Proceedings of the 2005 IEEE
Workshop on Information Assurance and Security
United States Military Academy, West Point, NY

User Requirements and Design of a Visualization for Intrusion Detection Analysis

John R. Goodall

Abstract – This paper reports on the user requirements gathering activities and design of an information visualization tool for analyzing network data for intrusion detection (ID). User-centered design methods have been widely used for many years. However, innovative visualization displays are often developed with limited consideration of user needs in the context of real-life problems. While it can be argued that this is required to generate creative new solutions, the resulting tools may not fully support actual users in their daily work. We studied ID analysts' activities in order to understand their work practices. This resulted in a simple task model of ID work and guidelines for visualization support. Noting the lack of current visualization support for the analysis ID task and grounded in the actual needs of ID analysts, we designed a visualization prototype for investigating network traffic.

Index terms – Information visualization, user centered design, intrusion detection.

I. Introduction

Intrusion Detection (ID), the monitoring of system or network events for signs of malicious or abnormal activity, has become a critical component of many organizations' security infrastructure; a recent survey showed that 81% of respondents employed ID technology [1]. Intrusion Detection Systems (IDSs) attempt to automatically identify successful and unsuccessful attacks or abuse of computer systems [2]. Because of the potential for false alerts, missed attacks, and self-damaging responses to inaccurate alerts, fully automated IDSs, while valuable for certain kinds of attacks, are unlikely to be a completely effective solution. Instead, such systems require vigilant oversight by human security analysts. However, there has been little research into understanding and supporting human ID-related tasks.

The work of analyzing network ID data is a complex, difficult task that requires experience, knowledge of networking and system protocols and behaviors, and knowledge of the operating environment. To compound the problem, analysts must constantly keep up to date with changing network and system configurations, newly discovered software and operating system vulnerabilities, and new intrusion methods. The sheer number of alerts generated by an IDS can be overwhelming, and the number of false positives may be extremely high. IDSs can trigger thousands of alerts per day, up to 99% of which are false positives [3]. Differentiating the large number of false positives from the true malicious activity is a daunting task that relies heavily on the knowledge and experience of the human analyst.

The output from many IDSs consists of either textual or tabular data, sometimes augmented with simple charting displays, but this fails to support the strong analytic capabilities of humans in doing ID. A recent technical report on the state of ID emphasizes this problem:

> Vendors attempt to fully automate intrusion diagnosis. A more realistic approach is to involve the human in the diagnostic loop. While computers are capable of examining large quantities of low level data, they cannot match a human's analytic skills. [4]

Keeping the human analyst "in the diagnostic loop" may be facilitated through information visualization, which uses computer graphics to amplify cognition by taking advantage of human perceptual abilities [5]. Information visualization takes advantage of strong human pattern recognition skills through visual representations that can often make patterns and anomalies evident to the user – finding patterns and anomalies is central to ID.

Our research seeks to gain an understanding of the human ID-related tasks in order to design support tools that build on the strengths of both human analytic skills and machine processing and display capabilities. This paper will present the results from our user requirements gathering activities for information visualization support for ID and introduce a prototype visualization tool designed to aid analysts in one specific ID task – the analysis of network data.

II. Related Work

This section highlights contemporary research both in understanding the needs of various groups doing intrusion detection work and in applying information visualization to ID.

John R. Goodall, jgood@umbc.edu
Department of Information Systems, UMBC
1000 Hilltop Circle, Baltimore MD 21250

A. User Requirements for Security Visualizations

Yurcik, Barlow, Lakkaraju, and Haberman [6] described user requirements gathering through interviewing security operators at the National Center for Supercomputing Applications and two incident response centers. One of their primary findings was the need for supporting "situational awareness." They built information visualization tools to support situational awareness by providing analysts with an overview of an entire network on a single display.

Stolze, Pawlitzek, and Wespi [7] also described the importance of situational awareness. This research investigated operators' problem-solving tasks in the 24/7 monitoring of multiple networks in a security operations center. They presented a descriptive model of operators' tasks that formed the basis of a visualization tool using parallel coordinates and scatterplot displays to support the task of new event triage. Their task model has several commonalities with our own, described below, but their model is tailored specifically to the event classification process rather than on the work of ID as a whole.

