Preserving the Big Picture: Visual Network Traffic Analysis with TNV

John R. Goodall Wayne G. Lutters Penny Rheingans Anita Komlodi

University of Maryland, Baltimore County

ABSTRACT

When performing packet-level analysis in intrusion detection, analysts often lose sight of the "big picture" while examining these low-level details. In order to prevent this loss of context and augment the available tools for intrusion detection analysis tasks, we developed an information visualization tool, the Time-based Network traffic Visualizer (TNV). TNV is grounded in an understanding of the work practices of intrusion detection analysts, particularly foregrounding the overarching importance of context and time in the process of intrusion detection analysis. The main visual component of TNV is a matrix showing network activity of hosts over time, with connections between hosts superimposed on the matrix, complemented by multiple, linked views showing port activity and the details of the raw packets. Providing low-level textual data in the context of a high-level, aggregated graphical display enables analysts to examine packet-level details within the larger context of activity. This combination has the potential to facilitate the intrusion detection analysis tasks and help novice analysts learn what constitutes "normal" on a particular network.

CR Categories: H.5.2 [Information Interfaces And Presentation]: User Interfaces—Graphical User Interfaces (GUI); C.2.0 [Computer-Communication Networks]: General—Security and Protection

Keywords: Network visualization, network analysis, information visualization, intrusion detection

1 INTRODUCTION

With the size and complexity of networks continuously increasing, security analysts face mounting challenges of securing and monitoring their computing infrastructure for attacks. This task is generally aided by an Intrusion Detection System (IDS), which attempts to automatically identify successful and unsuccessful attacks or abuse of computer systems [21]. Automated IDSs are a useful starting point for uncovering security compromises, but they are just that: a starting point. Analysts must then dig deeper into supplemental data sources to determine the accuracy and severity of an IDS alert. This usually includes the arduous task of collecting and identifying the relevant details of network traffic related to the event being investigated.

In addition to an IDS alert, the initial trigger event can come from other data sources as well. However, other sources of security related events often do not include the same level of detail as IDS alerts. These more ambiguous sources could come from, for example, a network monitoring system showing unusual spikes in traffic or users complaining to the help desk that the network seems slow. These types of vague starting points make further investigation problematic using current network analysis tools. From an IDS alert, an analyst can usually pinpoint the time of the event and the hosts involved in a potential attack.

Whether the starting point of analysis is data rich, such as an IDS alert, or data impoverished, such as a phone call from a user, analysis of a network security event is a complex task. The tools that analysts currently use, such as Tcpdump [4] or Ethereal [1], focus on extracting the details of individual packets, but lack a mechanism for providing a simultaneous "big picture" view of the data. This places the burden of putting individual packet details into a larger context of surrounding network activity on the analyst. Additionally, these kinds of tools excel at filtering and searching for details, but the analyst needs to know exactly what they are looking for in the data; they are not as useful for less structured data exploration.

To provide intrusion detection (ID) analysts with more complete support of the entire process of ID, we developed the Time-based Network traffic Visualizer (TNV). The design of this visualization is grounded in the work practices of ID analysts. Because of potentially ambiguous security trigger events, TNV emphasizes the temporal aspects of the data, which often serves as the starting point for analysis. The timeline of an event and putting an event into the larger context of activity were found to be crucial in ID analysis tasks. Thus, TNV preserves context by presenting a big picture view of the data linked to other visual and textual views of the data. This allows analysts to explore network traffic details at a simultaneous macro- and micro-levels.

This paper is organized as follows: section 2 outlines related research in the area of visualization for network security and analysis, section 3 presents TNV's design and interaction mechanisms, section 4 describes some typical scenarios of how TNV can facilitate analysts' decision making, section 5 presents our future work, and section 6 presents our conclusions.

2 RELATED RESEARCH

Many information visualizations of network data have used a link and node graph-based techniques to show communication patterns between nodes. Some visualizations of network data have placed nodes according to their geography (e.g., [6, 23]), while others clustered nodes according to their similarity (e.g., [8]). As one example, SeeNet uses a graph visualization that places nodes according to their natural geographic location and uses thickness and color to encode network statistics to provide a high-level view of network traffic [6]. These kinds of graph-based visualizations are useful in mapping networks and usage patterns by explicitly showing links between nodes, but can have problems with scalability, display clutter, and occlusion.

Researchers have recently begun applying information visualization to the particular problem of network security. Erbacher and colleagues have developed an animated glyph-based visualizations that use system log files to show connections from external hosts to a monitored server or small network environment [9, 10]. Several systems have adapted parallel coordinates of intrusion detection related data to: fingerprint network attack tools

email: jgood@umbc.edu
email: lutters@umbc.edu
email: rheingan@cs.umbc.edu
email: komlodi@umbc.edu

Workshop on Visualization for Computer Security
October 26, Minneapolis, MN, USA
0-7803-9477-1/05/$20.00 ©2005 IEEE.

[7], visualize log files [11], support new IDS event triage [25], and facilitate situational awareness [26]. Also in support of providing analysts with better situational awareness is NVisionIP, which visualizes NetFlow data in a scatterplot-based system with multiple levels of granularity for drilling down into the data visually [19]. PortVis takes summary network data and visualizes port activity as a scatterplot linked to several other views of the data [22].

