Home-Centric Visualization of Network Traffic for Security Administration

Robert Ball
rgb6@cs.vt.edu

Glenn A. Fink
finkga@vt.edu

Chris North
north@vt.edu

Department of Computer Science
Virginia Polytechnic Institute and State University
Blacksburg, Virginia 24061
http://infovis.cs.vt.edu/

ABSTRACT

Today's system administrators, burdened by rapidly increasing network activity, must quickly perceive the security state of their networks, but they often have only text-based tools to work with. These tools often provide no overview to help users grasp the big-picture. Our interviews with administrators have revealed that they need visualization tools; thus, we present VISUAL (Visual Information Security Utility for Administration Live), a network security visualization tool that allows users to see communication patterns between their home (or *internal*) networks and external hosts. VISUAL is part of our *Network Eye* security visualization architecture, also described in this paper.

We have designed and tested a new computer security visualization that gives a quick overview of current and recent communication patterns in the monitored network to the users. Many tools can detect and show fan-out and fan-in, but VISUAL shows network events graphically, in context. Visualization helps users comprehend the intensity of network events more intuitively than text-based tools can. VISUAL provides insight for networks with up to 2,500 home hosts and 10,000 external hosts, shows the relative activity of hosts, displays them in a constant relative position, and reveals the ports and protocols used.

Categories and Subject Descriptors

H.5.2 [**User Interfaces**]: Graphical user interfaces (GUI); K.6.5 [**Security and Protection**]: Invasive software

General Terms

Design

Keywords

Information Visualization, Security, Networks

VizSEC/DMSEC'04, October 29, 2004, Washington, DC, USA.

1. INTRODUCTION

Recent attacks (especially infamous worms like Slammer, SoBig, Blaster, MyDoom, and PhatBot, to name a few) have cost businesses an estimated $666 million in 2003, according to a recent survey of computer security executives. [13] This is arguably only a subset of the damages that can be both quantified and admitted to. The survey showed that more than 40% of the 500 executives polled said hackers have become the greatest cybersecurity threat to business and government networks, while 28% most feared internal threats such as disgruntled or recently fired employees.

In a recent address, Alan Paller, founder of the SANS Institute, cited system administrators' lack of awareness and formal training in security of as a major reason why hackers succeed. But training is costly, and maintaining awareness is difficult. We studied the activities and tools of local system administrators for over one year to determine their needs and to work with them in designing tools to allow them to work efficiently.

Regardless of training or experience, administrators must be able to rapidly understand the security state of their systems and networks, especially during a crisis. Our interviewees have indicated that most of the tools to they use are text-based (see Figure 1). We believe they can be more effective using visualization tools that take advantage of the parallel and preattentive nature of the visual-spatial cognitive modality. [23] Although several visualizations for security data exist, they either are not scalable enough, are not concrete enough, or over-process the data, reducing the user's confidence that he is looking at the real data. We are participatively designing Network Eye with our user community to provide effective visualizations for network and system security.

```
15:54:29.650359 IP (tos 0xc0, ttl 2, id 0, offset 0, flags [none], length:  48)
128.173.55.254.1985 > 224.0.0.2.1985:  [udp sum ok] HSRPv0-hello 20:  state=active
group=0 addr=128.173.48.1 hellotime=1s holdtime=4s priority=200 auth="cisco-@-@-@"
15:54:29.667909 arp who-has 128.173.54.230 tell 128.173.55.254
15:54:29.754435 IP (tos 0x0, ttl 58, id 23447, offset 0, flags [DF], length:  10 0)
128.173.37.81.42981 > 128.173.54.135.22:  P 1:49(48) ack 48 win 65535 <nop,nop ,timestamp
1011302959 14759405>
15:54:29.789216 IP (tos 0x10, ttl 64, id 15519, offset 0, flags [DF], length:  5 2)
128.173.54.135.22 > 128.173.37.81.42981:  .  [tcp sum ok] 48:48(0) ack 49 win 1 6704
<nop,nop,timestamp 14759489 1011302959>
15:54:29.790098 IP (tos 0xc0, ttl 2, id 0, offset 0, flags [none], length:  48)
128.173.55.253.1985 > 224.0.0.2.1985:  [udp sum ok] HSRPv0-hello 20:  state=stand by
group=0 addr=128.173.48.1 hellotime=1s holdtime=4s priority=100 auth="cisco-@-@-@"
15:54:29.989909 arp who-has 128.173.55.109 tell 128.173.55.253
15:54:29.995008 arp who-has 128.173.52.245 tell 128.173.55.253
```

Figure 1: Example packet trace data to visualize.