Ball, Fink, and North [8] report on the information security needs of systems administrators in a university environment based on interviews, and developed a visualization in collaboration with the interviewees. Noting network administrators' foremost interest in what is happening on their own network, they developed a "home-centric" visualization prototype.

Our requirements gathering process, described below, confirm and build on this body of work. Rather than focus on the needs of one particular group, such as a security operations center monitoring external sites or the specialized needs of a university environment, our work focuses on a broader group of users and looks for commonalities across different kinds of organizational security needs, levels of analysts' expertise, and size and makeup of operating environments.

B. Visualization for Network Analysis

Many information visualization displays of computer network data have used a link and node approach to explicitly show communications between various hosts on a network [e.g., 9, 10]. Some systems have placed nodes according to their geographical location, while others have made use of algorithms to cluster together "similar" nodes. However, designing effective, scalable displays using this approach can be difficult, including problems of meaningful node placement and links crossing or overlapping that lead to occlusion. Despite these problems, link and node displays are useful in explicitly visualizing small amounts of network traffic.

In the domain of information security, there have been several information visualization tools designed to show link relationships between hosts. VisFlowConnect used parallel coordinates to show links between hosts and animation to display temporal data attributes [11]. Girardin and Brodbeck [12], Stolze [7], and Conti and Abdullah [13] also used parallel coordinates to show relationships between multidimensional network data.

In addition to showing the relationships between hosts, it is essential in ID work to understand the current state of the network, displaying activities related to hosts on a network. NVisionIP used a scatterplot of IP addresses to show the current state of a class-B network using Netflow data [14]. PortVis also used a scatterplot-type approach in its main visualization to display port activity in a one-hour time period [15]. Other systems show both link and state information, such as the visualization of a single system's CPU activity and hosts connecting to it [16].

These visualizations present interesting, useful overviews of particular areas of interest and provide security analysts with much-needed situational awareness, particularly for the monitoring task, described below. However, in order for more detailed data analysis, the analyst must use a different suite of tools, which largely remain textually based. The prototype visualization presented in this paper is intended to complement these high-level, situational awareness tools by displaying an overview of network data linked to the data's precise details, which are required in ID analysis.

III. User Requirements Gathering

In order to both gain an understanding of how security analysts accomplish intrusion detection and determine characteristics of information visualization tools that will address the current limitations in ID, we conducted contextual interviews with nine ID analysts. These analysts had diverse backgrounds, primary job responsibilities (e.g., systems administrator), and organizational contexts. All participants were knowledgeable in networking and ID, and had hands-on experience with the open-source Snort IDS [17]. Interviews consisted of three core sections: participants' experience with and knowledge about ID, current ID work practices, and requirements and recommendations of potential information visualization tools. This section outlines some of the pertinent findings for information visualization in supporting ID and presents some lessons learned from the process of gathering user requirements. Quotes from our participants are identified by number (P1-P9) in the text.

0-7803-9290-6/05/$20.00 ©2005 IEEE.

A. Guidelines for Information Visualization Tools

1. Task-tool Fit

All of the participants adhered to a similar, high-level task model consisting of three broad phases: monitoring, analysis, and response. The monitoring task focuses on monitoring the IDS itself, but also includes the surveillance of other monitoring systems as well. The transition between monitoring and analysis occurs when an IDS alert or other trigger event (such as uncharacteristically high network traffic or a phone call from a colleague) is generated. In moving from monitoring to analysis, the analyst will triage the event, determining instantly if further analysis is required. If the event is determined to warrant further investigation, the next phase is a much more detailed analysis of the event. In moving from analysis to response, a diagnosis of the accuracy and severity of the event is constructed. If an alert is determined to represent an actual attack or other malicious activity, the analyst must form an appropriate response, such as creating an incident report for law enforcement or recovering data from backups. This section focuses on the monitoring and analysis tasks as they pertain to information visualization support.

