Visualization Tools for Network Security

Antoinette E. Attipoe; Bowie State University; Bowie, MD USA
Jie Yan; Bowie State University; Bowie, MD, USA
Claude Turner; Norfolk State University; Norfolk, VA, USA
Dwight Richards; The College of Staten Island-CUNY; Staten Island, NY, USA

Abstract

Network security visualization tool plays an important role in the network security filed. It's considered the first line of defense because it provides security analysts with visualized network information we need to either prevent or investigate an attack, an intrusion, an anomalous activity and much more. In this paper, we briefly describe the 13 network visualization tools we surveyed and we outline their advantages and disadvantages. We employ qualitative coding as part of our research design or framework to extract some metrics from the list of advantages and disadvantages of the tools to help us design an evaluation methodology, which we plan to use to measure the effectiveness of the visualization tools through usability studies.

Introduction

Network visualization tools play an important role in network security. Their primary purpose is to assist network security analysts in detecting, stopping, and defending against current and future network attacks [8]. For example, network visualization tools can be used to monitor network traffic and analyze network data for anomalous patterns or to investigate to determine if a network security event such as a network intrusion or attack has occurred. These activities provide the network security analyst or network administrator with useful information that can assist them in performing their daily tasks – which is, defending their organization or company's computer networks.

Our research primarily started with our attempt to identify the popular network visualization tools currently being used in the cyber security research field and to determine the contributions we could make to the field. In order to identify the tools currently being used and to learn about the research being done in this field, we decided to conduct a survey of as many of the popular visualization tools by first gathering and reviewing current survey papers as well as journal articles and other publications on the tools. A good starting point was to visit the Visualization for Cyber Security (VizSec) website at www.secviz.org to find resources that report the work currently being done in this research area. We found the DAVIX (The Data Analysis and Visualization Linux) 2014 Live CD that comes with many network visualization tools preinstalled and distributed as a VMWare image on this website. The DAIX VMWare image is primarily downloaded, installed on a virtual machine and used for data analysis and visualization. The DAVIX Live CD can be downloaded from the Visualization for Cyber Security website (www.secviz.org).

Since there were many tools to choose from, we decided to begin our survey by choosing tools based on the availability of journal articles and publications that provide detailed information about the tools. This decision led us to gather publications for 13 network visualization tools which we identified on the DAVIX 2014 Live CD. The tools we chose include InetVis (Internet Visualization), NVisionIP, VisAlert, IDS Rainstorm, Rumint, NetGrok, TNV (Time-based Network Visualizer), Portvis, Afterglow, Graphviz, Picviz, Flowtag, and Treemap.

In order to achieve our second goal which was to contribute to this research field, we looked for opportunities for further research by paying attention to the issues we encountered during our survey. We noted that there was a lack of publications reporting empirical evaluation of the network visualization tools. There were very few publications that described through statistical analysis and evaluation, the effectiveness of the visualization tools. We also noticed that the user study data needed to conduct such evaluations was not readily available. So we extended our research to address this problem.

In this paper, we briefly describe the 13 network visualization tools we surveyed and we outline their advantages and disadvantages. We employ qualitative coding as part of our research design or framework to extract some metrics from the list of advantages and disadvantages of the tools to help us design an evaluation methodology, which we will use to measure the effectiveness of the visualization tools through usability studies. We conclude the paper and discuss our future work.

Related Work

There are many works that have focused on different aspects of network security visualization tools. Some works report a description of the tools as well as their features and provide examples of how the tools are used by giving network security event scenario and showing how the tool is used in that setting. Examples of these are [2, 3, 6, 25]. Other works also show a comparison of two or more tools, outlining their characteristics and providing examples of how they are used. For example [18]. Some work also briefly describes some of the network visualization tools as a precursor to their own work in the related work section of their paper. Most often, the goal of reviewing these tools is to make a case for their own, proposed visualization tool. The tools outlined in this section usually have some weaknesses that the proposed tool intends to address.

The work of Fligg and Max [8] for example, were focused on the design of network visualization tools. Fligg and Max [8] conduct an in-depth study on the design of network security visualization tools. Ferebee and Dasgupta [26] also reviewed some of the current trends used in security visualization.

While there are many excellent works available on network security visualization tools, there are very few works that report empirical evaluations on the effectiveness of the visualization tools. Thompson et al. [40] mention in their work that, while some of these network visualization tools have been designed using a user-centered approach, very few have been empirically evaluated in the task of intrusion detection. Though Thompson et al. [40] mentioned this in relation to intrusion detection, it is true of any other network security event.

When Shiravi et al [22] were selecting papers to study the different network visualization systems they surveyed, they

mentioned that, one of the metrics they used in selecting the papers was "the satisfaction of empirical evaluation" done on the visualization system in the paper they were reviewing. They stated that most of the visualization systems they surveyed in their paper lacked formal evaluation. This reveals the need for more publications that have evaluated the effectiveness of these network visualization tools. They however stated that, though these papers lacked formal evaluation, many of them had been validated through the use of use case attack scenarios. They also ensured that visualization systems that lacked even the basic validation strategy, of use case attack scenarios, were not included in their work. It is important to also add that the work of Shiravi et al. [22] is one of the most comprehensive surveys on network security visualization tools. As part of their work, they classify visualization systems or tools into five use-case classes. Their argument for using this approach was that, the methodology behind the design of visualization systems should be use case driven and not data source driven because visualization systems should be built to support answering specific questions. Multiple data sources can be incorporated into a visualization system if this approach is used. We use this classification of network visualization as a starting point in our qualitative data analysis.

