.424860   192.150.11.111   98.114.205.102    SMB       193 NT Create AndX R
    27 1.542401   192.150.11.111   98.114.205.102    TCP        54 microsoft-ds > i
    28 1.670219   192.150.11.111   98.114.205.102    DCERPC    182 Bind_ack: call_i
    30 1.797886   192.150.11.111   98.114.205.102    TCP        54 microsoft-ds > i
    32 1.804003   192.150.11.111   98.114.205.102    TCP        54 microsoft-ds > i
    34 1.806001   192.150.11.111   98.114.205.102    TCP        54 microsoft-ds > i
```

Figure 33: P3D tool using InterSec's NUI Interactions

be reduced from 55 mouse/keyboard interactions to 13 NUI interfaces using InterSec. Although this analysis is error-free (i.e., no mistakes are performed by a user), our analysis is expected to apply more generally to more practical error-proned scenarios. In error-proned scenarios, a user performs non-optimal paths and introduces gestures that do not attribute to the detection of an attack. If the error introduced is constant across WIMP interfaces and NUIs, the reduction of interactions will still apply.

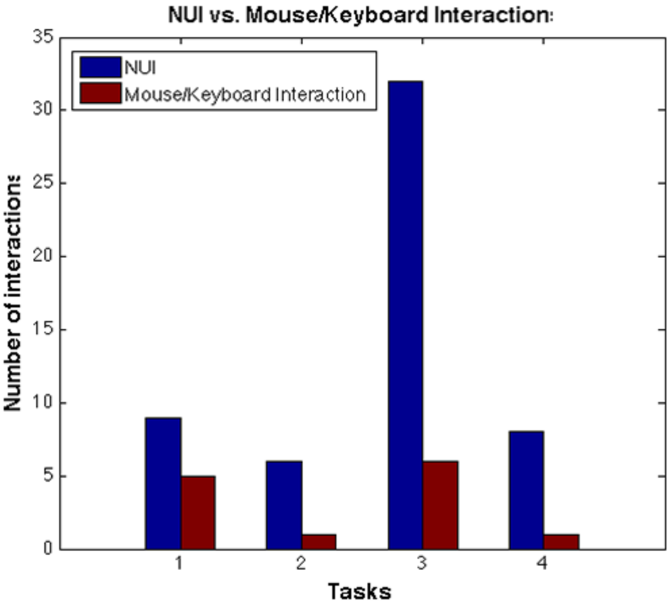


Figure 34: Minimum interactions for discovery of LSASS buffer overflow

7.5 User Study on Efficacy of InterSec

In this section, InterSec and WIMP interfaces are evaluated utilizing user testing methods. As previously mentioned, within the user testing, a user was presented with network security scenarios from a 3D visualization tool such as P3D and the task completion time and learnability were measured. The task completion time was measured by the recorded difference of the start and end time. Within user testing, two subsets of the users were asked to complete tasks using a mouse and multi-touch system respectively. Next, these tasks completion times of various users were compared.

Each scenario contains a 3D Parallel coordinate system using a WIMP interface or a NUI interface from the InterSec system. With user testing, 200 network attack scenarios are analyzed using InterSec to determine if InterSec validates the hypothesis of reducing the number of interactions and increasing response rates. Each task is evaluated using the timespan to complete each task. Also, both the number of user interactions and the sequence of interactions are measured. This data collection, in conjunction with post-survey responses provides valuable insights to understand the efficacy of the InterSec system. Similar to to P3D evaluation, both quantitative and qualitative data were collected from participant interactions with the user interface and the questionnaires, using the 5-point Likert scales.

7.5.1 Response Time Analysis

For our lab-based evaluation, 15 different subjects were recruited (13 male and 2 female), aged between 22 and 32 years (mean = 25, sd = 4.7) to explore multi-touch and NUI conditions. We asked the subjects to analyze 9 network scenarios. Similar to the procedure stated in section 5.4, we explained the task to the participants, asked each participant to sign a consent form, and the participants were instructed the use of InterSec. We also provided a list of gestures (provided in Figure 32) for both the WIMP and NUI experiments.

Table 6 lists the percentage increase in response times averaged over the 15 users. As expected, the response times of the InterSec is reduced approximately by 27.20 % largely due to large reduction in direct touch selection time and the multi-touch gesture set as shown in previous literature [20].

Participants from the main study found that WIMP was easier to use on initial attempts due to the familiarity of WIMP in other tools. However, as the participants became more familiar with the NUI, their interaction with the NUI became more natural. For example, when a user uses an interface on the first attempt, the ability of an interface to allow users to accomplish a task takes significantly longer than the second attempt because the user is unfamiliar with the visualization. During the warm-up sessions, our results shows that the user was able to reduce response time by 23 % in 3 attempts using the NUI.

7.5.2 Interaction Analysis

During experimentation, the number of user interactions for each session was recorded using programmable hooks. This data was used to determine the average number of user interactions for each scenario using both InterSec and WIMP interfaces. As shown in

Table 6: Average user response times of InterSec by scenario.

Scenario	*WIMP (s)*	*InterSec (s)*	*Percentage Decrease (%)*
Port scan	49.01	38.58	21.28
DoS	46.49	35.72	23.17
Port Confusion with DDoS and Scan	115.18	85.12	26.10
Port source confusion attack	125.67	80.72	35.77
DDoS using SYN Flood	126.95	87.86	30.79
DoS with background noise	182.86	132.24	27.68
2 Port Scans with large noise	180.69	132.24	26.81
FTP disguised attack	191.74	120.48	37.16
Legitimate traffic with no attacks	230.34	193.38	16.05
AVG	138.77	100.70	27.20

Table 7: Number of interactions of InterSec, on average.