Many different techniques are currently used for analyzing network traffic. Various visualizations show individual computers, or show traffic between a single host and external systems. However, there are not currently any techniques that allow the home network or the external network to be very large, but many of the administrators we interviewed are responsible for over 100 systems. We believe our approach is the most scalable of the available concrete network visualizations.

In this paper we first describe the requirements we have elicited from our user community in section 2. Next we discuss work done by others to meet similar requirements in section 3. We then explain our tool's design in section 4. A usability study that we conducted using the tool is in section 5. Future work, including an overview of the architecture (of which the prototype tool is a part) is explained in section 6, and our conclusions are presented in section 7.

2. REQUIREMENTS

To determine the needs of our user community, we interviewed 22 professional system administrators, many of whom specialize in computer and network security. The subjects had varied backgrounds and experience levels and were from two large universities. We asked them about their duties, their experience with intrusions and other security incidents, and the tools they use. Most of the subjects we interviewed were very enthusiastic about a graphical approach to network awareness. They showed us examples of tools they used (both graphical and text-based) to help us understand their visualization needs. From these interviews, we gleaned information about the requirements for a potential visualization tool.

We formed an overall architecture (Network Eye, see section 6.2) and designed a prototype of a critical portion of this architecture. Then we conducted a small usability study (eight subjects) to test the utility of our concepts. Finally, we brought the prototype to some of the original interviewees (and a few of their associates) to get freeform feedback on the utility of our designs. This section recounts the requirements gleaned from our user study.

2.1 Who are the users?

We began to interview system administrators, naively thinking of them as a homogenous group. Instead we found at least four distinct job types in the group of subjects studied:

1. System Administrator managing server machines (42% of jobs).
2. System Administrator managing end user machines (33% of jobs reported by interviewees).
3. Security Officer (13% of jobs).
4. Network Analyst or Researcher (4% of jobs).
5. Miscellaneous: programmer, supervisor, etc. (8% of jobs).

We believe that Network Eye will support the needs of all of these jobs, especially the security officers and network analysts, whose work intrinsically involves capabilities particular to Network Eye and available nowhere else.

2.2 System Administrator Activities

We identified the following set of security-related system administration activities in the course of our investigation:

1. General Administrative Activities: Patching software, managing users, maintenance, and ambient network monitoring
2. Pure Security activities: Staying informed of exploits, forensic work, response and recovery, and directed network investigation.

Of the security-related activities listed, visual network awareness tools such as Network Eye and others can support ambient network monitoring (administration) and directed network investigation (pure security). Our findings indicate that these two activities are inherent elements of security officer and network analyst jobs. Other system adminstrators also perform these activities to a lesser extent.

System administrator activities have both proactive and reactive facets. [22] For instance, an administrator may read log files proactively, to look for and head off suspicious activity, or reactively, to find out what is happening where during a crisis. For example, computer *worms* have a definite lifecycle that can determine what is most important for a system administrator to do at a given time. Before the outbreak, administrators may be primarily in proactive mode, but during and after a major worm attack, the same administrator may be in reactive mode. Understanding how visualization tools may be used during each of these periods is important, since they typify the kinds of activity administrators will be undertaking and what their needs are.

Network administrators are interested foremost what is happening on their own network(s). They want to see the impact of the "unsafe" external Internet on the machines they manage. By visualizing communications to reveal the subjects, objects, and duration of conversations, network administrators will be able to identify patterns that may be difficult to detect by conventional methods. One of the hardest parts of securing a network is constructing an accurate mental model of what is happening so that appropriate action can be taken.