In attempting to understand the challenges of network security work and how information visualization can be successfully used to facilitate this work, there have been several user studies of security analysts. Research at the National Center for Supercomputing Applications described the importance of providing analysts with situational awareness, derived from interviews with security analysts, and presents two information visualization tools to support this area of need [27]. Research at IBM also underlines the importance of situational awareness, particularly in relation to the difficult problem of new security event triage [25]. Ball, Fink, and North [5] noted the varied job types associated with system administration work and describe administrators' foremost interest in activity on their own network related to the machines they manage.

In our own research to understand the work practices of ID analysts, we interviewed analysts working in diverse job roles and organizations [12]. One important outcome from that research is the development of a basic task model of ID work. The work of ID includes three main tasks: *monitoring*, *analysis*, and *response*. The *monitoring* task is typically focused on the surveillance of the output of an IDS, involving the need for situational awareness. *Analysis* focuses on determining the accuracy and severity of a security event uncovered in the monitoring task. This is the most complex task, requiring a great deal of knowledge and experience to accomplish successfully. *Response* refers to an analyst's reaction to a security event. Both the monitoring and analysis tasks could be aided through information visualization tools. However, most of the visualizations for security to date seem to be targeted more towards facilitating monitoring than analysis.

All of the systems described above seem to be designed to support what we broadly refer to as the monitoring task. The reason for this classification is principally due to the level of detail available to these systems, which typically use aggregated or summary data, or system logs that do not have the raw packet data available for deeper analysis. These systems can alert analysts to anomalous activity on their networks or systems, augment the monitoring tasks, or increase situational awareness, but they are not likely to support the more detailed packet-level analysis that is necessary in the analysis of network security events. To provide analysts with more complete support beyond monitoring, we designed a visualization that focuses on the analysis task. One of the primary implications of designing for analysis is that the exact details of network traffic must be readily available. The tool presented in this paper places these details into the larger context of all network activity, allowing analysts to explore the details of an individual packet without sacrificing the contextual information needed to make decisions on the accuracy and severity of security events.

3 TNV: TIME-BASED NETWORK TRAFFIC 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. While it was designed specifically for the needs of ID analysts, TNV is expected to also be useful in network troubleshooting and aiding novice administrators in learning the idiosyncrasies of their network environments.

TNV is implemented in Java using the Sourceforge jpcap library [2]. This library utilizes libpcap, a widely deployed standard for network packet capture. TNV is capable of capturing packets in real time or opening previously generated libpcap files.

3.1 TNV Design Rationale

Our prior research identified some of the largest problems experienced by analysts, including data inundation and an inability to keep sight of the big picture when doing low-level analysis. The problem of data overload related to ID is well-known and there are many examples of these problems in the literature (e.g., [17, 18]). This pressing problem is one of the reasons that information visualization—which can make data more compact and understandable—presents such an appealing solution to the challenges of ID.

The need for retaining context when performing analysis was a recurring theme in our requirements gathering and forms the basis of the design of TNV. Analysts are rarely able to make a decision about a security event (such as an IDS alert) based solely on the data available from that artifact. Instead, analysts must build up a more complete picture of the event being investigated by reconstructing the event's timeline, the root cause of the event, and any related outcomes. These are what form the context surrounding an event, and of particular importance is the temporal context that helps reconstruct a timeline. For example, an IDS alert describing a potential web server attack will lead the analyst to attempt to decipher how the attack was carried out, if it was successful, if any other web servers might have been affected, and if the attacker may have attempted other exploits. Reconstructing the contextual details surrounding an event is crucial in formulating the proper diagnosis of a security event. However, the tools that analysts currently use do not fully support the discovery and comprehension of this important contextual data. Instead, they facilitate directed queries of low-level details through sophisticated searching and filtering mechanisms, but this presupposes that the analyst knows what to search for and does not help to see the big picture. Analysts repeatedly discussed how they would lose context when examining the details of packets and leaving the displays with the high-level contextual data. To overcome this lack of synchronization, analysts had to rely on their short-term memory to integrate low-level and high-level data without external representational support. Because of this, the design of TNV focuses on making the contextual data surrounding an event explicit and available on one screen; providing a big picture view, even when doing packet-level analysis.

In addition to providing analysts with context for the details they are investigating, time is crucial in analysis for several reasons:

- All of the data sources and tools used by our participants generate a timestamp, which despite being generated on different hosts correspond nearly exactly (all participants use Network Time Protocol on their systems). Because the security event triggering analysis may originate from any number of sources, the constancy of time across different sources allows analysts to synchronize different data elements from different sources.
- While the security event trigger may be an IDS or other monitoring system, it could also originate from a more ambiguous source, such as user feedback. This kind of vague trigger event often makes beginning the analysis task from anything other than time problematic. Time is available not only to all systems, but also to people.
- Events that occur before or after a trigger event can give the analyst vital clues about the nature of the event. As a