Scenario	*WIMP*	*InterSec*	*Percentage Decrease (%)*
Port scan	19.50	13.40	31.28
DoS	25.02	18.20	27.26
Port Confusion with DDoS and Scan	48.64	32.20	33.80
Port source confusion attack	50.40	32.80	34.92
DDoS using SYN Flood	60.20	40.40	32.89
DoS with background noise	63.40	38.80	38.80
2 Port Scans with large noise	67.00	36.20	45.97
FTP disguised attack	65.02	55.60	14.49
Legitimate traffic with no attacks	83.00	70.60	14.94
AVG	53.58	37.58	30.48

Table 7, InterSec produces a 30.48 % (on average) reduction in the number of interactions for InterSec over its WIMP counterpart. This reduction occurs because InterSec's ability to use one interaction that would require the user to perform multiple interactions within WIMP interfaces. The result shows as much as a 45.97 % reduction for attacks (e.g., 2 Port Scans with large noise) that require the user to perform many zooms and rotates because InterSec can perform a zoom and rotate with one interaction.

To further investigate the number of interactions, users were asked to perform four tasks (mentioned in Section 7.4) using both InterSec and WIMP interfaces. For each task, an average of the total number of interactions of the users for both WIMP InterSec interfaces was calculated. As denoted in Figure 35, on average, the number of interactions is reduced by as much as 63 % for tasks that require the user to open Wireshark and apply filters. This reduction occurs because the number of interactions the user performs to open tools

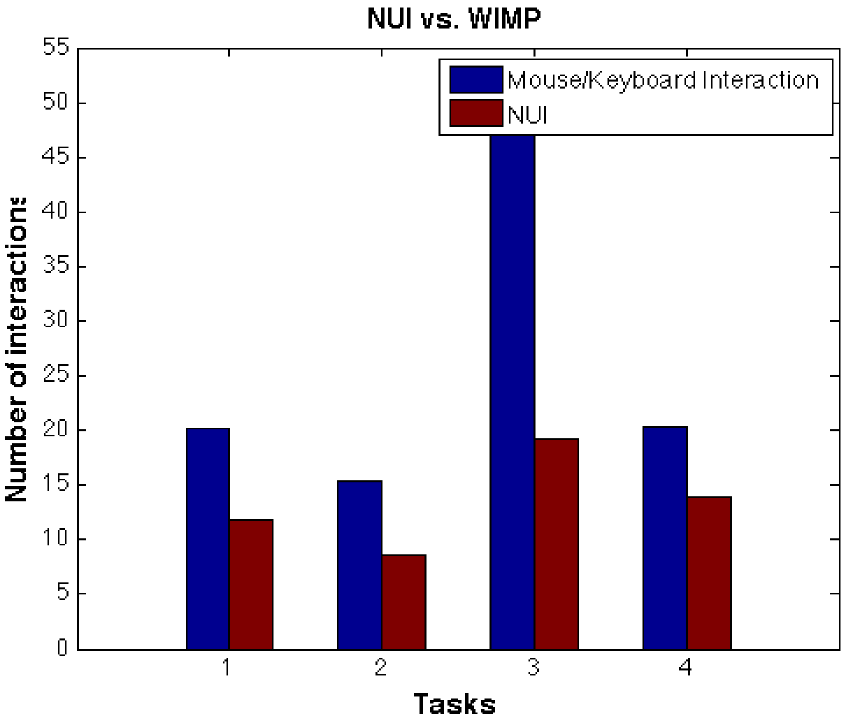


Figure 35: Number of user interactions from LSASS buffer overflow tasks mentioned in Section 7.4

.

such as Wireshark could be reduced to one interaction with InterSec. However, users found that searching through packets within Wireshark was difficult on a multi-touch monitor because the buttons were too small for touch interface. This observation is expected due to Wireshark's limited multitouch support. Wireshark developers could enhance the multi-touch capabilities by integrating its UI with InterSec. The results show that the total average reduction for the number of interactions is 45 % using the InterSec system in comparison to its WIMP counterpart.

7.5.3 Qualitative Feedback

To conduct qualitative analysis, the participants were given 5-point Likert scale questionnaires to understand the ease-of-use of the InterSec system. As shown in Figure 36, 60 % of the users ranked the InterSec system as very high or as having a high ease-of-use. Also, 40 % of the users ranked the system as having a neutral ease-of-use partly due to difficulties of remembering gestures and using tools (e.g., Wireshark) on the touch monitor that is primarily designed for WIMP interfaces. The qualitative feedback from the study revealed that some participants developed strategies for learning gestures. For example, one participant used the imagery (four finger hold is like creating a window or frame with your fingers). In some cases, participants found difficulty memorizing gestures and these participants constantly referred to the cheat sheet. It is assumed that these participants did not attach the gesture to a natural gesture like taking a picture or physically sliding a task to a colleague using a five-finger swipe. This issue can be addressed by introducing a natural example of why the gesture was chosen for a task. This method allows the user to attach a natural action. For example, physically pushing a sheet of paper to a colleague is similar to five finger swipe because in both cases, data is sent to another user. Also, during open responses, some users mentioned potential arm fatigue for long term usage. Although the experiment was performed on 32 inch multitouch display, the monitor could be reduce to a smaller display to reduce moving the arms large distances. Also, the monitor could be ergonomically positioned to a slight vertical tilt to further reduce arm strain.