For the monitoring task, a visualization that allows analysts to process data preattentively, without the need for focused attention [18], will be the most effective. This is described by one of the participants:

> So you can sit back … looking at the screen or glancing at the screen, and with that glance, you're going to get that information that you need to know [if] you got to react or you got to investigate further. (P4)

Monitoring does not need to support complex data analysis, it only needs to provide the analyst with enough information to make a snap decision whether or not something needs to be investigated further. One participant described a home-grown solution that supported a simple form of preattentive processing in a web-based console that scrolled new alerts as they came in; when the analyst noticed the screen "flashing," that was cause for further investigation: "you can see if the alerts [are] changing more than a few times a minute, you're going to pick up on it" (P3). A simple glance is enough to notice the screen rapidly changing, which gave this analyst a clue that something needs to be investigated without needing to focus his full attention on a display.

The analysis task, however, must emphasize accuracy and completeness above speed of human processing. In beginning the analysis task, the analyst has some idea about the event, a hypothesis that must be proved or disproved. This initial insight comes from data generated during the monitoring task and experience. From there the analyst must delve deeper into the data to reveal the true nature of the event. This calls for information visualization support that emphasizes data exploration. Visualizations for the analysis task should support multidimensional data analysis and present the data from multiple viewpoints. Another goal in supporting analysis should be the correlation of multiple data sources together in one display. Simple static information like what operating system or services are running on a host can be difficult to keep track of mentally when networks are large or susceptible to change, so analysis tools could benefit even from this relatively simple information. Relevant dynamic data from other sources, such as firewalls or system logs, would also be useful for analysis. Participants described this ad hoc data gathering and mental correlation process as particularly onerous, but vital to analysis.

2. Overview and Details

For both the monitoring and analysis tasks, providing an overview is critical. In monitoring this can provide situational awareness. In analysis, an overview will keep analysts from losing sight of the "big picture" (P3) when they are examining low-level details of an event. For the analysis task, the precise details of an event must be readily available. Participants described the difficulties in moving back and forth between these macro- and micro-levels of details. For information visualization tools, this implies a combination of linked visual and textual displays to accommodate switching between levels. Participants were also adamant about the need for all of the details in network data for analysis tasks, not just packet headers, but the packet payloads as well. One participant described the importance of having all packet details available in the analysis task: "the most important [thing] would still have to come down to just having the raw data" (P7).

3. Contextual Information

Contextual information, both historical and current, is essential in the analysis task. In order to gain this context, analysts rely on a myriad of data sources and tools that provide historical and current state information. The most common contextual data sources for the participants were:

- Firewalls and other networking devices
- Network packet capture tools
- System and application logs
- Current system and network performance data
- Vulnerability and security scanner results
- IDS alert history
- File integrity checking tools

The analyst must locate or gather the data, determine if the data are relevant, and correlate the data with the event being analyzed. This is a time-consuming, difficult task

that is necessary to gain a full understanding of an event. This includes not just current state information, such as what services is a host running at immediately after the alert, but also historical information, such as if host under attack has been targeted previously in a similar fashion.

Much of this contextual information is gathered or queried on an ad hoc basis, often using a network sniffer, such as Tcpdump (tcpdump.org) or Ethereal (ethereal.com), to gather all network traffic related to an event's targeted host. The difficulty lies in not just collecting and parsing all of this data, which in itself is a nontrivial task, but in correlating all of the data together with the information surrounding the event.

4. Flexibility

Every network is different. Computer network topologies are unique, systems and services on a network are constantly evolving and changing, and network traffic is rarely stable. Participants repeatedly stressed the importance of "knowing" the network they are charged with protecting: "I know my network" (P3). This means that "in some environments, it is normal to have 200 alerts a minute, but for me it's [not]" (P4). Because of the idiosyncratic nature of computer network, any visualization tool should be flexible and allow the user to customize it for their own particular environment: "different folks need to see different things" (P2).