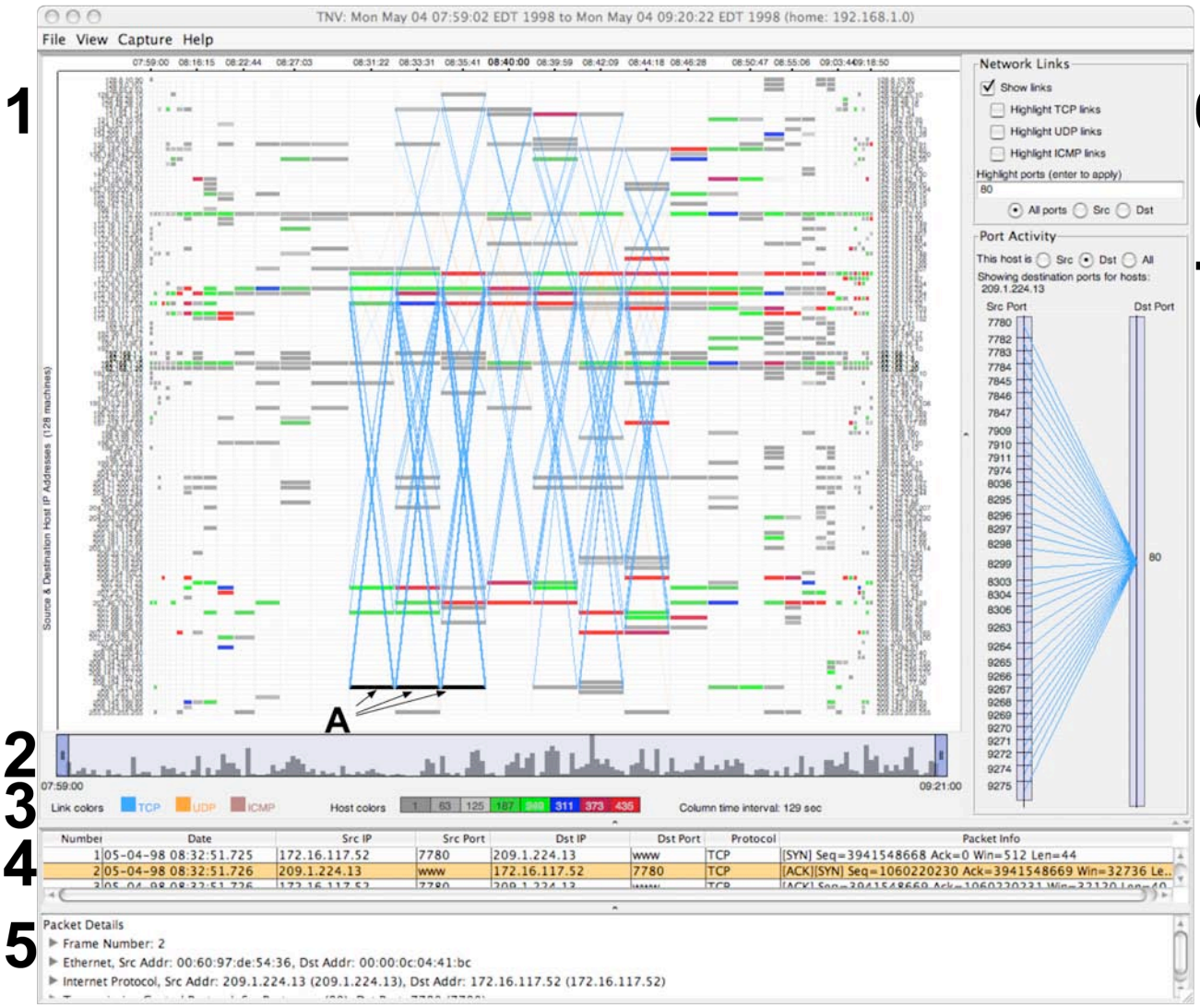

Figure 1. TNV showing 50,000 network packets in almost one and one half hours; network links with web (TCP port 80) activity are highlighted and the details of a selected host (***A***) are shown at right (port activity) and below (packet details) the main display. (***1***) is the main visualization matrix; (***2***) is the navigation with data overview mechanism; (***3***) shows the legend of colors and column time interval; (***4***) is the table of all packets for selected host; (***5***) shows packet details for selected row in table; (***6***) is the emphasis filtering panel; (***7***) shows port activity.

straightforward example, if immediately prior to the event being investigated, every host on the network was port scanned from a single destination, this could indicate that an attacker doing reconnaissance identified and exploited a vulnerability.

For these reasons, TNV emphasizes the temporal aspects of network data and permits the insights learned from using TNV to be easily correlated and compared with other tools, which will likely also have a corresponding timestamp.

3.1.1 Data Overview

At the highest level of aggregation, to provide analysts with a visual overview of the entire data set, TNV includes a histogram of the relative network traffic activity of the entire dataset (labeled *2* in Figure 1). This overview display also provides the primary navigation for moving around in the data. By default, the entire data set is shown when a new file is opened or a live packet capture completes. The analyst can then move either of the scroll handles to effectively zoom in or out the data on the main display, described below. Moving the scroll handles reduces or increases the time interval for each column in the visualization matrix and determines how much of the data is shown in the main visualization. The current time interval represented by all columns is shown to the right of the legend panel (labeled *3* in Figure 1). The data currently displayed in the main visualization (labeled *1* in Figure 1) is the shaded area between the handles. To zoom into the data, the handles are moved closer together, decreasing the time interval for each column; to increase the amount of time for each column, the handles are moved farther apart. This provides a very high-level overview of the data and keeps the analyst aware of the currently displayed location within the data set.

3.1.2 Matrix Visualization

The main visual component of TNV (labeled *1* in Figure 1) combines a matrix display of host IP address and network packet timestamp with a link display explicitly showing connectivity between hosts. The visualization matrix displays time on the x-axis and all host IP addresses (source and destination) available in

the data set along the y-axis, sorted by IP address. Each column represents a time interval, and each row a host, labeled identically on both sides of the display. The number of packets for that time interval is encoded in the color of the resulting box. This user-defined color-to-number-of-packets mapping is shown in the center of the legend panel (labeled *3* in Figure 1). In the example shown in Figure 1, gray represents a relatively low number of packets and red a high number of packets, with lighter hues within each color scale representing gradations along the scale. Thus, the analyst can very quickly identify hotspots, areas with higher amounts of traffic, within the data set. Because the visualization is designed around a timeline, the analyst can also easily identify trends in network activity for individual hosts. For example, if each time interval has low number of packets interrupted by a time interval with a very large number of packets, this may warrant further investigation.