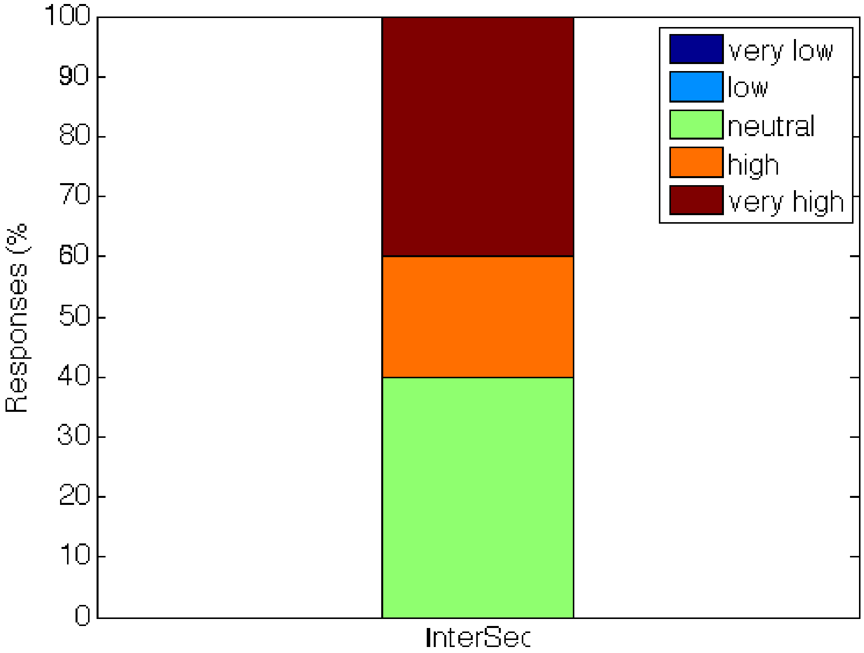


Figure 36: Distribution of responses to question: ”How do you rate the ease-of-use of InterSec system?”.

CHAPTER 8

CONCLUSIONS AND FUTURE WORK

In this thesis, we developed efficient interactions and 3D visualization techniques for a network administrator to enhance situational awareness using stereoscopic 3D technologies, reduce task completion time using a recommender system, decrease user error by increasing the visualization space and increase learnability of complex visualizations for network security applications by incorporating NUIs. These techniques utilize multi-dimensional data in a format suitable for simplified human interpretation and analysis. Although there have been several studies on 2D/3D visualization techniques for network analysis, there has been little work on stereoscopic 3D techniques, NUI techniques and recommender systems for network security visualization. The following four areas were investigated in this thesis and conclusions for each of them are described in the Research Contributions subsection (Section 8.1).

8.1 Research Contributions

8.1.1 Designing a Framework for Rendering Enhanced 3D Visualizations

In Chapter 3, the FRE3DS framework was presented for administrators to absorb and perceive large amounts of visual information, particularly when the 3D senses were enabled by binocular vision. Furthermore, it established a way to allow administrators to evaluate the effects of emerging technologies in computer visualization and interaction. In this thesis, the effects of the stereoscopic and NUI technologies were examined in network security applications. From this framework, 4 prototypes were developed and evaluated with use-cases and usability testing.

8.1.2 3D Stereoscopic Vulnerability Assessment Tool (3DSVAT)

The 3DSVAT tool uses the FRE3DS framework to reveal vital vulnerability characteristics of local area network data and determine correlations of vulnerability data between nodes. This is essential for strategically determining which node to patch first and rapidly determining highly vulnerable nodes on networks. 3DSVAT rendered both monocular and binocular depth cues to enhance the user's experience, perform faster analysis of the network vulnerability data, reduce clutter, and increase efficiency

8.1.3 Parallel 3D Coordinate Visualization (P3D)

P3D used the FRE3DS framework to understand and analyze scans or attacks used to mislead and overwhelm the user analyzing large networks. P3D revealed vital scanning characteristics of data and determined correlations between data and attacker nodes on a network. P3D is essential for strategically determining distributed coordinated attacks. Specifically, this thesis showed that when using P3D, it is less likely to obscure data through occlusion attacks particularly meant to visually overwhelm the user.

8.1.4 NAVSEC Recommender System

In Chapter 6, NAVSEC was presented, and it was shown that it can assist users in navigating 3D visualizations and to reduce the possible number of interactions within a visualization. Although there have been several studies on 2D/3D visualization techniques for network analysis, there has been little work on addressing the issue of navigation complexity within these visualization techniques [40]. NAVSEC uses advanced visualization recommender techniques based on a database of interaction sequences of an expert community to (1) enhance the novice user's experience, (2) provide easier understanding of 3D network security in complex visualization environments, and (3) perform faster analysis of the network data thereby increasing efficiency. NAVSEC was implemented using P3D and the FRE3DS framework [9] and vital characteristics of a node on a network were revealed. Specifically, it was shown that using NAVSEC, novice users are less likely to be confused when discovering advance attacks.

8.1.5 Interaction System for Network Security

The use of novel interactions to analyze multidimensional data is discussed in Chapter 7. Although there have been several studies on 2D/3D visualization techniques for network analysis, there has been little work on interaction techniques aimed at understanding and analyzing attacks. InterSec enables administrators to develop NUI gesture sets to reduce the interactions performed on a 3D security tool and assist users in navigating in the 3D visualizations. In addition, the interaction space was extended using InterSec and a gesture set was implemented that was used to simulate multiple actions used to discover new data. InterSec was used to reveal vital scanning characteristics of data and to determine correlations between data and attacker nodes on a network.