McKenna et al. [38] also identified this problem (the lack of empirical evaluation measuring the effectiveness of network visualization tools) in their work. They stated that, the practice of user-centered design incorporates careful consideration of user's needs, wants, and limitations throughout the design process, which helps in evaluating both the effectiveness and appropriateness of tools [38, pp. 1]. However their survey of the proceedings on the Visualization for Cyber Security (VizSec) website showed that, about 40% of the 51 papers included evaluation with users, mirroring the findings of a recent survey looking back a full 10 years [38, pp. 1]. Of these papers, only 7 discussed iterative evaluation with users to improve the design of a tool, with the more common case being evaluation with users only after the design of a tool is complete. They noted the vast opportunity within the VizSec community to improve the efficacy of visualization tools by using evaluation and user-centered design methods throughout the entire design of a tool process, which includes gathering user needs, design opportunities, and ideas before building a tool. While our focus is not building a tool, we realize that, evaluating the effectiveness of the currently existing network visualization tools is still very important.

Goodall [39] also identified and noted this same problem in his work. In his work [38, pp. 1], Goodall noted that that, while the Visualization for Cyber Security (VizSec) has rapidly matured over the past several years and there are now many techniques and tools applying information visualization to the problems of cyber security, particularly in network traffic analysis and while the design of several of these tools are grounded in the tasks that real world users face, these tools are *rarely* tested empirically.

Based on our preliminary research we identified empirical evaluation of the network visualization tools as an important research area to address and subsequently focused our research in that area.

Network Visualization Tool Descriptions

InetVis (Internet Visualization)

InetVis is a visualization tool used to monitor the network traffic in animated, three-dimensional scatterplots. It represents network events as colored points in the animated 3D scatterplot [25]. Traffic is mapped into a cube, highlighting the specific patterns for particular anomalies [24]. It has several features worth noting. One of InetVis' features is its color map. Its' source port, destination port, packet size and protocols are colored differently. It also has a timer that animates the replay of capture files according to playback rate. The perspective projection conveys three dimensionality and depth to reflect a sense of spatial locality. The orthographic projection is useful for obtaining an accurate reflection of geometry and obtaining flat planar views along a particular axis. There is also filtering and user interface interaction such as zoom, move and rotate through a navigational control. InetVis uses packet captures for its visualization. [25, 27] describes how the packets are plotted. The packets are plotted by:

- Destination address (home or internal network range) are plotted along the blue x-axis (horizontal)
- Source address (external internet range) is plotted along the red z-axis (depth)
- Ports TCP and UDP are plotted along the green y-axis (vertical)
- ICMP traffic is plotted below TCP/UDP cube. It is a grey/white ICMP plane

InetVis is an effective tool for visualizing and analyzing internet traffic and port scanning events.

NVisionIP

NVisionIP is a security visualization tool that provides a view of the entire network (a Class B network). In other words, it is designed to increase the security analyst's situational awareness [35]. It follows the Visual Information Seeking Mantra: "Overview first, zoom and filter, then details-on-demand." It therefore allows users to drill down and gather more details about the hosts on the network [13]. It represents the state of all IP addresses within an IP address space using multilevel grid interface [21]. It also facilitates the understanding of the state of a network [2]. Lakkaraju et al. [13] describe the three views of NVisionIP and other features in their work. For example, the galaxy view feature of NVisionIP shows high level data (visual summary) about the entire network. The small multiple views give a reasonable amount of information on a user selected subnet of machines. And the machine view shows all (detailed) information for a single machine. The drill down and zoom allows a security engineer to choose a subnet of machines and view them in the Small Multiple View [13]. The difference view allows a user to compare log files by subtracting one from the other [21]. Standard zoom increases the size of the galaxy view underneath the zooming tool [13]. Coloring is used to differentiate the various machines. The color of each machine represents the number of unique ports used by that machine to send and receive data [13]. Sparkline shows context of how displayed values compare to recently past values to help determine if a value is within range or out-of-range as well as recent trends [21]. Shape enhancement, for example line, triangle and box shapes are used to enhance detection of different metrics (along with magnification) [21]. NvisionIP was extended to include the "close the loop" functionality by allowing users to create rules from the visualization that can then automatically alert on new data [2]. With filtering, user can filter or aggregate an interesting set of hosts based on any combination of IP addresses, ports or protocols.