Text data is absorbed sequentially via the auditory cognitive modality, [23] as is speech. Graphical data can take advantage of the parallel nature of the visual/spatial modality. Thus we hypothesize that by visualizing packet traces, network administrators more can quickly and efficiently identify communication patterns in their networks. Currently, network administrators have to sift through large amounts of mostly text data (packet traces, log files, etc.) to gain insight into their networks. This procedure can be time-consuming and inefficient. In a moderate-sized Class B network, log files and packet traces may easily approach terabytes of information each day.

2.3 Design Approach

We use participative design as part of our development methodology, not only gleaning wishlists, anecdotes, wisdom, and experience from our users, but asking them to work with us as user-designers. As we gather this information, we modify the Network Eye architectural plan. To test specific concepts and hypotheses, we construct prototypes of pieces of the system at varying levels of fidelity. We derive our prototypes directly from designs produced in

collaboration with the interviewees. Every feature of the overall design and the prototype discussed in this paper was derived from interactions with interviewees and application of human-computer interface (HCI) knowledge and information visualization techniques. Our prototypes stimulate further design discussions with our user community.

The prototype whose design and testing is discussed in this paper is called Visual Information Security Utility for Administration Live (VISUAL). VISUAL aids network administrators by showing a graphical, home-centric overview of their network. Aside from seeing abnormal traffic, VISUAL allows network administrators to develop an accurate mental model of what is normal on their own network so that they can diagnose problems better.

3. RELATED WORK

We classify network awareness tools according to purpose (security or other), type (visual, textual, etc.), form (abstract or concrete), data (direct from the monitored network or post-processed), and perspective (the level that the observations apply to). In this section we compare VISUAL with other related work. A summary of the comparison is shown in Table 1.

Name	Purpose	Type	Form	Data	Perspective
Teoh, *et al.*	general	visual	abstract	direct	internetwork
NIVA	security	visual and other	concrete	post-processed	single network
Erbacher *et al.*	security	visual	concrete	post-processed	single or few hosts
Girardin	management	visual	abstract	direct	single network
EtherApe	general	visual	concrete	direct	single host
VISUAL	security	visual	concrete	direct	home network

Table 1: Comparison of VISUAL to related work.

Teoh, *et al.*, [20] have developed a visual IDS that allows users to interactively explore connection log data. They provide a variety of plots that enable users to view high-dimensional data and discover anomalous behavior. Their approach relies on abstract presentations of the data that may require significant expertise to interpret. According to our classification, Teoh's work is a general-purpose, visual approach, abstract presentation of direct data from an internetwork perspective.

Network Intrusion Visualization Application (NIVA) [16] is an intrusion detection data visualizer with haptic integration that allows the user to interactively detect and analyze structured attacks over time using three dimensional space. NIVA's novel haptic interface allows users to "feel" virtual objects to analyze intrusion detection data. According to our classification, NIVA is a security-purpose, visual and other approach, concrete type presentation of post-processed data from an individual network perspective.

Erbacher and Frincke [3] discuss a visualization technique they have developed for the Hummer IDS [6, 5]. The system generates an event list for visualization by processing an alert database from Hummer. Erbacher and Frinke arrange host dots in five concentric circles, where the center is the home host, and each enclosing circle matches one less octet than the previous one. They visualize the network data by drawing connection lines between host dots for traffic where line color represents the time of day. They use a set of glyphs to encode further information about connections. The authors claim the system is capable of visualizing traffic in real-time. By our classification, Erbacher and Frincke's work is a security-purpose, visual, concrete presentation of post-processed data from an individual host perspective. VISUAL differs from Erbacher, et al. in that we have a *home-centric* perspective of networks instead of a single or few computers. We are able to visually show thousands of home hosts instead of just one or a few.