8.2 Future Work

In the future, this work could be expanded to the IPv6 address space, implement more user interactions in InterSec, and can be deployed into small organizations for a more exhaustive user analysis. Other improvements are mentioned below.

8.2.1 Long Term User Study

The evaluation of user analysis in the current design is rather limited, and assumptions are largely based on a small set of users. To further verify claims addressed in this thesis, the FRE3DS framework could be deployed to a large number of users across multiple organizations in the the network security field. To perform a widely distributed network of user testing, the framework should be refined to address scalability (such as cloud computing and load balancing) of the platform. In addition, to prepare for adoption from many users, the framework must must be made available on multiple platforms (e.g., OSX, Windows, and Linux).

8.2.2 Effects of Multiple Visualization Designs

One of the biggest challenges in visualization is to address the best design for an network incident. In this case, the best design refers to reduced response times, low error rates, and allowing the users to gain new insights into a large dataset. Although the current design evaluates stereoscopic conditions for one tool. The current design does not deal with the evaluation of various tools under stereoscopic conditions. In the future, many different tools can be evaluated to produce the optimal user-centered design.

APPENDIX

Consent Form

Georgia Institute of Technology

Project Title: Information process of 3D parallel network scans

Principle Investigator: Dr. Raheen Beyah

Co-investigators: Troy Nunnally, Kulsoom Abdullah, Penyen Chi, Selcuk Uluagac

Consent title: Main 12/1/12v1

Research Consent Form

You are being asked to be a volunteer in a research study.

Purpose:

The purpose of this study is to evaluate the benefits of stereoscopic 3D in network security applications by portraying information from a parallel network scan in 3D will help make network traffic easier and faster to decipher.

Procedures:

If you decide to participate, your part will involve completing a paper-and-pencil survey during a scheduled session. It is anticipated that completion of the survey will take about 90 minutes or at most 2 hours of your time. Your responses on the survey will remain anonymous and kept under a code number rather than any personal identifier. Only group-level results will be used for research purposes.

If you decide to be in this study, your part will involve one visit to the laboratory for approximately one to two hours. In this visit you will be asked to first look at the result of a 2D parallel scanner. This information is generated through the Georgia Tech network and will be from several leading 2D parallel network scanners. Then the volunteer participant, you, will be asked to wear a pair of 3D shutter vision glasses and be seated in front of a 3D monitor. Then a series of the same network traffic information previously generated will be displayed, but this time using the 3D parallel scanner. Then a research assistant will ask you some questions about the differences between the two ways of portraying network traffic. You will be asked to compare and contrast the generated images for clarity of information and easiness of view. The total amount of time you will be in the laboratory is no more than 2 hours. Remember, you may stop at any time.

Risks or Discomforts:

The study involves no more than minimal risk associated with wearing 3D shutter vision glasses and watching a 3D monitor.

Benefits:

You are not likely to benefit in any way from joining this study. We hope that what we learn will someday help network administrators to process network data more efficiently. The speed up in information processing will help to initiate a protective response against network threats.

Compensation to You:

There is no compensation for participation.

Confidentiality:

The following procedures will be followed to keep your personal information confidential in this study: The data collected about you will be kept private to the extent allowed by law. To protect your privacy, your records will be kept under a code number rather than by name. Your records will be kept in locked files and only study staff will be allowed to look at them. Your name and any other fact that might point to you will not appear when results of this study are presented or published. Your privacy will be protected to the extent allowed by law. To make sure that this research is being carried out in the proper way, the Georgia Institute of Technology IRB may review study records. The Office of Human Research Protections and/or the Food and Drug Administration may also look over study records during required reviews.

Costs to You:

There are no costs to you, other than your time, for being in this study.

In Case of Injury/Harm:

If you are injured as a result of being in this study, please contact the Principal Investigator, Troy Nunnally at troy.nunally@gatech.edu. Neither the Principal Investigator nor Georgia Institute of Technology has made provision for payment of costs associated with any injury resulting from participation in this study.

Participant Rights:

Your participation in this study is voluntary. You do not have to be in this study if you don't want to be. You have the right to change your mind and leave the study at any time without giving any reason and without penalty. Any new information that may make you change your mind about being in this study will be given to you. You will be given a copy of this consent form to keep. You do not waive any of your legal rights by signing this consent form.

Questions about the Study:

If you have any questions about the study, you may contact
Troy Nunnally
troy.nunnally@gatech.edu
If you have any questions about your rights as a research participant, you may contact

Ms. Kelly Winn, Georgia Institute of Technology
Office of Research Compliance, at (404) 385-2175.

If you sign below, it means that you have read (or have had read to you) the information given in this consent form, and you would like to be a volunteer in this study.

Participant Name (printed)

Participant Signature

Date

Signature of Person Obtaining Consent

Date

User Test Script

The computer monitor should show the desktop and scenarios should be closed.

Hi, *User's name*. My name is *Investigator's Name*, and I will be walking you through this session today.

Before we begin, I would like to verify that you are 18+ years of age. Are you 18 or above 18 years of age?

I have some information for you, and Im going to read it to make sure that I cover everything. You probably already have a good idea of why we asked you here, but I will briefly go over it again. Were asking people to try using a 3D network security tool that is in development to determine whether it works as intended. In this visit, you will be asked to compare and contrast generated images using 3D shutter glasses for clarity of information and ease of viewing. These images will be visualizations from network traffic data generated from network scanners. Then, a research assistant will ask you a few questions about the differences between the two ways of portraying network traffic. The total amount of time you will be in the laboratory will not exceed 2 hours. The session should range between 60 min to 90 minutes.