Similar to the "home-centric" perspective described in Ball, Fink, and North [5], TNV emphasizes local hosts because of their paramount importance from an ID analyst's perspective than remote hosts. The analyst can set an IP address range that constitutes their home, or local, network, and the hosts in the data that meet this criterion are subtly differentiated. These local hosts have bolded, larger labels (shown in the identical labels at the right and left of the display) and the height of the rows can be increased based on a user-defined setting. In this way, it is easy for the analyst to see which hosts are part of their network, and thus under their purview. This is important because certain kinds of network traffic, such as network monitoring traffic, can be expected from hosts within a network, but would be suspicious if originating from an external host.

3.1.3 Network Links

While visualizing the number of packets is a useful starting point for exploring network data, it is not enough by itself to allow analysts to draw detailed conclusions from the data. It would be obvious where hotspots in the data are, but would not reveal any additional insight as to why those areas have a large amount of activity. To provide the analyst with additional information about the nature of the data, TNV displays network links between hosts within a single time period, as shown in the center of the visualization in Figure 1. The lines are drawn within each column to represent the network links between two hosts that occurred within that time period. Links are drawn from the source of the link to the destination of the link, starting from either side of the column. In most cases links will be displayed as an *X* pattern, because hosts often send and receive data as they communicate. It should be noted that the absence of this *X* pattern may indicate a scan where a host sends packets crafted to determine if ports are active, but without expecting a reply. This can lead to display clutter and future versions may include an option to draw unique links only once, rather than for each source.

The color of the links, defined by the user, shows the protocol of traffic those links represent. Like the color-coding for the hosts, these preferences are always displayed in the legend panel (labeled *2* in Figure 1, at left). By default, links are drawn with a very low opacity and a fixed, relatively narrow width. This allows the analysts to get an overall sense of the link communications between hosts without having the display too cluttered. There are various filtering mechanisms, described below, for highlighting or emphasizing links that match the analyst's criteria to encourage data exploration.

It is crucial in network traffic analysis to understand the communication patterns between hosts, but the current tools used by ID analysts, such as Ethereal, require that the analyst mentally correlate these patterns. By making the links visually explicit, it is hoped that not only will TNV assist ID analysis, but also make it easier for novices to learn what *normal* activity on their network looks like. For example, it would be expected that a public web server would have links from both local and remote hosts, so it would not be unusual to see a high number of links connecting to it. Whereas an intranet web server should only have connections from local hosts. Using textual tools like Ethereal, the analyst must store and recall IP addresses as they scan through their data, but using a visual tool like TNV that explicitly shows the links between machines, the cognitive burden placed on the analyst is greatly reduced allowing them to focus on solving the intrusion problem and not on deciphering the textual data.

The link display has similarities with parallel coordinates, which consists of parallel axes and line segments between them [15, 16]. In parallel coordinates the axes represent data attributes and the intersection of the line segments and axes represent the value of that observation for the variable represented by that axis. Whereas in parallel coordinates the axes each represent different data attributes, in TNV the parallel axes represents different values for the same attribute, time. The value of each intersection is constant, the host IP address. Using this kind of display can reveal relationships between hosts and how those relationships change and evolve over time. Unfortunately, the distinct advantage of being able to view the relations between multiple attributes that parallel coordinates offers cannot be duplicated in this kind of display. However, some of the recognizable patterns in parallel coordinates can be seen in the link display (and in the port activity display, discussed below), such as fan-in and fan-out links, where one host is communicating to multiple other hosts. This link display could also be compared to more traditional link and node displays, but here the node placement is fixed by the position of the host in the matrix.

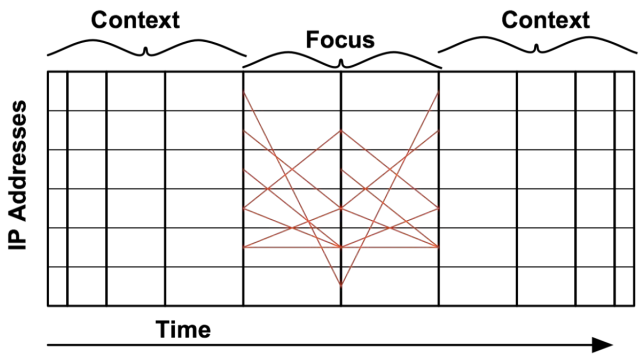

Figure 2. Bifocal display used in TNV.

3.1.4 Focus + Context

The importance of context in the ID analysis performed by our participants suggested the use of a focus + context approach. Thus, TNV utilizes a type of bifocal display [24], shown in Figure 2. The focal area, in the center of the display, shows link details within wider columns. The context area, to either side, has gradually decreasing column widths, based on a user-defined scaling preference. The gradually decreasing width of the columns provides continuity between the focal and contextual areas, without a sudden jump to a smaller width. To reduce clutter without sacrificing much additional information, the context area does not show link activity. This approach allows the analyst to explore the area of interest in greater detail, zooming in on traffic patterns, while still presenting the contextual information about the time periods before and after the area of interest. This distortion along the x-axis is similar to the Perspective Wall [20],

but without the simulated three-dimensional effect used in that system.