IDS Rainstorm

IDS RainStorm is a visualization tool that is useful for visualizing IDS alarms on a large network, observing time patterns,

and knowing location (local and external IPs) severity [10]. IDS RainStorm has several features that are worth noting. Its main view visualization uses a set of rectangular regions that represent (top-to-bottom) the set of contiguous IP addresses, where 20 addresses are allocated to a row of pixels. For the zoom view, when a user clicks on the zoom overview, a secondary screen appears in a separate window with an enlarged view of the portion enclosed by the red box. The filtering feature ensures that in both the main overview and zoom views, IDS RainStorm allows the user to filter on alarm severity by choosing the show only the high critical alarms (red), medium concern alarms (yellow), or the low concern alarms (green). And glossing happens when a user moves the mouse cursor over an icon or particular text, and expanded information is presented. IDS RainStorm takes IDS alarm logs as its data source [10].

VisAlert

VisAlert is a visualization tool that correlates various network-based and host-based alerts from disparate IDS logs. It is based on the notion that an alert must possess three attributes, namely *What*, *When* and *Where* [20]. There are visual indicators used in the VisAlert tool; color coding for instance. Color is used to determine user selected ranges and severity levels. Icon size: large node size with larger number of different alert types. And alert beam: larger beam size for persistence of the same problem, Goodall et al. [12]. The data source for VisAlert visualization is IDS logs and alerts. Time is represented as a radial coordinate of a polar coordinate system. Resources are individual network nodes such as hosts, switches, and routers. And each node may contain additional information such as its name, IP address, mission(s), how critical it is to the organization, its operating system, the OS patch level, etc. [20].

Rumint

Rumint provides users with the ability to view large number of network packets in a way that supports rapid comparison, deep and broad semantic understanding, and highly efficient analysis [10]. There are 7 visualizations in the Rumint tool. These include the *thumbnail toolbar* that provides a real-time overview of each visualization window in a thumbnail size display. The s*crolling text display* presents network packets, one per horizontal row, in a user selectable encoding (ASCII, hexadecimal and decimal). The *parallel coordinate plot display* uses the parallel coordinate plot technique to display scaled values from packet header fields. The *detail display* displays the selected packet's content in a traditional hex/ASCII format. The *glyph-based animation display* combines three display panels to animate any two attributes (header fields). *Binary rainfall visualization* displays packet content, one per line. It has three primary views which map packet content to display pixels. The *scatterplot display* allows users to select any two header fields (19 are implemented) and plots them on a traditional X, Y display. *Byte frequency display* displays the presence and frequency of bytes within each packet. *PVR interface* allows the playback of packets that are captured live from the network or loaded from capture files and stored in an internal cache, in any of the visualization windows. The network component that is visualized are network packets and these serve as the data source to the Rumint tool as well [23]. Rumint uses raw packets as its data source.

NetGrok

NetGrok is a tool for visualizing computer network usage in real-time. It enables fast understanding of network traffic and easy problem detection. That is, it also allows for the viewing of network traffic at a glance and the discovery of phenomena such as network host scanning [23, 29]. Two types of visualization techniques are used in NetGrok visualization. These are network graphs and treemaps. Both of these visualizations capture IP hosts, the host's bandwidth usage, and links between hosts. The network graph aids in finding patterns n network traffic, and developing familiarity with a particular network. Treemap on the other hand complements the network graph in that, it is able to handle considerably more nodes, without occlusion, than the network graph. And they layout nodes using all of the available space [23]. NetGrok handles both real-time and static network packet data also. The features of NetGrok include the main visualization, that is, the overview. Then there is the timeline histogram, the zoom and filter functionalities, the details on demand feature and coloring feature [23].

TNV (Time-based Network Visualizer)

TNV is a visualization tool designed to facilitate the analysis processes related to intrusion detection by providing a focused view on packet-level data in the high-level network traffic context [6]. TNV provides at the high-level of aggregation, a visual overview of the entire data set [6]. TNV also provides a visual display that can facilitate the recognition of patterns and anomalies over time [2]. The data used in the TNV tool are raw network packets. Some of the TNV usage scenarios such as attack analysis, port scanning and learning the network are described in [6]. A description of the features of TNV outlined by Goodall et al. [6] is presented here.

The main visual component of TNV combines a matrix display of host IP address and network packet timestamp with a link display explicitly showing connectivity between hosts. TNV also displays network links between hosts within a single time period. There is also the data overview which gives the analyst a visual overview of the entire data set. A histogram of the relative network traffic activity of the entire dataset is also included in TNV. The focus + context interface approach was suggested and used in TNV because of the importance of context in intrusion detection (ID) analysis in TNV. The details on demand feature of TNV also allow items or group of items to be selected and their details viewed. Filtering of links and details on demand are also two types of user interface interaction functionality used in TNV [6]. In summary, by providing several linked views (overview histogram, main matrix and link visualization, port activity, textual packet details), the analyst retains the big picture while exploring the data set from multiple perspectives [6].