The first thing I want to clarify is that we are testing the tool, not you. Do not worry about making mistakes or incorrect answers. We want you to feel free to use the tool naturally. Testing here today is one place where you dont have to worry about making mistakes.

As you use the tool, Im going to ask you to think out loud as much as possible. In other words, please talk out your thoughts and the actions you are taking to complete a task. This will be a big help to us.

Please do not hold anything back or worry that youre going to hurt our feelings. We want to hear your honest reactions so that we can improve this tool. If you have any questions as we go along, please ask. I may not be able to answer them right away during testing as we are interested in how people perform when they do not have guidance from someone. If you still have any questions when everything is completed, I will try to answer them then. Please let me know if you need to take a break at any time. I will be here taking notes and observing this session to help us understand the user experience and further evaluate the tool. Additionally, the actions you perform and the time to complete each task will be recorded.

Now that you have a basic understanding of our tests, do you have any questions?

Answer questions

Great, please read and fill out this consent form so that we can proceed with the testing process. Please ask any questions that you may have about the consent form once you have finished reading it.

Give them the consent form and a pen

Ok. Before we look at the tool, Id like you to take a quick survey to get an understanding of your expertise level.

Give them the pre-survey

Here is your task: you are a security network analyst for a huge company and you are asked to look at the network traffic and point out any peculiar activity and explain what is going on as quickly as you can by looking at various visualization images.

In front of you is a desktop with the tool.

Optional script for 3D: In addition, there is a pair of 3D glasses. Please put on the glasses and let me know if 3D properly works on the screen. If it works, please let me know. Now, I will explain the components of the user interface and the visualization techniques involved in the session so you completely understand the interface. I will also introduce the various scenarios understand the concept of the visualization techniques they will interface with during the experiment. The scenarios contain both malicious and non-malicious network activity.

Explain the components and perform warm-up session.

Once you press the start button, we would like you to do a narrative of what you think looks peculiar on the network and explain everything you see. If you do not see anything peculiar, then let us know that as well. While viewing the visualization, please answer the questions on the provided on the post survey form. Once you have fully completed all the questions, press the end button.

Repeat for all scenarios.

Do you have any questions for me, now that we are finished?

Thank them and escort them out.

Recruitment: Email

From: troy.nunnally@gatech.edu Subject: Volunteers Wanted for a Research Study

Hello,

We are currently looking for volunteers for a Research Study that investigates network scans/attacks using 3D shutter glasses and multi-touch systems. The purpose of the research is to evaluate whether portraying information from network activity in three dimensions (3D) or with multi-touch system will reduce human error and decrease task completion time. If you decide to participate, your visit to the laboratory will take 60 - 90 minutes but no more than 2 hours of your time. The research study will be conducted on January 22- February, 2014 located in the Georgia Institute of Technologys Klaus Building, room 3361. In this visit, you will be asked to compare and contrast generated 2D images and 3D images using 3D shutter glasses for clarity of information and ease of view or navigate within network application using a multi-touch system.

With your assistance, we hope to gain significant insights into understanding network data more efficiently and initiate a proactive response against network security threats. The study requires basic knowledge in computer networking.Subjects must be 18+ years of age. If you are interested in participating, please sign up for an available time slot at SignUp Genius or email Troy Nunnally at troy.nunnally@gatech.edu.

The PI for this project is Raheem Beyah. If you have any questions for the Principal Investigator (PI), Raheem Beyah, by email at rbeyah@ece.gatech.edu or by telephone 404-894-2531.

Thanks,

Troy Nunnally

Pre-survey

Survey given before the experimentation starts. This survey ensuresWe will assume that most of the participants possess a rudimentary knowledge of network security. Thus, the participants might not be familiar with the specific types of port scans or attacks. However, they will possess adequate knowledge to provide insight about any peculiar activity and describe the events.

Knowledge of Internetworking (check one level):

Novice Amateur Advanced Expert

Knowledge of Network Security (check one level):

Novice Amateur Advanced Expert

List Internetworking related classes taken, if any:

List any network attacks you may know, if any:

Survey

Survey given after each scenario

We will assume that most of the participants possess a rudimentary knowledge of network security. Thus, the participants might not be familiar with the specific types of port scans or attacks. However, they will possess adequate knowledge to provide insight about any peculiar activity and describe the events.

Session 1 (2D vs 3D)

Session 1 Scenarios:

1. Please Identify the source IP and the source port. (timed)

2. Please identify the destination IP and the destination port or port range. (timed)

3. Does any port scan exist (Is any port scan being performed)? (timed?) If so, can you describe the port scan in detail? (verbally explained)

4. How many attacks are present in the network data? (possibly timed)

5. Rate the effectiveness of each visualization. 1-10 (10 being the best)

Session 2 (3D vs 3D stereoscopic)

Session 2 Scenarios:

1. Can you list the number of vulnerable nodes? (timed)

2. List the nodes that are most vulnerable. (timed)

3. On ”specific node” list the vulnerabilities. (timed)

4. List the safe nodes (timed)

5. Rate the effectiveness of the Stereoscopy. 1-10 (10 being the best)

REFERENCES

[1] R. Ball, G. A. Fink, and C. North, “Home-centric Visualization of Network Traffic for Security Administration,” in *Proceedings of the 2004 ACM Workshop on Visualization and Data Mining for Computer Security*, VizSEC/DMSEC ’04, pp. 55–64, 2004.