3.1.5 Port Activity

The main visualization allows the analyst to easily recognize areas of high activity, through host box color, and trends of link communications between hosts. For a more detailed view of the host communications, TNV offers a display of port activity (labeled *7* in Figure 1). Port activity is shown by the connections between two parallel axes representing the source and destination ports for the selected host. The source port activity is shown on the left axis, the destination port on the right axis, and the connections between them are drawn as line segments between each axis. The color of these connections is the same as the color of the links in the main visualization, set by the analyst. Either the source or destination port activity, or both, for a selected host or hosts can be viewed. For example, if source port is selected and the selected host is a web server, the source port for that host will likely be TCP port 80. So the analyst selects whether to view the port activity where the selected host(s) is the source or the destination. While generally the source ports are going to be important for servers and destination ports for clients, it is left to the analyst to decide if they want to specify one or the other to examine.

The relative number of connections for each port is shown by the height of the boxes drawn on top of the parallel axes. Each box represents a single port. In Figure 1, there are a number of relatively equally active source ports, all going to one destination port. This kind of many-to-one relationship is important in ID analysis and is easily recognized in this kind of display. By showing the relative amount of port activity over time, this display can also show the kind of strange, one-time port activity often indicative of a slow port scan that can often go overlooked with text-based tools. For example, if most of the traffic to a particular host is Telnet traffic and for several successive time periods this is interspersed with single connections to other ports, this could be indicative of a port scan that is using evasive measures.

3.2 Interaction

TNV is designed to encourage the exploration of network traffic in support of the analysis task and the tool includes mechanisms for filtering links and picking areas of interest to gain packet details.

3.2.1 Emphasizing Links

By default, links are drawn translucently in order to give the analyst a sense of network connections, but TNV also provides several filtering mechanisms to highlight areas of interest. Rather than completely remove links that do not meet the analyst's filtering criteria, the display emphasizes those links that do by increasing their opacity, while keeping the non-matching links more transparent. This alerts the analyst to the areas of interest without removing the context that all of the other links provide.

The simplest of these emphasis filters allows the analysts to highlight certain protocols, such as TCP or UDP (labeled *6* in Figure 1). In addition, the analyst can also enter in a list of ports to be highlighted, and whether or not to match on the source, destination, or both. These simple filters can be used in combination or separately to alert the analyst quickly to areas of interest. For example, if an ID analyst becomes aware of a new vulnerability associated with a particular port, then emphasizing that port in the links will immediately identify those hosts that use that port for communication.

Analysts can also select a host box or multiple boxes, as described in the next section, to highlight the links associated with

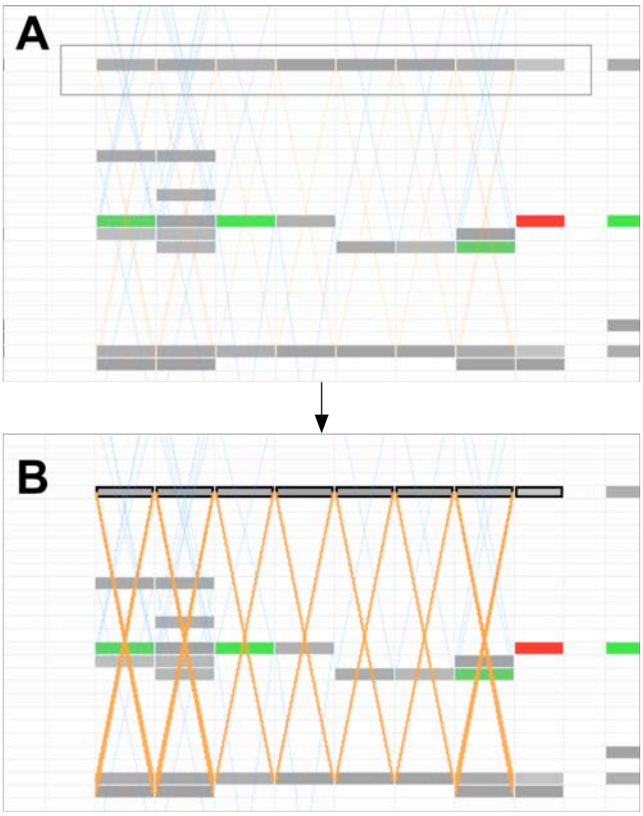

Figure 3. Selection: top (A) shows the user dragging the mouse over a single host across time periods; bottom (B) shows the resulting highlighting of both hosts and links.

that host or hosts in the selected time period. These links become opaque and the stroke thickness of the lines is increased, making it clear to which other hosts the selected host(s) is communicating.

Another means of emphasizing links is accomplished by selecting an area directly on the visualization. Any hosts and links within the selection area are highlighted. As the analyst clicks and drags the cursor over an area a bounding box is drawn around all hosts that fall within that area and the links whose endpoints are within that area have their opacity increased. This is especially useful for identifying all of the activity associated with a particular host, as shown in Figure 3. In this example, a single host is emphasized across multiple time periods. This direct manipulation method is similar to the concept of "timeboxes" in the TimeSearcher tool, in which boxes are drawn over line graphs and the lines that meet the constraints set by the box are highlighted [14].