5. Intuitive Visual Layout

Participants were deeply mistrustful of techniques that used "black box" methods for clustering or organizing data. While clustering or aggregation of data is likely to be a necessary component of any information visualization with such vast amounts of data to visualize, participants wanted to understand *how* the data is clustered. That is, the visual layout should be intuitively understood by the user, exposing rather than hiding the reasoning for the structure. It is not enough to provide users with the ability to easily recognize an anomaly, an ID analyst must know *why* it is anomalous in order to construct an accurate diagnosis.

B. Lessons Learned

1. Understanding Work Practice

Understanding ID work practices was imperative in beginning to design tools to match the realities of analysts' tasks. This understanding confirmed that information visualization could facilitate some ID tasks and allowed us to develop requirements for supporting the individual tasks. For example, our initial assumption that speed is paramount in ID, while true for monitoring, proved less significant for the analysis task. In designing visualization support for ID, or any area of information security, it is crucial to be aware of exactly what tasks are being designed to support. While a more comprehensive solution that allowed users to seamlessly shift between the monitoring, analysis, and response tasks should be the end goal in supporting ID work, understanding those tasks and their contexts is a critical first step in design.

2. Using Visual Aids

In attempting to elicit requirements for any information visualization support tools, it can be extremely difficult for users to envision potential displays. Most users have never heard of, or misunderstand, the term information visualization, and lack significant experience with such systems. It was constructive to show paper-based screenshots of existing visualizations to give participants an understanding of what visualization was and how it might be applied to supporting their work. Users were shown sample screenshots from an ID domain specific visualization [16] and from a more generic visualization that used a scatterplot structure with dynamic queries [19]. In doing so, we noticed several interesting reactions:

- Participants were immediately excited by the potential of information visualization in ID, and were eager to suggest how the approaches shown to them could facilitate their work.
- Participants seemed to find it difficult to envision new displays, but instead focused on what attributes of their data would be useful in the visualizations shown to them.
- Possibly due to the relatively easy to understand layout and familiarity of the structure, the generic visualization was considered more applicable to ID tasks than the more complex, domain specific visualization.

The use of these paper-based props was effective in getting users to think about how information visualization could be used in ID, even if they often limited their discussion to displays that were similar to the ones we showed them. Perhaps showing more examples would stimulate a greater flow of ideas.

3. Context

Also important was being able to observe analysts interacting with their current systems. This not only gave us a framework for understanding the context that analysts work in, but also gave us an appreciation of the difficulties analysts faced in managing large amounts of data and confirmed our suspicion that information visualization could greatly facilitate ID work. In those cases where this was not permissible due to organizational security policies prohibiting outsiders from the workplace, analysts brought screen captures of their tools and walked us through their use. While these screen captures provided some understanding of their system interaction, it was much more limited than those cases

0-7803-9290-6/05/$20.00 ©2005 IEEE.

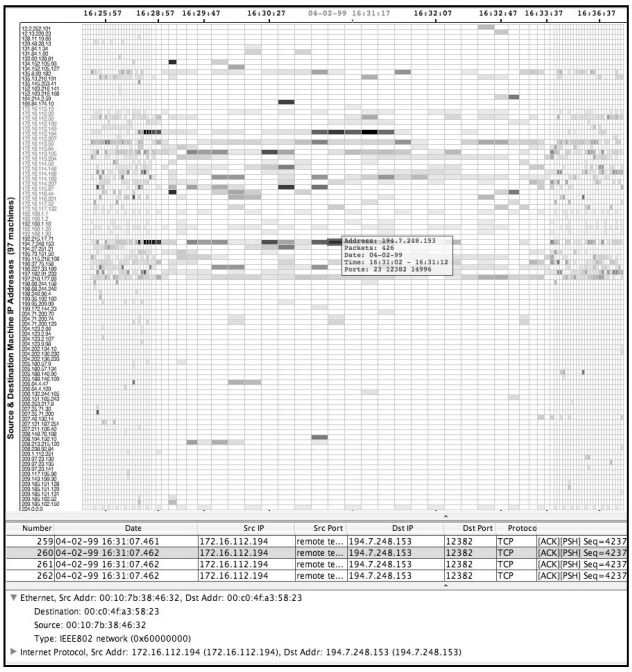

Figure 1: TNV showing 15 minutes of network traffic for 97 hosts. The top shows the visualization by time interval and IP address, the lower section shows packet details. The majority of the filled in boxes (in the center) are hosts on the "local" network.

when we were able to actually watch the interaction "live" as analysts described what they were doing.