[2] Y. Livnat, J. Agutter, S. Moon, R. Erbacher, and S. Foresti, “A visualization paradigm for network intrusion detection,” in *Information Assurance Workshop, 2005. IAW ’05. Proceedings from the Sixth Annual IEEE SMC*, pp. 92–99, June 2005.

[3] Z. Kan, C. Hu, Z. Wang, G. Wang, and X. Huang, “Netvis: A network security management visualization tool based on treemap,” in *Advanced Computer Control (ICACC), 2010 2nd International Conference on*, vol. 4, pp. 18–21, March 2010.

[4] T. Takada and H. Koike, “Tudumi: Information Visualization System for Monitoring and Auditing Computer Logs,” in *Information Visualisation, 2002. Proceedings. Sixth International Conference on*, pp. 570–576, 2002.

[5] S. Lau, “The Spinning Cube of Potential Doom,” *Commun. ACM*, vol. 47, pp. 25–26, Jun. 2004.

[6] T. Nunnally, P. Chi, K. Abdullah, S. Uluagac, and R. Beyah, “P3D: A Parallel Coordinate System for Network Security,” in *Proceedings of the IEEE International Conference on Communications (ICC)*, June 2013.

[7] M. Kinect, “Microsoft Kinect,” 2013.

[8] G. Works, “GestureWorks,” 2013.

[9] T. Nunnally, A. S. Uluagac, J. Copeland, and R. Beyah, “3DSVAT: 3D Stereoscopic Vulnerability Assessment Tool for Network Security,” in *Proceedings of the 37th IEEE Conference on Local Computer Networks (LCN)*, 2012.

[10] T. Nunnally, K. Abdullah, U. Selcuk, J. Copeland, and R. Beyah, “Navsec: A Recommender System for 3D Network Security Applications,” in *Proceedings of the tenth International Symposium on Visualization for Cyber Security*, VizSec, 2013.

[11] T. Nunnally, K. Abdullah, U. Selcuk, J. Copeland, and R. Beyah, “InterSec: A Interaction System for 3D Network Security Applications,” in *in submission to Proceedings of the Global Communications Conference*, GLOBECOM, 2014.

[12] D. Keim, “Information Visualization and Visual Data Mining,” *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–8, Mar 2002.

[13] W. Ark, C. D. Dryer, T. Selker, and S. Zhai, "Representation Matters: The Effect of 3D Objects and a Spatial Metaphor in a Graphical User Interface," in *Proceedings of HCI on People and Computers*, pp. 209–219, 1998.

[14] C. Ware, *Information Visualization Perception for Design*, vol. 1. Morgan Kaufmann, 2004.

[15] H. Koike and K. Ohno, "SnortView: Visualization System of Snort Logs," in *Proceedings of the ACM Workshop on Visualization and Data Mining for Computer Security (VizSEC/DMSEC)*, pp. 143–147, 2004.

[16] B. Lee, P. Isenberg, N. Riche, and S. Carpendale, "Beyond Mouse and Keyboard: Expanding Design Considerations for Information Visualization Interactions," *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 12, pp. 2689–2698, 2012.

[17] B. Shneiderman and P. Maes, "Direct Manipulation vs. Interface Agents," *interactions*, vol. 4, pp. 42–61, Nov. 1997.

[18] J. Guenther, F. Volk, and M. Shaneck, "Proposing a Multi-touch Interface for Intrusion Detection Environments," in *Proceedings of the Seventh International Symposium on Visualization for Cyber Security*, VizSec, pp. 13–21, 2010.

[19] G. Goth, "Brave NUI world," *Commun. ACM*, vol. 54, pp. 14–16, Dec. 2011.

[20] K. Kin, *Investigating the Design and Development of Multitouch Applications*. PhD thesis, EECS Department, University of California, Berkeley, Dec. 2012.

[21] M. Friendly, "Milestones in the history of thematic cartography, statistical graphics, and data visualization," in *13th International Conference on Database and Expert Systems Applications (DEXA)*, pp. 59–66, Press, 1995.

[22] X. Li, Q. Wang, L. Yang, and X. Luo, "The research on network security visualization key technology," in *Multimedia Information Networking and Security (MINES), 2012 Fourth International Conference on*, pp. 983–988, 2012.

[23] C. Kan, C. Hu, Z. Wang, G. Wang, and X. Huang, "Netvis: A Network Security Management Visualization Tool based on Treemap," in *Proceedings of the 2nd International Conference on Advanced Computer Control (ICACC)*, vol. 4, pp. 18–21, Mar. 2010.

[24] K. Lakkaraju, W. Yurcik, and A. Lee, "NVisionIP: Netflow Visualizations of System State for Security Situational Awareness," in *Proceedings of the ACM Workshop on Visualization and Data Mining for Computer Security (VizSEC/DMSEC)*, pp. 65–72, 2004.

[25] K. Abdullah, C. Lee, G. Conti, J. Copeland, and J. Stasko, "IDS RainStorm: Visualizing IDS Alarms," in *Proceedings of the IEEE Workshop on Visualization for Computer Security (VizSEC)*, pp. 1–10, Oct. 2005.

[26] S. Krasser, G. Conti, J. Grizzard, J. Gribschaw, and H. Owen, "Real-time and Forensic Network Data Analysis using Animated and Coordinated Visualization," in *Proceedings fo the Sixth IEEE SMC Information Assurance Workshop (IAW)*, pp. 42–49, Jun. 2005.