3.2.2 Details on Demand

In addition to displaying port activity and emphasizing the links associated with the host, picking a host or multiple hosts reveals the packet details associated with that time period. When a host box or boxes are selected (labeled *A* Figure 1), the table (labeled *4* in Figure 1) is populated with a summary of all of the packets showing the time of arrival, the source and destination address and port, and a summary of the other packet headers. When a row in the table is selected, the details of an individual packet are displayed (labeled *5* in Figure 1). This shows all of the packet headers as well as the packet contents, or payload, in a tree format, similar to Ethereal.

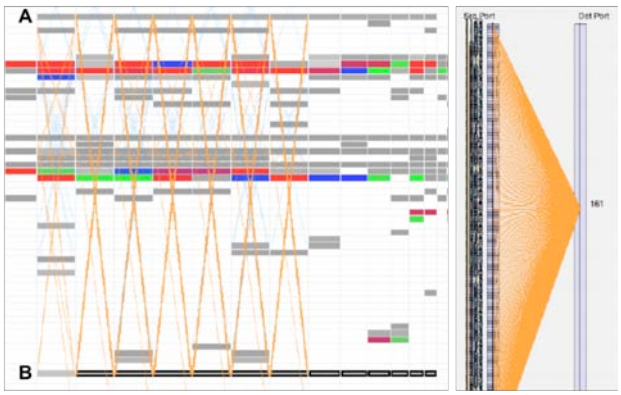

Figure 4. (Left) Host B is attacking Host A with prolonged SNMP attack; (right) port activity associated with Host B for selected time periods.

For highlighting links, viewing port activity, and viewing packet details, a single host or multiple hosts can be selected. The direct manipulation method of highlighting can be used to select multiple hosts or a modifier key can be pressed while selecting hosts. These methods permit the comparison of packet details of multiple time periods for a single host or multiple hosts in a single time period, or both.

While these textual details are the least visually interesting part of TNV, they form the essential basis for packet-level analysis. High-level visualizations are useful at providing an overview of activity and can reveal patterns and anomalies that would be difficult to discover by examining the raw data, but for ID analysis, the details must be present. In TNV, the visualization provides the starting point to aid the analyst in identifying areas of interest and prevents the analyst from getting lost while examining the low level details. 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.

4 TNV Usage Scenarios

In this section, several hypothetical scenarios demonstrating the usage of TNV will be presented. It is hoped that this discussion will demonstrate the utility of TNV in several real-world situations, but it should also be noted that because TNV is intended for analysis, which ultimately requires examining details, these scenarios represent more of a starting point than a complete demonstration of how TNV can be used. The scenarios below demonstrate typical uses of a network analysis tool, as described by the participants in our field study. The data used in these examples were collected ad hoc on a university network or from the Lincoln Laboratory 1999 intrusion detection evaluation [3].

4.1 Attack Analysis

The security event that acts as the starting point for ID analysis tasks could be an alert generated by an IDS or other monitoring system. In this scenario, it is assumed that the analyst has a time period and the involved hosts, but needs to examine the data in detail to determine the accuracy and severity of the attack. Figure 4 shows two cutouts of the activity surrounding this attack. On the left is a section of the matrix display, with the hosts involved in the attack labeled ‘A’, a local host, and ‘B’, an external host. The analyst knows that the attack being analyzed has to do with simple network management protocol (SNMP) traffic, so port 161 is entered in the port highlighting filter, which emphasizes the links between host A and B. This would be immediately suspicious,

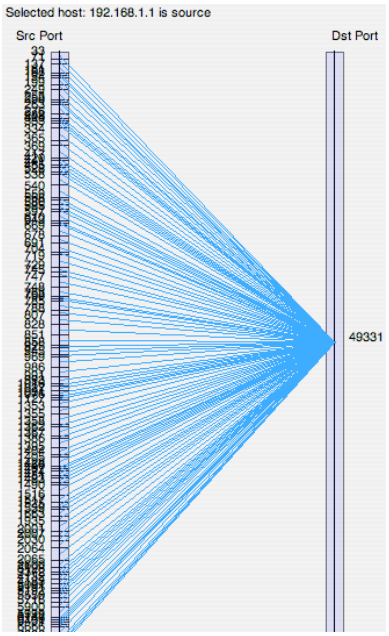

Figure 5. Port scanning activity.

since external hosts should rarely, if ever, be legitimately querying for SNMP data. To see more detail about the SNMP traffic, the analyst would select the host boxes associated with the external host (labeled as B in Figure 4). This shows the port activity to the right of Figure 4, where it becomes clear that host B is directing all of its traffic at port 161. In order to learn more details about the attack, the analyst would look at the packet detail table to uncover the actual content of the packets.

In this very simple example, it is assumed that the analyst has a starting point to begin their analysis, but even without this information, some exploration of the data could have revealed the attack. Aside from this SNMP attack, the other UDP links in this data set are primarily domain name service (DNS) requests, which typically occur in bursts as clients query the DNS server. This SNMP UDP traffic, on the other hand, is sustained over a long period of time (about 30 minutes). Because of this, the traffic would likely be thought of as suspicious enough to warrant further investigation.

4.2 Port Scanning

The port activity display can easily reveal an indication of port scanning activity, as shown in Figure 4. In this display, the port activity view shows the source port activity for the selected host, which is being port scanned. All of the packets being sent back to the host doing the scanning are RST + ACK packets, which tell the scanner that those ports are closed. In this case, the scan occurred quickly, so TNV had several host boxes with very high numbers of packets (denoted by color), that would lead the analyst to look more closely at the details of those time periods. For a slow scan occurring over a longer period of time, multiple time periods for a host can be selected to reveal the same type of many-to-one relationship.