IV. Visualization Design

TNV, the Time-based Network traffic Visualizer, is a visualization specifically designed to support the detail-oriented analysis task of ID. This main visualization provides an overview of the state of the network over time and can reveal patterns in the data. For example, in Figure 1, hosts that have near constant traffic (adjacent gray colored boxes) are likely involved in an interactive login session (Telnet), while those that have more sporadic traffic are likely client-server requests (such as web traffic or file transfers). This main visualization provides a high-level starting point for analysis, but because details are crucial in this task, analysts can examine individual network packet details.

The prototype is implemented in Java using the SourceForge jpcap library (jpcap.sourceforge.net), allowing the tool to run on a wide variety of platforms. Data can be captured live or read into the tool in libpcap (standard on Unix-type systems and available for Windows) format, which allows interoperability of data sets with other network traffic capture and analysis tools.

The remainder of this section will outline the rationale for TNV's design, the data source used, and the interface and interaction mechanisms of TNV.

A. From Requirements Gathering to Design

While functionality is currently limited, the basic structure takes into account several of the requirements from the users, notably the need for simultaneous high and low-level views within a single display. Because TNV is designed for analysis, exploration rather than speed is emphasized. In the analysis phase, the analyst has a hypothesis of what they are looking for and are often trying to prove or disprove the accuracy of an event and determine its severity.

In learning how analysts perform the analysis task and how they transition between tasks, we determined that time would be the focus of the visualization because it is most often used as the starting point for the analysis task. Time is the central theme of TNV for several reasons:

- All of the data sources and data collection tools used by our participants generate a timestamp, which despite being generated on different hosts often correspond nearly exactly (all participants use Network Time Protocol on their systems). Because the security trigger event may originate from any number of sources, the constancy of time across different sources lends itself to being the starting point for analysis.
- While the trigger event from monitoring may be an IDS or another monitoring system, it could also be something more ambiguous. The event could be a user reporting, for example, that "this morning it took much longer than usual to log in to the mail server." This kind of vague trigger event often makes beginning the analysis task from anything other than time problematic. Time is the one attribute universally 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 simple example, if immediately prior to the event being investigated, every host on the network was portscanned from a single destination, this could indicate that an attacker doing reconnaissance found and exploited a vulnerability.

Additionally, time was unanimously agreed among our participants as being the most important network data attribute. While source and destination address and port were deemed to be vital in both monitoring and analysis, time is the glue that allows multiple events to be correlated together.

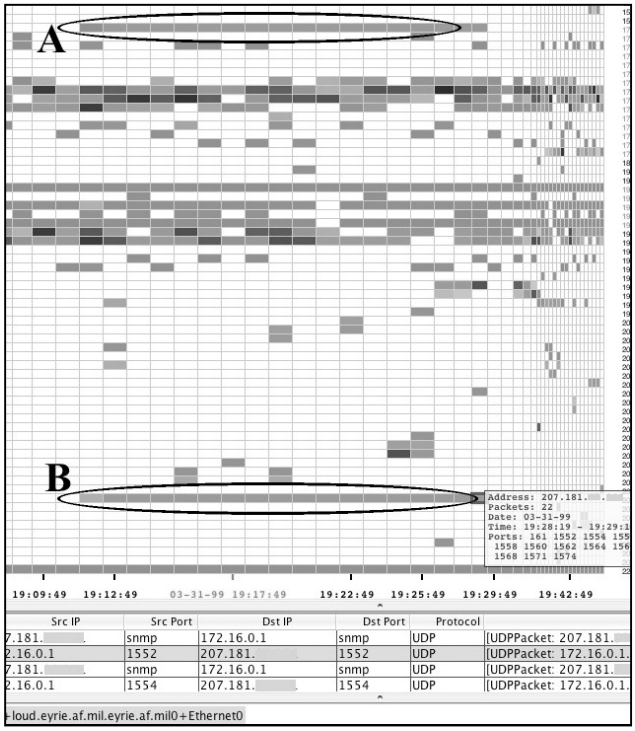

Figure 2: TNV showing 40,000 packets over one hour, where host A under a prolonged SNMP attack (port 161, shown in tooltip) from host B.