Portvis

Portvis analyzes high level summaries of packet data from a large network. Its primary focus is to detect large scale network events. And it provides multiple views of the same information to help correlate data and allow an operator to mentally shift between visualizations [18]. PortVis has several features. One is the timeline visualization. The timeline is a visualization of the entire time range available to the system, PortVis, from its data source [3]. The main (hour) visualization depicts the activity during a given time unit [14]. It consists of a 256 x 256 grid in which each dot represents one of the 65,536 ports [3]. And the port visualization is a view of all the activity and data available that concerns a particular port [3, 14]. The histogram corresponds to the relative frequencies of each data value. It serves to identify trends and/or patterns in the data. Last but not the least is the variance visualization, also called the variance analysis system. It's a feature that allows analysts to select any arbitrary set of time units and see on the main visualization not

depiction of the actual values at each port but rather a depiction of the variance of the values at each port [14]. PortVis uses network traffic as its data source.

Afterglow

Afterglow is a visualization tool that facilitates the generation of graphs [31]. It is a series of PERL scripts designed to be used with Graphviz to generate link graphs from Common Separated Values (CSV) formatted files [5]. Logfile data and tcpdump are its data source. AfterGlow is not a standalone tool. It must be used with Graphviz for example in order to generate data visualizations.

Graphviz

Graphviz is a visualization tool used for viewing and interacting with graph diagrams [32]. It generates a variety of graph layouts [5]. Graph visualization is a way of representing structural information as diagrams of abstract graphs and networks [32]. After being processed by Afterglow's conversion scripts or processes, the extracted CSV files from Afterglow is fed through scripts to produce a DOT attributed graph language file. This is the input required by the Graphviz library. There are also base utilities that generate radial layouts, spring model layouts and circular layouts. They all interpret files that have been described using the DOT language [5].

Flowtag

Flowtag is a system used to visualize network flows and to tag the data to support analysis and collaboration [2]. It enables quick analysis, reporting, and sharing of attack data [7]. According to [37], Flowtag operates on PCAP files and produces a database of flows. It then visualizes the results. The next step after the visualization of the results is for the user to filter for flows of interest, view the payload, and tag the flow with relevant keywords. Flowtag's interface has six components namely, flow table, flow tag, payload view, connection visualization, filters and tags list. Lee [37] describe them as follows: The flow table is a list of matching flows (source IP, destination IP, source port, destination port, and time). When a flow in this table is clicked on, the contents of the flow will be displayed in the payload view. Flow tags is a small entry box allows the user to associate keywords (tags) with the currently selected flow. When the user clicks on a flow in the flow table, the reconstructed payload of the currently selected flow is displayed in this text (payload view) box. The connection visualization is a canvas that displays a parallel coordinate plot with the left axis mapping the TCP ports (using a cube root scaling to emphasize the lower ports) and the right axis mapping the IP addresses in order of appearance in the network trace file. Filters allow the user to remove uninteresting flows based on time, the number of packets in the flow, or the number of bytes in the flow. The time slider is a double-ended linear slider and the packets and bytes sliders are double-ended logarithmic sliders. Tags list is a selector that lists all the defined tags and allows the user to filter for flow matching the selected tag.

Picviz

Picviz is a software for transforming the acquired data into a parallel coordinates plot image to visualize the data and discover interesting results quickly [16]. Tricaud [16] also discusses the features of Picviz and we describe them next.

Picviz has a graphical frontend that provides a skillful interaction to find relationship among variables, allows to apply filters, drag the mouse over the lines to see the information displayed and to see the time progression of plotted events. The grand tour generates as much images as pairs permutation of axes possible. And the idea is to show every possible relation among every available axes. The filtering feature allows Picviz to use filters to select lines that one wants to display. The filters can be used on the real value to match a given regular expression, line frequency, line color or position as mapped on the axis. It's a multi-criterion filter. Picviz has a command line interface as well.

Treemap

Treemap is a space-constrained visualization of hierarchical structures [30]. It's a visualization tool that uses 100% of the available display space, mapping two attribute of the data into the size and color of nested rectangular regions [9]. It is very effective in showing attributes of leaf nodes using size and color coding. Treemap enables users to compare nodes and sub-trees even at varying depth in the tree, and help them spot patterns and exceptions [30]. It provides a rapid overview of the relative size of nodes [9]. Treemap controls include the main tab which allows users to select any one of the three layout algorithms, that is, squarified, slice and dice, and strip, depending on their need as well as font size and border options. The legend tab allows users to assign attributes to be used for label, size and color options. And the filter tab allows users to filter data using dynamic query sliders [9].

Table 1: Summary of Network Visualization Tool Descriptions

[Table 1 goes here]

Table 2: Use Case Categorization of Network Visualization Tools

[Table 2 goes here]

Advantages and Disadvantages of Visualization Tools

For this research, it was important for us to identify and outline the advantages and disadvantages of the network visualization tools as this would help us identify some metrics to help us measure the effectiveness of the tools. A summary of the visualization tools are presented in Tables 3 through 6. It is important to note that we categorized the tools we surveyed according to the use case classes identified by Shiravi et al. [22].

Table 3: Table 3 - Advantages and Disadvantages of Tools Host-Server Monitoring Category

[Table 3 goes here]

Table 4 - Advantages and Disadvantages of Tools in Internal-External Monitoring Category

[Table 4 goes here]

Table 5 - Advantages and Disadvantages of Tools in Port Activity Category

[Table 5 goes here]

Table 6 - Advantages and Disadvantages of Tools in Attack Pattern Category

[Table 6 goes here]

Research Framework

Evaluation Design and Setup

A starting point to help us evaluate the effectiveness of the network visualization tools was to design a research framework to assist us in achieving our goal. We reasoned that a good place to start identifying metrics was to review the tool features as well as review the advantages and disadvantages of the tools.