[27] R. Erbacher, "Intrusion behavior detection through visualization," in *Systems, Man and Cybernetics, 2003. IEEE International Conference on*, vol. 3, pp. 2507–2513 vol.3, 2003.

[28] H. Choi, H. Lee, and H. Kim, "Fast Detection and Visualization of Network Attacks on Parallel Coordinates," *Computers and Security*, vol. 28, no. 5, pp. 276 – 288, 2009.

[29] X. Yin, W. Yurcik, M. Treaster, Y. Li, and K. Lakkaraju, "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*, VizSEC/DMSEC '04, pp. 26–34, 2004.

[30] B. Claise, "Cisco Systems NetFlow Services," 2004.

[31] G. Conti and K. Abdullah, "Passive Visual Fingerprinting of Network Attack Tools," in *Proceedings of the ACM Workshop on Visualization and Data Mining for Computer Security(VizSEC/DMSEC)*, pp. 45–54, 2004.

[32] G. Conti and K. Abdullah, "Passive Visual Fingerprinting of Network Attack Tools," in *Proceedings of the 2004 ACM Workshop on Visualization and Data Mining for Computer Security*, VizSEC/DMSEC '04, (New York, NY, USA), pp. 45–54, ACM, 2004.

[33] H. Hauser, F. Ledermann, and H. Doleisch, "Angular Brushing of Extended Parallel Coordinates," in *Proceedings of the IEEE Symposium on Information Visualization (InfoVis)*, pp. 127–, 2002.

[34] Z. Jiawan, L. Liang, L. Liangfu, and Z. Ning, "A Novel Visualization Approach for Efficient Network Scans Detection," in *Proceedings of the International Conference on Security Technology (SECTECH)*, pp. 23 –26, Dec. 2008.

[35] Snort, "Snort."

[36] R. Shi, F. Zhou, and Y. Zhao, "Netsecradar: A real-time visualization system for network security: Vast 2012 mini challenge. award: Honorable mention for interesting use of radial visualization technique," in *Proceedings of the 2012 IEEE Conference on Visual Analytics Science and Technology (VAST)*, VAST '12, (Washington, DC, USA), pp. 281–282, IEEE Computer Society, 2012.

[37] Y. Zhao, F. Zhou, X. Fan, X. Liang, and Y. Liu, "Idsradar: a real-time visualization framework for ids alerts," *Science China Information Sciences*, vol. 56, no. 8, pp. 1–12, 2013.

[38] E. Le Malécot, M. Kohara, Y. Hori, and K. Sakurai, “Interactively Combining 2D and 3D Visualization for Network Traffic Monitoring,” in *Proceedings of the 3rd International Workshop on Visualization for Computer Security (VizSEC)*, pp. 123–127, 2006.

[39] G. Hubona, P. Wheeler, G. Shirah, and M. Brandt, “The Relative Contributions of Stereo, Lighting, and Background Scenes in Promoting 3D Depth Visualization,” *ACM Transactions on Computer-Human Interaction*, vol. 6, pp. 214–242, Sep. 1999.

[40] G. Conti, M. Ahamad, and J. Stasko, “Attacking Information Visualization System Usability Overloading and Deceiving the Human,” in *Proceedings of the Symposium on Usable Privacy and Security (SOUPS)*, pp. 89–100, 2005.

[41] “RT Graph 3D.”

[42] J. McPherson, K. Ma, P. Krystosk, T. Bartoletti, and M. Christensen, “Portvis: A Tool for Port-based Detection of Security Events,” in *Proceedings of the 2004 ACM Workshop on Visualization and Data Mining for Computer Security (VizSEC/DMSEC)*, pp. 73–81, 2004.

[43] T. Taylor, S. Brooks, J. Mchugh, and S. Brooks, “Netbytes viewer: An Entity-based Netflow Visualization Utility for Identifying Intrusive Behavior,” in *Proceedings of the 2007 Workshop on Visualization for Computer Security (VizSec)*, pp. 101–114, 2008.

[44] “Nessus 3D.”

[45] I. Xydas, G. Miaoulis, P. Bonnefoi, D. Plemenos, and D. Ghazanfarpour, “3D Graph Visualization Prototype System for Intrusion Detection: A Surveillance Aid to Security Analysts,” in *Proceedings of the 9th International Conference on Computer Graphics and Artificial Intelligence*, May 2006.

[46] A. Oline and D. Reiners, “Exploring Three-Dimensional Visualization for Intrusion Detection,” in *Proceedings of the IEEE Workshop on Visualization for Computer Security (VizSEC)*, pp. 113–120, Oct. 2005.

[47] J. Oberheide, M. Karir, and D. Blazakis, “VAST: Visualizing Autonomous System Topology,” in *Proceedings of the 3rd International Workshop on Visualization for Computer Security (VizSEC)*, pp. 71–80, 2006.

[48] G. Hubona, P. Wheeler, G. Shirah, and M. Brandt, “The Relative Contributions of Stereo, Lighting, and Background Scenes in Promoting 3D Depth Visualization,” *ACM Transactions on Computer-Human Interaction*, vol. 6, pp. 214–242, Sep. 1999.

[49] W. Harrop and G. Armitage, “Real-time Collaborative Network Monitoring and Control using 3D Game Engines for Representation and Interaction,” in *Proceedings of the 3rd International Workshop on Visualization for Computer Security (VizSEC)*, pp. 31–40, 2006.