B. Data Source

TNV uses libpcap data, which is also used by the common Unix networking tools tcpdump and Ethereal, as the data source for TNV for the following reasons:

- *Familiarity*. All of the participants were already familiar with tcpdump and Ethereal, so we wanted to take advantage of this knowledge.
- *Flexibility*. A visualization that uses this source of data allows ID analyst to collect data remotely on any platform, and bring the data set to their workstation for analysis.
- *Details*. Because TNV is designed to support the analysis task, the data needed to include low-level details unavailable in many other data sources (not just all of the header information, but the packet payload as well).

Using this kind of data does involve a compromise, however, because libpcap data can grow extremely large very quickly, the current version of the prototype can run out of memory after fifty-thousand packets. TNV is able to open existing libpcap data files or capture data in real-time "on the wire."

C. User Interface

TNV shows network traffic in discrete, user defined, time intervals. The visualization is divided vertically by time, with each column representing a fixed time interval, and horizontally by IP address, so that each host has a series of adjacent rectangular "boxes" for each of the time intervals. (Although it is possible for a host to have multiple addresses, in this paper we refer to each address as a "host" for simplicity.) Thus, the display is essentially a grid of time interval and host. The color of each host box for a given time interval is defined by the number of packets within the time interval. Using the grayscale color setting, as seen in Figure 2, the darker the color the higher the number of packets. Figure 2 shows a prolonged Simple Network Management Protocol (SNMP) attack. The sudden sustained activity from the external host (B) is suspicious, and hovering the mouse over one of the host boxes, a tooltip shows port 161, the SNMP port, as one of the ports associated with that time interval. Tooltips show the host address, number of packets, time interval, and ports. It is rare that an SNMP query from an external host would be legitimate, since it is most often used for monitoring network performance. In the scenario shown in Figure 2, the user may have been alerted by an IDS alert, or may have noticed the sudden continuous network traffic from host B that corresponds to the traffic pattern for host A.

In ID analysis the activity immediately surrounding an event is of great importance, but even temporally distant data may be relevant. Because of this, TNV uses a simple focus + context approach, comparable to a 2D version of the Perspective Wall [20]. In TNV, the center columns of the display are equally sized and the outer columns become increasingly smaller (notice that the columns get increasingly smaller at the right of Figure 2). The center of the screen represents the "focus" area and to either side, the column widths decrease as each is drawn closer to the edge of the screen. This preserves the temporal context surrounding the time of interest to the analyst and allows larger chunks of time to be displayed at once.

The main visualization shows an overview that allows the analyst to quickly assess the state of the network and view trends, patterns, and anomalies over large or small spans of time. For more detailed inspection of the data, TNV displays communications between hosts and the details of those communications. To view the connections between hosts within a time interval, a user "unzips" a column, splitting the column in two and shifting each away from each other so the links can be displayed between them. The traffic protocol is encoded in the color of the link, defined by the user. Although overlapping links are currently a problem, this display can give the user a quick indication of communications between hosts. However,

since the links are aggregated, the prototype includes a mechanism for viewing the low-level details of individual network packets.

The details of the network communications are displayed in a sortable table, similar to the default summary panel in Ethereal, that shows the source and destination IP addresses and ports and a brief description of each packet. A row of the table can be selected to provide a more detailed view of a single packet. This detailed view allows the user to work from a highly aggregated overview level, down to the very detailed level of individual packets.