Qualitative Coding

We used qualitative data analysis, specifically, qualitative coding to help us identify some initial metrics which we could use to measure the effectiveness of the network visualization tools. Qualitative coding is the process of organizing and sorting your qualitative data using codes which could be a word, phrase, number or symbol. To build our storyline to identify potential codes which could later be refined and used as our preliminary metrics, we asked two questions:

(i) What are we trying to find out?
(ii) What do we want to convey with the information we find out?

And our storyline simply reads:
"We would like to measure the effectiveness of the visualization tools. By effectiveness, we mean, we would like find out if they are really doing what they were designed to do; such as detect an intrusion, detect anomalous patterns in network traffic or effectively monitor and visualize network traffic."

As we attempted to answer these questions in light of measuring the effectiveness of the visualization tools, we noted the following were important to us:

(i) We would like the outcome of our evaluation to show that some network visualization tools may be effective in addressing certain types of security incidents or events better than other visualization tools
(ii) We also want to know the limitations of the tools. At a minimum, the display of each visualization tool should address and satisfy
    a. The visual information seeking mantra (over first, zoom and filter, details-on-demand)
    b. The three major questions that a network security tool must answer:
        - *Where* in the network is the attack happening?
        - *When* is the attack happening?
        - *What* type of attack is happening?
(iii) We would like to know what makes the tool unique over other tools.

Answering the above questions will provide us with a wealth of information to assist us with the qualitative coding and to help us identify the metrics we needed to measure the tool's effectiveness.

Identifying Codes

The list of advantages and disadvantages we outline for each tool was instrumental in helping us identifying codes for our qualitative data analysis. In qualitative coding, there are pre-set codes and emergent codes. Preset codes are a "start list" of codes. These initial codes can be derive from the conceptual framework, list of research questions, problem areas, etc. [42]. One source of pre-sets codes we gathered was from the work of Shiravi et al. [22] on network security visualization systems or tools. In their work, they identified five use case classes which were host/server monitoring, internal/external monitoring, port activity, attack patterns and routing behaviors. Our tools however fell within the first four categories. Table 7 shows our initial pre-set codes.

There were other pre-set codes that we knew we could be measured based on research area and tool design concepts such as processing speed of the visualization tools. Table 8 shows the list of codes we identified.

Table 7: Initial pre-set codes adopted from the work of Shiravi et al. [22]

[Table 7 goes here]

Table 8: Second set of pre-set codes identified through research area and tool design concepts

[Table 8 goes here]

Identifying Metrics

Next, we applied open coding, which is the first level of coding to analyze our data or information (list of advantages and disadvantages). At this first level, we were looking for distinct concepts and categories in their data. As we read through our list, we identified and extracted important keys words and phrases explaining the tool's advantages and/or disadvantage and we wrote down. Once we identified our first list of codes, we applied axial coding, which is where we used our concepts and categories while re-reading our list of advantages and disadvantages again. And we recognized some emergent codes. Emergent codes are those ideas, concepts, actions, relationships, meanings, etc. that come up in the data and are different than the pre-set codes [42]. See Table 9 for the final, derived metrics.

Table 9: Final metrics derived from pre-set and emergent codes using open and axial coding

[Table 9 goes here]

User Study Design

Setup

As part of another project we are working on, we are developing instructional laboratory guides for each of the tools we surveyed. The need to develop these instructional guides arose when we realized there were hardly any readily available materials that document a step-by-step procedure to help students learn these network visualization tools we surveyed. Thankfully, there is a DAVIX manual (version 0.5.0) manual that can be downloaded from the Visualization for Cyber Security website. This manual guides a user to get started with each of the network visualization tools installed on the DAVIX CD. It outlines a quick setup guide and for each visualization tool, it outlines (1) the *purpose* of the tool, (2) helpful *links* to get more information, (3) *important*

installation locations on the Linux operating systems, and (4) an *example* of how to load and visualization network data. While this is a good start, we realize that there is, one, a need for more detailed, real-world examples that can help students learn important computer and network security concepts while learning to use these tools. And two, there is also the need for a visualization tool manual or instructional guide that a novice, intermediate or expert level security student can all benefit from. With this idea in mind, we set out to develop a framework to create an instructional laboratory guide for each of the tools which will address the points stated above. This framework is outlined in another work.

Instructional Laboratory Guides

We plan to conduct user studies after we have developed an instructional laboratory guide for each tool. Our evaluation will cover both the effectiveness and efficiency of the laboratory guide as well as measuring the effectiveness of the visualization tools. Details of evaluation for the instructional laboratory guide is covered in another work. The instructional laboratory guide for each tool will include

(i) An introduction explaining the purpose of the tool, a description of how the tool works, and any other important information about the tool such as its features (visualization techniques, data sources, etc.).