Luc Girardin [7] uses self-organizing maps to show an overview of network activity. He has implemented a neural network to reduce the dimensionality of the space of network and logging information down to two-dimensional topological maps that illustrate the state of a network. Girardin's work uses foreground and background colors, sizes, and relative positions on the map display to display both network state and the deviance of current state (i.e., quantized error) from the normal. The method does not require any prior knowledge of the network, but the resulting self-organizing maps are somewhat difficult to read. We believe that though the maps were an attempt to make the abstract network data more concrete by making it visible the result is nearly as abstract as traditional charts and graphs. The approach shows great promise though. We would classify this method as a general-purpose, visual, abstract presentation of direct data from an individual network perspective.

EtherApe [2] is an open-source tool that features link layer, IP and TCP modes. EtherApe displays network activity graphically, animating host and link sizes according to traffic levels. It uses color and size (thickness) to encode information about protocol and traffic intensity. EtherApe uses Berkeley Packet Filtering (BPF) to narrow the scope of what is displayed. EtherApe is an excellent visualization for up to about 30 hosts and a moderate number of connections simultaneously. At this point and beyond, the display becomes very garbled with text labels, etc. Our classification of EtherApe is a general-purpose, visual, concrete presentation of direct data from an individual host perspective.

In addition to the above, we recommend the following two papers as helpful guides in visualizing networks and network traffic. Cheswick [1] uses a simulated spring-force algorithm to present graphs from their database. With the amount of data that they present, they show colorful tree diagrams that represent portions of the Internet. They explain how for intranets they were able to use their tool to find potential problems in routing and "leaks" where the internal network was connected to the Internet in ways that bypassed the protected network borders. Herman [8] portrays a summary of various techniques for visualizing graphs. Herman's paper surveys the many different graph presentation techniques that exist in a concise manner. Their topics range from visualizing trees, general graphs, spanning trees, to layout issues and solutions.

VISUAL's purpose is to provide more concrete visualizations that will require much less training to interpret. In addition, VISUAL is also more scalable, showing up to ap-

proximately 10,000 external hosts and 2,500 internal hosts. VISUAL is a security-purpose, visual, concrete presentation of direct data from a home-centric perspective.

4. SYSTEM DESIGN

As figure 1 shows, the type of data that we wish to visualize comes from network monitoring tools such as TCPDump [19] or Ethereal: communication packets sent between computers. Packet data includes: Source and destination IP address, source and destination port (for TCP/UDP), protocol, and time of observation.

VISUAL's design follows Shneiderman's information visualization mantra [18]: "Overview first, zoom and filter, then details on demand." We wish to display an overview of network traffic for a small to mid-size network. We show a home-centric, internal vs. external perspective of the network. VISUAL shows a representation of each home (internal) host as a small square within a larger grid that stands for the set of home hosts (see figure 2). This approach shows communications fan-in and fan-out and helps users develop an accurate mental model of network activity. A user can see which home hosts received connections from a large number of external hosts (fan-out) and which external hosts communicate with a large number of internal hosts (fan-in). Although there are many ways to detect these phenomena, we have found that a concrete visualization allows even untrained users to detect these patterns. Furthermore communications patterns are visible in the context of the total network state. Our visualization also shows relative amounts of activity among external hosts by the size of their markers.

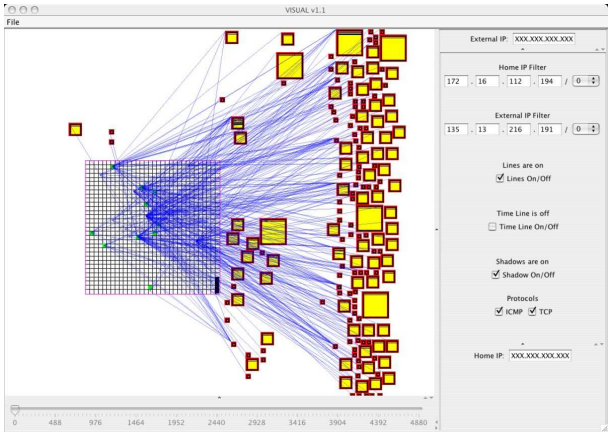


Figure 2: VISUAL displaying 80 hours of network data on a home network of 1020 hosts.

Our home-centric perspective is based on the assumption that network administrators are most concerned with the security of the hosts they manage (internal, home hosts). These hosts are predefined in a text file. After reading the list of home IP addresses, VISUAL loads and displays the network traffic.