The user has control over the various aspects of the display through the control panel. The time interval (e.g., 30 minutes) is set using a combination of numeric text and the appropriate time unit. By starting with a higher time interval and progressively moving to a lower one, the user can effectively "zoom" into the data, showing a less aggregated view of the network traffic as the time interval value decreases. The chronological time (e.g., 06/01/05, 13:30:15) of the center column is controlled with a slider. If the slider is moved to the left, which marks the beginning of the data set, the focus of the display in the center is redrawn to the start of the data set. By default, the time is set to the center and the interval is calculated to display the entire data set; so an overview of the entire data set is presented to the analyst from the start. This combination of the time interval and the center column time determines which portions of the data set are drawn on the screen. This two-stepped operation is somewhat clunky and will be modified in future versions.

V. Uses of TNV

TNV is designed to support the ID analysis task. Because it is based on time, and all IDS alerts or other monitoring systems contain a precise timestamp, TNV can be used as an effective starting point for analysis. TNV should ground analysts in the "big picture" while allowing low-level exploration of traffic details. Determining the accuracy of a security event requires knowledge of events immediately preceding and following the alert, and TNV's time interval can be progressively reduced in order to get more precise information on these surrounding events. If an event, such as an IDS alert, is known to be representative of a true attack, TNV can facilitate identifying other hosts that may have been targeted by the same attacker. It is always useful to discover what a known attacker is doing, and this visualization can show trends of that attacker's activity on a network over time.

Although the focus of TNV is for the ID analysis task, it may also be useful in other situations when a network packet analyzer is used, such as troubleshooting network problems or to facilitate learning. We envision that an information visualization such as TNV would enable novice ID analysts to better learn their network. Because participants in our user requirements gathering reported "knowing the network" as being the most essential aspect of successful ID, a visual tool – particularly one that allows moving from a high-level overview to a very detailed display – may facilitate learning what is "normal" on a particular network. Many of the participants in our user study described their learning process as one of experimentation. They would use tools like Ethereal and Snort to learn how their networks functioned, but this was described as an arduous, time-consuming process. Gaining this understanding of normal activity is crucial in accomplishing ID, and providing a tool that links a high-level overview and network packet details could shorten this learning process.

VI. Conclusion

We have demonstrated the importance of user involvement in both providing a baseline understanding of intrusion detection work and how this understanding can facilitate the design of tools to support that work. From this understanding of ID analyst's work we developed a simple task model: monitoring, analysis, and response. Many of the currently available information visualizations designed to support ID appear most useful in the monitoring task, particularly in providing situational awareness. Instead, the TNV prototype focuses on the analysis task, providing a linked display of a high-level overview that serves as the starting point for analysis with the low-level details needed to make an accurate diagnosis.

ID analysts often begin the analysis task with a hunch or hypothesis about an event based on experience and the event itself. Because of the pervasiveness of time data and the importance of events occurring before and after a security event in providing context, TNV uses time as its central theme. TNV depicts network traffic by time intervals, organized by host IP addresses. The overview display creates a high-level picture of the network data that can easily and quickly alert an analyst to trends, patterns, and anomalies in network traffic over time. TNV has many potential uses, including facilitating the analysis task of ID and aiding novice analysts in learning what constitutes normal on their network.

Our future work will include broadening our understanding of ID through additional field-work and refining TNV to better support the analysis task. In future revisions of TNV, we will include support for dynamic filtering based on network packet attributes, improve the link display, improve the user interaction mechanisms for controlling the visualization, and provide an intermediary step in moving from high- to low-level details.

0-7803-9290-6/05/$20.00 ©2005 IEEE.

VII. Acknowledgements

Thanks to Wayne G. Lutters, Anita Komlodi, and Penny Rheingans for directing this work. This project has also benefited from the contributions of Medha Umarji, Elizabeth Chung, Chris Liang, and Nick Marangoni. It was funded in part by NSF-REU (EIA-0244131).

VIII. References

[1] "2004 E-Crime Watch Survey: Summary of Findings," *CSO Magazine/U.S. Secret Service/CERT Coordination Center*, 2004.

[2] J. McHugh, "Intrusion and Intrusion Detection," *Int'l J. of Information Security*, vol. 1, no. 1, 2001, pp. 14-35.