(ii) A list of all the materials required and provided to complete the hands-on exercises.

(iii) A minimum of two examples (hands-on exercises) to be performed; a general hands-on exercise and a detailed hands-on exercise. The general hands-on exercise will help students become familiar with the tool environment and graphical user interface. This exercise will also include important tasks such as showing students how to load and visualize their data. The DAVIX manual will be helpful in developing this section of the instructional guides. The in-depth hands-on exercises will aim at introducing students to important computer and network security concepts while they learn to use the visualization tool. These exercises will introduce a network security event or scenario and will require students to apply the visualization tool to analyze and investigate the security event. We plan to include visual elements such as pictures, screenshots, and diagrams in this section to help students understand these tasks and grasp these important security concepts.

(iv) Additional exercises for students who are interested in exploring and getting some more hands-on practice on using the visualization tools. These exercises will be more exploratory than the well-defined, two, hands-on exercises explained above.

(v) Other helpful information about the tools such as helpful website links and a glossary of important terms. Details about the development of the instructional laboratory guide are outlined in another work.

These instructional guides will serve as the vehicle to conduct user studies and gather data to evaluate the effectiveness of each visualization tool. Once the instructional guide has been developed, usability studies will require students to follow the steps in the laboratory guide to learn how to use the visualization tools. The laboratory guide will be improved over time based on user feedback through surveys and observations. We will also gather user feedback through surveys and observation about the effectiveness of the visualization tools.

Participants

We plan to first gather user study data about the effectiveness of the visualization tools from students who have some basic knowledge about network security. So we will be employing understand and graduate students who have taken a computer networking course. The computer science department at our university has a rich pool of undergraduate students who are majoring in computer technology and are required to take a computer networking class. These will be good candidates for our user studies.

Graduate students who are following the computer/network or cybersecurity track will also be good candidates for our user studies.

User Study

We will conduct the user studies in the dedicated cybersecurity laboratory in the computer science department at our university. The laboratory will be setup with the visualization tools installed on the computers. Participants will be provided with consent forms to complete and provided with the instructional laboratory guide. Students will also be given an introduction of the study and answer any questions they might have. At this time, the user study setup is still be developed and a detailed report of the user study setup will be provided in another work.

This focus of this user study however is to generate the data we need to conduct our evaluation of the network visualization tools.

Evaluation

Performing the Evaluation

To gather the information we need to conduct our evaluation of the visualization tools, we plan to use surveys and observations. The questionnaires for the surveys will be structured to generate both quantitative and qualitative data. The metrics we identified from the advantages, disadvantages and features of the visualization tools will aid us in developing the questionnaires.

Once we have gathered the information we need, we plan to conduct evaluations similar to the work of Goodall [39] and Thompson et al. [40]. Using the metrics we identified will be a starting point to measuring the effectiveness of the tools. The information we gather through this preliminary evaluation will enable us to conduct further statistical analysis to measure other metrics such as *accuracy* (the ability of the tools to identify attacks; that is, measuring and rating tool performance) [39], *efficiency* (how well the tool performs a task and how fast the tool completes a task) [39], *user perfection* (positive versus negative user perception or experience) [39], *confidence or user confidence level* [40].

We also plan to gather some qualitative data which will primarily be in the form of user feedback or problems they encountered that our questionnaires did not address.

Reporting Results

We plan to report our results on an on-going basis; that is, we will report the results of our evaluation for the first visualization tool we create an instructional manual for. Once we create a second laboratory guide, we will conduct another user study to generate data to evaluate the tool and also to conduct a comparative evaluation of these first, two tools.

We plan to build on this work to conduct more empirical evaluations as we develop instructional guides for more of the network visualization tools we surveyed.

Primary studies usually report information such as mean and standard deviation for subjects. We consider the user studies we conduct as primary studies which will assist us in generating and reporting interesting evaluations of the visualization tools through statistical analysis.

With time, we also plan to take our work a step further to conduct a meta-analysis (with hypothesis testing) of the network visualization tools we surveyed in our work as well as those for which we can find published statistical data, such as the work of Goodall [39] and Thompson et al. [40]. A meta-analysis takes "published" statistical results and performs a post analysis.

Conclusion and Future Work

In this paper we surveyed and briefly described 13 network security visualization tools. We outlined their advantages and disadvantages and used this information the basis to identify some metrics which we plan to use to conduct preliminary evaluations of the tool's effectiveness. We plan to extend the evaluation further to measure other metrics such as the accuracy and efficiency of the tools. We also plan to measure user perception and user confidence levels through our evaluations.

This is an on-going work and we plan to report our evaluation results as we complete each phase of the project. Sections of the research framework which are still being developed such as the detailed, structure for the instructional laboratory guides and the user study setup will be reported once the development is complete.

References

[1] T. Zhang, Q. Liao, and L. Shi, "Bridging the Gap of Network Management and Anomaly Detection through Interactive Visualization," *2014 IEEE Pacific Vis. Symp.*, pp. 253–257, Mar. 2014.