In this scenario, while the indications of a port scan are noticeable, the strength of TNV would be not simply identifying the scan (for which there are tools designed specifically for that purpose), but to learn more about the scanning. In this example, the analyst could determine what the host being scanned revealed to the scanner—in this case the host being scanned responded to

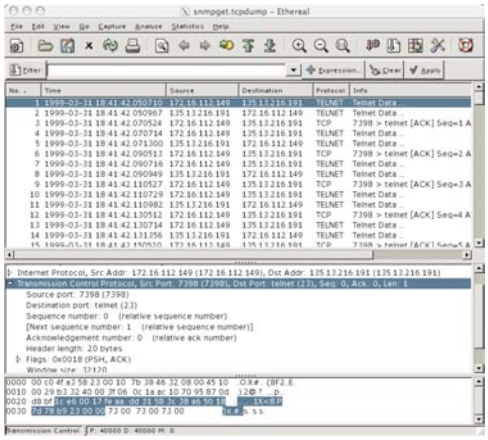

Figure 6. Ethereal network analysis tool.

https requests with a SYN + ACK packet, indicating that the port is open. TNV allows the analyst to not just see an indication of a port scan, but to view the details of how the scan is manifested and how each of the involved hosts responds.

4.3 Learning the Network

The importance of "knowing the network" in intrusion detection cannot be understated. In our previous work, ID analysts repeatedly defined the most crucial aspect of accomplishing ID was having an intimate knowledge of the environment that they work in [13]. Gaining this understanding of what is normal (and therefore abnormal) on a network is a nontrivial task. Most ID analysts use tools such as Ethereal (shown in Figure 6) in order to gain an understanding of the way network packets look on their network. This daunting task can help analysts learn about how individual packets appear, but it does not help them gain an understanding of normal network activity at a more macro-level. This is one of the greatest strengths of a tool like TNV—it encourages both the high-level understanding of network traffic through the main visualization, while also allowing multiple levels of exploration down to the raw packets.

Although network traffic manifests itself differently on different networks, TNV can facilitate the process of learning what constitutes normal traffic. Different kinds of traffic have different signatures in a visualization like TNV. For example, web traffic is generally displayed as sporadic bursts of high activity as clients request pages from a web server. Login traffic, like secure shell or telnet, however, tend to show up as sustained traffic with high levels of activity. Domain name or network time protocol traffic are usually sporadic bursts of low activity. These examples are highlighted in Figure 7. These are generalizations of just a few examples, but learning how different protocols are displayed in TNV can help analysts learn the big picture of network traffic, while also allowing the analyst to learn the more subtle differences in lower-level packet details.

5 Future Work

TNV has great potential for facilitating the analysis task of intrusion detection and the learning process for understanding normal network traffic. Preliminary usability testing on an earlier version of the tool demonstrated that TNV is easy to learn, and helped to discover some usability issues that have been fixed in the current version. However, in order to determine the utility of TNV, we will need to conduct evaluations. Possible evaluations include: 1) a lab-based comparison of TNV versus Ethereal, the current standard for analysis; and 2) a field-based evaluation of TNV with ID analysts using the tool on their own network with their own data. In the former, we will examine whether a visual tool like TNV can augment the current functionality of Ethereal in typical network analysis tasks. In the latter, we acknowledge the importance of the intimate knowledge that analysts have (or must gain) of their own network, and will evaluate the utility of TNV in the natural context of use for both novices and experts.

Figure 7. Sample traffic patterns: web, Telnet, and NTP traffic.

Functionally, we plan on increasing the filtering capabilities of TNV to permit more flexible data exploration, such as filtering on TCP flags. Another planned area of improvement is to allow the rows in the matrix to be reordered, possibly by a clustering algorithm or by hand. Currently, the y-axis is ordered arbitrarily by IP address, which ensures that hosts on the same subnetwork are adjacent, but does not allow a great deal of flexibility. Currently, about one hundred hosts can easily fit on a 1280x1024 display while still showing the detail table. To make TNV more scalable, it may be useful to extend the bifocal display to the y-axis as well, so that hosts towards the edges are displayed as smaller, to allow more hosts on a single display. Finally, it would be useful to integrate port activity onto the main visualization display, so that the analyst can view port activity for the entire data set or selected portions of the data set, in addition to the current functionality of viewing port activity for a single host.

6 Conclusion

In order to accomplish the analysis task, ID analysts need access to the details of network traffic, require tools to help them reconstruct events, and need contextual information to assist their decision-making processes. In order to facilitate the analysis process, we developed an information visualization tool to support ID analysis, TNV. TNV includes a navigation mechanism that provides the analyst with an aggregated overview of the entire data set and constant awareness of the sub-sample currently being viewed on the main display. The bifocal, matrix-based visualization can alert the analyst to hotspots in the data and reveal patterns of activity in the network over time. Linked to the main visualization are a port activity view and a table of the textual network packet details. The port activity view provides a visual overview of relative port activity and connections for selected hosts, while the details table offers access to the raw packet-level details required for the analysis task. TNV also includes several filtering and highlighting mechanisms for exploring link patterns and activity.

The TNV visualization tool augments the analysis task of intrusion detection by permitting analysts to explore the details of the network traffic data while preserving the "big picture" of how those details fit into the larger context of network activity. This kind of visual tool can help novices learn the idiosyncrasies of their network from both a macro- and micro-level. While further evaluation is required to validate TNV's utility, it provides

support beyond the monitoring tasks that has potential benefits for both novice and expert networking and security analysts.