To provide key insights to administrators about the state of their networks, we used the following list of display priorities:

1. Statically-placed markers for external hosts.
2. Spatially related grid of home hosts.
3. The external hosts that communicate with hosts on the home network.
4. Amount of traffic exchanged between each external host and the home network.
5. Protocols and ports used during the communication.
6. Replay of network communication events.

4.1 External IP Layout

The first display priority is to provide markers for external hosts (4 billion possible addresses in IP version 4) that communicate with the home network. We attempt to maintain a constant position for each external host. If a given host appears in two different data sets, we wish its screen position would be approximately the same in both plots. Therefore, our mapping function assigns every external IP address a unique virtual position in the display area. We then use techniques adapted from Keim [10] that help to display large amounts of spatially referenced data on a limited-size screen display.

In *dotted quad* notation an IP address is represented as $A.A.A.A$, where each A is called an *octet* (eight bits) and ranges from 0 to 255. Our mapping function uses the first two octets (the most significant 16 bits) to determine the X screen coordinate and the last two octets (the least significant 16 bits) as the basis for the Y coordinate. Figure 2 shows 183 external host markers (yellow squares of varying sizes with a red borders), positioned based on the host's IP address.

A drawback of this simple mapping scheme is that IP addresses that are similar (especially those that differ only in the second or last octet) map to points very close to each other. Thus, we use adaptive techniques from [17] that guarantee that no two markers will overlap each other if there is enough screen space available. If two external IPs map to the same space in the display area then the one plotted last is moved to the nearest available position. A further problem that arises from the anti-overlap algorithms is that a given IP addresses may end up being displayed in a different location depending on when it is mapped compared to other similar IP addresses. We adopt a heuristic of plotting the initial set of markers in IP address order that takes care of this problem for the time being.

The anti-overlap algorithm pushes overlapping markers down (increasing the y coordinate) unless there is not any more space on the bottom and then to the right (increasing the x coordinate and resetting the y coordinate to 0). If a marker is pushed to the bottom right corner of the display area, then the algorithm tries again at the top left corner of the display area. If there is no extra space in the display area, the algorithm scales all the markers down and repositions them.

A feature of this mapping is that external IP addresses with the same first two octets appear clustered in vertical lines. This fact may help network administrators to see subnetwork patterns. The mapping scheme takes advantage of the fact that there are likely to be empty areas due to non-uniform communication and spreads out the dense clusters. As a demonstration of scale, figure 2 shows approximately 1020 home hosts, 183 external hosts and 915 communications. In general, we believe our mapping approach is able to display approximately 10,000 external hosts.

At this time we are not giving any special treatment to nonroutable (private) IP addresses, broadcast, multicast, or any other special address classes. They will appear on the place their address maps to on the screen. Relative positioning greatly helps users because as the data changes from day to day, the user can find a given external IPs in the same relative position. Since computers do not frequently change IP addresses (at least within a tightly constrained range) our simple mapping approach will help administrators detect patterns from session to session.

4.2 Home Network Layout

The second display priority is the set of home hosts. The large square grid in the display area represents the home network. Each grid square represents a computer on the home network. The home grid is automatically positioned in an empty area of the display space. The home network may be moved or resized to avoid overlap.

4.3 Communication

The third display priority is to show which home network computers are communicating with which external computers. To facilitate this, we display lines from individual computers (grid squares) in the home network to the external host(s) they are communicating with. Each line represents a communication of one or more packets but does not indicate how much. The communication bandwidth is encoded in the size of the external host markers (see section 4.4). We use a single line for each connection is to reduce occlusion.

Line color shows the direction of traffic. For example, figure 3(a) shows an external host sending packets to many different internal computers without any reply. Communication sent from an external host to an internal host without response, is presented in red. Bidirectional communication is shown in blue. Communication that left the home network without receiving a reply, is appears in green. We consider all established TCP traffic to be bidirectional. UDP, ICMP, and most other protocols may be unidirectional unless the recipient of the traffic also responds. In the figure the red lines indicate the target hosts did not reply. However, there are a few blue lines showing some hosts did reply. Assuming the policy is not to reply to ICMP from external hosts, these few may be replying because they have been compromised.