[2] J. R. Goodall, "Introduction to Visualization for Computer Security," in *Proc. Work. Vis. Comput. Secur. - VizSEC '07*, J. Goodall, G. Conti and K. Ma, Eds. Berlin: Springer, pp. 1-17, 2007.

[3] A. Shabtai, D. Klimov, Y. Shahar, and Y. Elovici, "An Intelligent, Interactive Tool for Exploration and Visualization of Time-Oriented Security Data," *Proc. 3rd Int. Work. Vis. Comput. Secur. - VizSEC '06*, p. 15, 2006.

[4] R. Lengler and M. J. Eppler, "Towards a Periodic Table of Visualization Methods for Management," *2007 IASTED Int. Conf. Graph. Vis. Eng. GVE 2007*, pp. 83–88, 2007.

[5] C. Valli, "Visualisation of Honeypot Data Using Graphviz and Afterglow," *J. Digit. Forensics, Secur. Law*, vol. 4, no. 2, pp. 27–38, 2009.

[6] J. R. Goodall, W. G. Lutters, P. Rheingans, and a. Komlodi, "Preserving the Big Picture: Visual Network Traffic Analysis with TNV," *IEEE Work. Vis. Comput. Secur. 2005. (VizSEC 05),* pp. 47–54.

[7] C. Lee and J. Copeland, "Flowtag: A Collaborative Attack-Analysis, Reporting, and Sharing Tool for Security Researchers," *Proc. 3rd Int. Work. Vis. Comput. Secur.,* pp. 103–107, 2006.

[8] K. Fligg and G. Max, "Network Security Visualization," *IEEE Network Special Issue on Recent Dev. Network Intrusion Detection*, pp. 1–12, 2012.

[9] G. Chintalapani, C. Plaisant, and B. Shneiderman, "Extending the Utility of Treemaps with Flexible Hierarchy," *Proc. 8th Int. Conf. Inf. Vis. 2004 (IV 2004),* pp. 1–10, 2004.

[10] G. Conti, K. Abdullah, J. Grizzard, J. Stasko, J. a. Copeland, M. Ahamad, H. L. Owen, and C. Lee, "Countering Security Information Overload through Alert and Packet Visualization," *IEEE Comput. Graph. Appl.*, vol. 26, no. 2, pp. 60–70, 2006.

[11] A. Cockburn, A. Karlson, and B. B. Bederson, "A Review of Overview + Detail, Zooming, and Focus + Context Interfaces," *ACM Comput. Surv.*, pp. 1–42, 2008.

[12] J. Goodall, G. Conti, and K. L. Ma, "Mathematics and Visualization," in *Proc. Work. Vis. Comput. Secur. VizSEC*, 2007.

[13] K. Lakkaraju, W. Yurcik, R. Bearavolu, and A. J. Lee, "NVisionIP : An Interactive Network Flow Visualization Tool for Security," *IEEE Int. Conf. Syst. Man Cybern.*, pp. 2675–2680, 2004.

[14] J. McPherson, K.-L. Ma, P. Krystosk, T. Bartoletti, and M. Christensen, "PortVis: A Tool for Port-Based Detection of Security Events," *Proc. Int. Symp. Vis. Cyber Secur. - VizSec*, p. 73, 2004.

[15] B. Shneiderman, "The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations," in *Proc. IEEE Symp. Visual Languages*, 1996, pp. 336–343.

[16] S. Tricaud, "Picviz: Finding a Needle in a Haystack," *WASL'08 Proc. First USENIX Conf. Anal. Syst. logs*, pp. 3–3, 2008.

[17] S. Tricaud, K. Nance, and P. Saade, "Visualizing Network Activity Using Parallel Coordinates," *2011 44th Hawaii Int. Conf. Syst. Sci.*, pp. 1–8, 2011.

[18] C. P. Lee, J. Trost, N. Gibbs, R. Beyah, and J. A. Copeland, "Visual Firewall : Real-time Network Security Monitor Workshop on Visualization for Computer Security," pp. 129–136, 2005.

[19] Z. Jiawan, Y. Peng, L. Liangfu, and C. Lei, "NetViewer: A Visualization Tool for Network Security Events," *2009 Int. Conf. Networks Secur. Wirel. Commun. Trust. Comput.*, vol. 1, pp. 3–6, 2009.

[20] Y. Livnat, J. Agutter, S. Moon, R. F. Erbacher, and S. Foresti, "A visualization paradigm for network intrusion detection," *Proc. from 6th Annu. IEEE Syst. Man Cybern. Inf. Assur. Work. SMC 2005*, vol. 2005, no. June, pp. 92–99, 2005.

[21] W. Yurcik, "Tool Update : NVisionIP Improvements (Difference View, Sparklines, and Shapes)," *Hum. Factors, VizSEC 2006*, no. C, pp. 65–66, 2006.

[22] H. Shiravi, A. Shiravi, and A. a. Ghorbani, "A Survey of Visualization Systems for Network Security," *IEEE Trans. Vis. Comput. Graph.*, vol. 18, no. 8, pp. 1313–1329, 2012.

[23] R. Blue, C. Dunne, A. Fuchs, K. King, and A. Schulman, "Visualizing real-time network resource usage," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 5210 LNCS, pp. 119–135, 2008.