REFERENCES

[1] Ethereal. http://www.ethereal.com/.

[2] Jpcap. http://jpcap.sourceforge.net/.

[3] MIT Lincoln Laboratory 1999 DARPA intrusion detection evaluation data set. http://www.ll.mit.edu/IST/ideval/data/1999/1999_data_index.html.

[4] Tcpdump. http://www.tcpdump.org/.

[5] Robert Ball, Glenn A. Fink and Chris North. Home-centric visualization of network traffic for security administration. In *ACM Workshop on Visualization and Data Mining for Computer Security (VizSEC/DMSEC)*, pages 55-64, 2004.

[6] Richard A. Becker, Stephen G. Eick and Allan R. Wilks. Visualizing network data. *IEEE Transactions on Visualization and Computer Graphics*, 1 (1): 16-28, 1995.

[7] Gregory Conti and Kulsoom Abdullah. Passive visual fingerprinting of network attack tools. In *ACM Workshop on Visualization and Data Mining for Computer Security (VizSEC/DMSEC)*, pages 45-54, 2004.

[8] Kenneth C. Cox, Stephen G. Eick and Graham J. Wills. Visual data mining: Recognizing telephone calling fraud. *Journal of Data Mining and Knowledge Discovery*, 1(2): 225-231, 1997.

[9] Robert F. Erbacher, Zhouxuan Teng and Siddharth Pandit. Multi-node monitoring and intrusion detection. In *Proceedings of the IASTED International Conference on Visualization, Imaging, and Image Processing*, pages 720-725, 2002.

[10] Robert F. Erbacher, Kenneth L. Walker and Deborah A. Frincke. Intrusion and misuse detection in large-scale systems. *IEEE Computer Graphics and Applications*, 22(1): 38-48, 2002.

[11] Luc Girardin and Dominique Brodbeck. A visual approach for monitoring logs. In *Proceedings of Twelfth Systems Administration Conference (LISA '98)*, pages 299-308, 1998.

[12] John R. Goodall, Wayne G. Lutters and Anita Komlodi. The work of intrusion detection: Rethinking the role of security analysts. In *Proceedings of the Americas Conference on Information Systems (AMCIS)*, pages 1421-1427, 2004.

[13] John R. Goodall, Wayne G. Lutters and Anita Komlodi. I know my network: Collaboration and expertise in intrusion detection. In *Proceedings of the ACM Conference on Computer-Supported Cooperative Work (CSCW)*, pages 342-345, 2004.

[14] Harry Hochheiser and Ben Shneiderman. Dynamic query tools for time series data sets: Timebox widgets for interactive exploration. *Information Visualization*, 3(1): 1-18, 2004.

[15] Alfred Inselberg. The plane with parallel coordinates. *The Visual Computer*, 1: 69-91, 1985.

[16] Alfred Inselberg, Multidimensional detective. In *Proceedings of IEEE Symposium on Information Visualization*, pages 100-107, 1997.

[17] Klaus Julisch. Clustering intrusion detection alarms to support root cause analysis. *ACM Transactions on Information and System Security*, 6(4): 443-471, 2003.

[18] Klaus Julisch and Marc Dacier. Mining intrusion detection alarms for actionable knowledge. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 366-375, 2002.

[19] Kiran Lakkaraju, William Yurcik and Adam J. Lee. NVisionIP: NetFlow visualizations of system state for security situational awareness. In *ACM Workshop on Visualization and Data Mining for Computer Security (VizSEC/DMSEC)*, pages 65-72, 2004.

[20] Jock D. Mackinlay, George G. Robertson and Stuart K. Card. The perspective wall: Detail and context smoothly integrated. In *ACM Conference on Human Factors in Computing Systems (CHI)*, pages 173-179, 1991.

[21] John McHugh. Intrusion and intrusion detection. *International Journal of Information Security*, 1(1): 14-35, 2001.

[22] Jonathan McPherson, Kwan-Liu Ma, P Krystosk, T Bartoletti and M Christensen. PortVis: A tool for port-based detection of security events. In *ACM Workshop on Visualization and Data Mining for Computer Security (VizSEC/DMSEC)*, pages 73-81, 2004.

[23] Tamara Munzner, Eric Hoffman, K. Claffy and Bill Fenner. Visualizing the global topology of the MBone. In *Proceedings of the IEEE Symposium on Information Visualization*, pages 85-92, 1996.

[24] Robert Spence and Mark D. Apperley. Data base navigation: An office environment for the professional. *Behaviour and Information Technology*, 1(1): 43-54, 1982.

[25] Markus Stolze, Rene Pawlitzek and Andreas Wespi. Visual problem-solving support for new event triage in centralized network security monitoring: Challenges, tools and benefits. In *GI-SIDAR conference IT-Incident Management & IT-Forensics (IMF)*, 2003.

[26] Xiaoxin Yin, William Yurcik, Michael Treaster, Yifan Li and Kiran Lakkaraju. VisFlowConnect: NetFlow visualizations of link relationships for security situational awareness. In *ACM Workshop on Visualization and Data Mining for Computer Security (VizSEC/DMSEC)*, pages 26-34, 2004.

[27] William Yurcik, James Barlow, Kiran Lakkaraju and Mike Haberman. Two visual computer network security monitoring tools incorporating operator interface requirements. In *ACM CHI Workshop on Human-Computer Interaction and Security Systems (HCISEC)*, 2003.