[50] N. S. Team, “Front end 3d (fe3d).” http://map.gsfc.nasa.gov.

[51] Z. Jiawan, Y. Peng, L. Liangfu, and C. Lei, “NetViewer: A Visualization Tool for Network Security Events,” in *Proceedings of the International Conference on Networks Security, Wireless Communications and Trusted Computing (NSWCTC)*, vol. 1, pp. 434–437, April 2009.

[52] S. Krasser, G. Conti, J. Grizzard, J. Gribschaw, and H. Owen, “Real-time and Forensic Network Data Analysis using Animated and Coordinated Visualization,” in *Proceedings from the Sixth IEEE Workshop on Information Assurance (IAW)*, pp. 42–49, June 2005.

[53] C. Papadopoulos, C. Kyriakakis, A. Sawchuk, and X. He, “Cyberseer: 3D Audio-visual Immersion for Network Security and Management,” in *Proceedings of the ACM Workshop on Visualization and Data Mining for Computer Security (VizSEC/DMSEC)*, pp. 90–98, 2004.

[54] J. Johnston, J. Eloff, and L. Labuschagne, “Security and Human Computer Interfaces,” *Computers and Security*, vol. 22, no. 8, pp. 675 – 684, 2003.

[55] M. Mehrnejad, E. Toreini, and A. Bafghi, “Security Analyzing and Designing GUI with the Resources Model,” in *Information Security and Cryptology (ISCISC), 2011 8th International ISC Conference on*, pp. 29–36, 2011.

[56] W. T. Lo, W. K. Fung, Y. H. Liu, K. C. Hui, N. Xi, and Y. C. Wang, “Real-time Teleoperation via the Internet with 3D Stereoscopic Video Feedback,” in *Proceedings of the IEEE International Conference on Robotics and Automation*, Apr. 2004.

[57] W. Fung, W. Lo, W. Liu, and N. Xi, “A Case Study of 3D Stereoscopic vs. 2D Monoscopic Tele-reality in Real-time Dexterous Teleoperation,” in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, pp. 181–186, Aug. 2005.

[58] C. Ware and P. Mitchell, “Visualizing Graphs in Three Dimensions,” *ACM Transactions on Applied Perception*, vol. 5, Jan. 2008.

[59] J. Matejka, W. Li, T. Grossman, and G. Fitzmaurice, “Communitycommands: Command Recommendations for Software Applications,” in *Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology (UIST)*, pp. 193–202, 2009.

[60] D. M. Krum, O. Omoteso, W. Ribarsky, T. Starner, and L. F. Hodges, “Speech and Gesture Multimodal Control of a whole Earth 3D Visualization Environment,” in *Proceedings of the symposium on Data Visualisation 2002*, VISSYM ’02, pp. 195–200, 2002.

[61] R. Kosara, “Poster: Indirect Multi-Touch Interaction for Brushing in Parallel Coordinates.”

[62] J. Bryner, “Kinectasploit,” in *Defcon*, Defcon 19, 2011.

[63] D. Kennedy, J. O’Gorman, D. Kearns, and M. Aharoni, *Metasploit: The Penetration Tester’s Guide*. San Francisco, CA, USA: No Starch Press, 1st ed., 2011.

[64] L. Motion, “Leap Motion,” 2013.

[65] F. Ritter, . Freed, A., and O. Haskett, “Discovering User Information Needs: the Case of University Department Web Sites,” *Interactions - HCI and Higher Education*, vol. 12, pp. 19–27, Sept. 2005.

[66] D. Labbe and N. Martin, “Design of package artworks for pleasurable food experience by a user centric approach,” in *Proceedings of the 2011 Conference on Designing Pleasurable Products and Interfaces*, pp. 20:1–20:4, 2011.

[67] “QualysGuard.”

[68] J. Gadge and A. Patil, “Port Scan Detection,” in *Proceedings of the IEEE International Conference on Networks (ICON)*, pp. 1–6, Dec. 2008.

[69] S. Sanfilippo, “Hping2,” 2006.

[70] C. Correa, T. Crnovrsanin, and K.-L. Ma, “Visual Reasoning about Social Networks Using Centrality Sensitivity,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 1, pp. 106–120, 2012.

[71] I. Hsi and C. Potts, “Studying the evolution and enhancement of software features.,” in *Proceedings of the 1st International Visual Informatics Conference on Visual Informatics (IVIC)*, 2009.

[72] J. Rasmussen, K. Ehrlich, S. Ross, S. Kirk, D. Gruen, and J. Patterson, “Nimble Cybersecurity Incident Management through Visualization and Defensible Recommendations,” in *Proceedings of the Seventh International Symposium on Visualization for Cyber Security (VizSec)*, pp. 102–113, 2010.

[73] Y. Low, J. Gonzalez, A. Kyrola, D. Bickson, C. Guestrin, and J. Hellerstein, “GraphLab: A New Parallel Framework for Machine Learning,” in *Conference on Uncertainty in Artificial Intelligence (UAI)*, July 2010.

[74] FileZilla, “FileZilla.” http://filezilla-project.org, 2009.

[75] M. Livingston, J. Sebastian, Z. Ai, and J. Decker, “Performance Measurements for the Microsoft Kinect Skeleton,” *Proceedings of the IEEE Conference on Virtual Reality Conference*, vol. 0, pp. 119–120, 2012.

[76] D. Avrahami, “RGB-D: Techniques and Usages for Kinect Style Depth Cameras,” 2011.