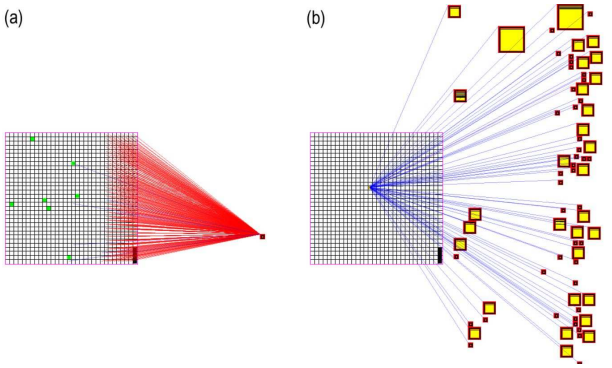


Figure 3: (a) Example of fan-in. An external computer performing a ping sweep on a subnet in the home network. (b) Example of fan-out. Many different external computers are downloading information from a server on the home network.

In the internal-external view, we summarize the internal traffic (home computers communicating with other home computers) by shading home computers green if they communicated with other home computers (see figure 2). As is the case with the lines, a green square does not represent the amount of activity, but simply that there was activity. Darker shades of green indicate greater proportional degree of internal traffic.

VISUAL emphasizes internal to external and vice-versa traffic because administrators we interviewed indicated that they generally monitor the internal-external traffic until they notice a potential security hazard. Then they turn their attention inward to locate and eradicate problems within their own bailiwick. To view internal traffic, VISUAL provides an internal-internal mode where all internal hosts are plotted twice: once within the home grid, and a second time outside it as if they were external hosts. In this view, external hosts are not plotted. In this way, we can quickly identify which internal host is responsible for which traffic.

As mentioned earlier, fan-in and fan-out are evident with our visualization. Figure 3(a) shows an external host performing a ping sweep (a method attackers use to discover what computers are on the network). As can be clearly seen from the figure, the external host systematically goes through every IP address in a particular subnet in our home network. This visual representation of an attack is easily recognizable by most users, without regard to experience in networking (see the usability study, section 5 for details).

Figure 3(b) illustrates fan-out, where several external IPs are contacting a particular computer in our home network. Although this pattern could indicate a denial-of-service attack, it happens to be a public FTP server that external hosts are accessing.

4.4 Activity Level of External Hosts

The fourth display priority is to show the rough amount of traffic each external host is responsible for during the observation time period. Thus, we size the marker of every external host in proportion to how many packets it received/sent during the time frame compared to the other computers that communicated with the home network. We elected to use a discrete scale of only three marker sizes to make differences more noticeable. We can group observed traffic levels from each host into three clusters via the K-Means clustering algorithm. The largest markers represent hosts in the cluster that contributed the largest amount of traffic (about 5 percent of the total bandwidth in the case of figure 2) while the smallest markers represent hosts in the cluster that contributed the least traffic (less than 0.001 percent in figure 2). In figure 2 there are six large markers that represent the external computers that communicated with the home network the most, while several medium-sized and small markers represent hosts that contributed less traffic proportionally.

4.5 Port Visualization

The fifth insight-generating display priority is to visualize what ports and protocols each external host uses to communicate. To show each port number on the screen space as an icon or as text for each square would clutter the screen space. Instead we display the destination and source ports as horizontal lines within the external host's marker. Each line represents a port used by that host. There are 65,535

possible TCP and UDP ports that a computer can use, so we scale this range to fit within the marker. Low-numbered ports are appear towards the top of the marker. This shows at a glance approximately how many ports an external host used and whether it was a high or low (well-known) port. We show horizontal lines for both the destination and source TCP and UDP ports used by the external hosts only. The home network computers do not display the port visualization except in internal-internal display mode. Figure 4 shows an external host that is communicating on a range of ports with many different home computers.