[3] K. Julisch and M. Dacier, "Mining Intrusion Detection Alarms for Actionable Knowledge," *Proc. ACM SIGKDD Int'l Conference Knowledge Discovery and Data Mining*, 2002, pp. 366-375.

[4] J. Allen, A. Christie, W. Fithen, J. McHugh, J. Pickel, and E. Stoner, *State of the Practice of Intrusion Detection Technologies*, Tech. Report CMU/SEI-99-TR-028, 1999.

[5] S. K. Card, J. D. Mackinlay, and B. Shneiderman, *Information Visualization: Using Vision to Think*. San Francisco, CA, USA: Morgan Kaufman Publishers, 1999.

[6] W. Yurcik, J. Barlow, K. Lakkaraju, and M. Haberman, "Two Visual Computer Network Security Monitoring Tools Incorporating Operator Interface Requirements," *ACM CHI Workshop HCI and Security Systems (HCISEC),* 2003.

[7] M. Stolze, R. Pawlitzek, and A. Wespi, "Visual Problem-Solving Support for New Event Triage in Centralized Network Security Monitoring: Challenges, Tools and Benefits," *GI-SIDAR Conf. IT-Incident Management & IT-Forensics (IMF),* 2003.

[8] R. Ball, G. A. Fink, and C. North, "Home-Centric Visualization of Network Traffic for Security Administration," *ACM Workshop Visualization and Data Mining for Computer Security (VizSEC/DMSEC),* 2004, pp. 55-64.

[9] R. A. Becker, S. G. Eick, and A. R. Wilks, "Visualizing Network Data," *IEEE Transactions on Visualization and Computer Graphics*, vol. 1, no. 1, 1995, pp. 16-28.

[10] K. C. Cox, S. G. Eick, and G. J. Wills, "Visual Data Mining: Recognizing Telephone Calling Fraud," *J. of Data Mining and Knowledge Discovery*, vol. 1, no. 2, 1997, pp. 225-231.

[11] X. Yin, W. Yurcik, M. Treaster, Y. Li, and K. Lakkaraju, "VisFlowConnect: NetFlow Visualizations of Link Relationships for Security Situational Awareness," *ACM Workshop Visualization and Data Mining for Computer Security (VizSEC/DMSEC),* 2004, pp. 26-34.

[12] L. Girardin and D. Brodbeck, "A Visual Approach for Monitoring Logs," *Proc. 12th Systems Admin. Conference (LISA),* 1998, pp. 299-308.

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

[14] K. Lakkaraju, W. Yurcik, and A. J. Lee, "NvisionIP: NetFlow Visualizations of System State for Security Situational Awareness," *ACM Workshop Visualization and Data Mining for Computer Security (VizSEC/DMSEC),* 2004, pp. 65-72.

[15] J. McPherson, K.-L. Ma, P. Krystosk, T. Bartoletti, and M. Christensen, "PortVis: A Tool for Port-Based Detection of Security Events," *ACM Workshop Visualization and Data Mining for Computer Security (VizSEC/DMSEC),* 2004, pp. 73-81.

[16] R. F. Erbacher, K. L. Walker, and D. A. Frincke, "Intrusion and Misuse Detection in Large-Scale Systems," *IEEE Computer Graphics and Applications*, vol. 22, no. 1, 2002, pp. 38-48.

[17] M. Roesch, "Snort - Lightweight Intrusion Detection for Networks," *Proc. 13th Systems Admin. Conference (LISA),* 1999, pp. 229-238.

[18] C. G. Healey, "Building a Perceptual Visualization Architecture," *Behaviour and Information Technology*, vol. 19, no. 5, 2000, pp. 349-367.

[19] C. Ahlberg and B. Shneiderman, "Visual Information Seeking: Tight Coupling of Dynamic Query Filters with Starfield Displays," *ACM Conf. Human Factors in Computing Systems (CHI),* 1994, pp. 313-317.

[20] J. D. Mackinlay, G. G. Robertson, and S. K. Card, "The Perspective Wall: Detail and Context Smoothly Integrated," *ACM Conf. Human Factors in Computing Systems (CHI),* 1991, pp. 173-179.