[24] R. Fontugne, T. Hirotsu, and K. Fukuda, "A Visualization Tool for Exploring Multi-scale Network Traffic Anomalies," *J. Networks*, vol. 6, no. 4, pp. 577–586, 2011.

[25] J.-P. van Riel and B. Irwin, "InetVis, a Visual Tool for Network Telescope Traffic analysis," *Proc. 4th Int. Conf. Comput. Graph. Virtual Reality, Vis. Interact. Africa - Afrigaph '06*, pp. 85–89,

2006.

[26] D. Ferebee and D. Dasgupta, "Security visualization survey," *Proc. 12th Colloq. Inf. Syst. Secur. Edu,* pp. 119–126, 2008.

[27] R. McRee, "Security Visualization: What You Don't See Can Hurt You," *ISSA J.*, pp. 38–41, 2008.

[28] R. McRee, "Tools for visualizing IDS output, PICTURES," *Linux Magazine*, no. 106, 2009.

[29] Cs.umd.edu, "NetGrok," 2015. [Online]. Available: http://www.cs.umd.edu/projects/netgrok.

[30] Cs.umd.edu, "Treemap: Home page," 2015. [Online]. Available: http://www.cs.umd.edu/hcil/treemap.

[31] R. Marty, "AfterGlow," 2015. [Online]. Available: http://afterglow.sourceforge.net.

[32] Research.att.com, "AT&T Labs Research - Software Tools," 2015. [Online]. Available:

http://www.research.att.com/software_tools?fbid=hUkKrKdo2Vj.

[33] Visual-literacy.org, "A Periodic Table of Visualization Methods," 2015. [Online]. Available: http://www.visual-literacy.org/periodic_table/periodic_table.html.

[34] S. K. Card, J. D. Mackinlay, and B. Shneiderman, *Readings in Information Visualization: Using Vision to Think*, Illustrate. San Francisco: Morgan Kaufmann, 1999.

[35] K. Lakkaraju, E. S. Ave, and A. J. Lee, "NVisionIP : NetFlow Visualizations of System State for Security Situational Awareness," *Proc. 2004 ACM Work. Vis. Data Min. Comput. Secur.*, vol. 29, pp. 65–72, 2004.

[36] K. Abdullah, C. Lee, G. Conti, J. a. Copeland, and J. Stasko, "IDS RainStorm: Visualizing IDS alarms," *IEEE Work. Vis. Comput. Secur. 2005, VizSEC 05, Proc.*, pp. 1–10, 2005.

[37] D. Lee, "FlowTag," *Chrislee.dhs.org*, 2015. [Online]. Available: http://chrislee.dhs.org/projects/flowtag.html.

[38] S. Mckenna, D. Staheli and M. Meyer, "Unlocking User-Centered Design Methods for Building Cyber Security Visualizations," *2015 IEEE Symp. Vis. Cyber Secur. (VizSec)*, 2015.

[39] J. Goodall, "Visualization is better! A Comparative Evaluation," *2009 6th Int. Work. Vis. Cyber Secur.*, 2009.

[40] R. Thompson, E. Rantanen, W. Yurcik and B. Bailey, "Command Line or Pretty Lines?" *Proc. SIGCHI Conf. Hum. Fact. Comput. Sys. - CHI '07*, 2007.

[41] A. C. Del Re, "A Practical Tutorial on Conducting Meta-Analysis in R," *The Quantitative Methods for Psychology,* vol. 11, no. 1, pp. 37-50, 2015.

[42] Tobaccoeval.ucdavis.edu, "Tobacco Control Evaluation Center," 2015. [Online]. Available: http://tobaccoeval.ucdavis.edu/index.html.

Funding

This work was supported by the National Science Foundation: NSF LUCID Grant [1303424]

Author Biography

Antoinette Attipoe has a Master's degree in Management Information Systems and in Computer Science as well from Bowie State University. She is currently pursuing a doctorate in Computer Science at the same school. Her current research focus is on network security visualization tools. She is also extending her research to address the importance and use of network visualization tools in security education as well as measuring the effectiveness of the network visualization tools.

Jie Yan (Member, IEEE) received her Ph.D degree in Computer Science at Harbin Institute of Technology, Harbin, China in 1999. She is currently an Associate Professor at Bowie State University. Her research interests include multi-view human face detection, tracking and recognition; realistic human face and body animation; and human facial expression, lip motion, body language synthesis, and virtual reality techniques. Dr. Yan is also a member of ACM and a full member of Sigma X.

Claude Turner is an Associate Professor in the Department of Computer Science at Norfolk State University, Norfolk.. His current research interest lie in the areas of computer and network security, cybersecurity education, biometrics, and digital and network forensics.

Dwight Richards is an Associate Professor in the Department of Engineering Science and Physics at the College of Staten Island (CSI), CUNY. He has developed system simulation models for the analysis of plastic optical fiber (POF) systems in commercial airplanes. More recently, he has been involved in cybersecurity research. His research interests are in the areas of optical network simulations, POF communication systems, computer and network security, and cybersecurity education.