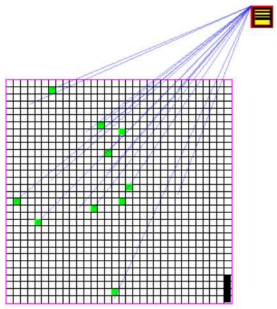


Figure 4: One external host using multiple ports.

4.6 Interactive Filtering

After seeing the overview, the user may want to focus on a few internal computers of interest. A user may select a one or more internal host grid squares (see figure 5), to see only the traffic that occurred with those particular computers, filtering out all other data. External host markers may also be selected to filter out communication lines by any other external host.

A user selects a single host by clicking on its marker. She may also selecting a range of hosts by dragging out a selection rectangle that touches all of them. The user may select a complex group of multiple hosts located in different parts of the display by clicking while pressing the *Control* key. A selected computer that is clicked again while pressing the Control key is deselected. A range of computers may be selected by holding down the Shift key and moving a selection rectangle around the desired area. Finally, the user may filter a particular set of addresses by using a Classless Internet Domain Routing (CIDR) bit mask filter. For example, if the user wanted to see only external IP addresses whose first octet was 125, then the user would set the external IP filter to 125.x.x.x /8.

4.7 Time Line

Although the size of the markers of the external hosts shows how many packets were received and sent relative to other external hosts, the size does not show *when* the packets were received relative to other packets. Instead, *all* transactions for the analysis period are shown in the overview at a single time. To see the time relationships of the communications, the user may activate the time line by selecting a checkbox on the filters pane. A slider widget beneath the main display then allows the user to view only the network activity that occurs within a particular one-second window.

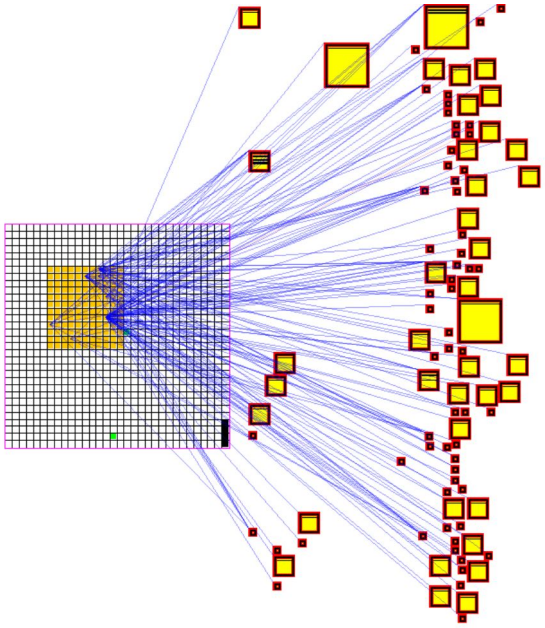


Figure 5: An example of multiple selection. All the traffic for the selected home computers is displayed. The selected home computers are represented as orange squares.

4.8 Shadows

The timeline helps the user quickly track the chronological flow of events. We use *shadows* to mitigate "change blindness." [15] As the time-slider moves, the external hosts that communicate with the home network appear and disappear. According to [21] we expect it would be difficult for the human mind to keep track of what external host markers were recently on and which were off. VISUAL helps the user remember which external hosts just communicated with the home network (within 200 seconds) by placing a light-gray shadow where the external host was. The shadows compensate for the bursty nature of Internet traffic. With shadows enabled an intermittently communicating external host would flash normal and then light-gray rather than disappearing completely (see figure 6). The default time window for displaying shadows can be changed by user preference.

4.9 Filters

Other available filters are located in the control panel on the right side of the application. There are check boxes for toggling the following features on or off:

- Lines
- Time line
- Shadows in the time line
- Different protocols (e.g., TCP, UDP, ICMP, etc.)

The user can also control the physical size of the home network grid and the sizes of the external host markers. Host markers of either kind can be reduced to a single pixel or allowed to overlap one another to avoid moving markers from their original spaces by to the overlap algorithm. We intend to implement a focus+context (a.k.a. "fish-eye") view of

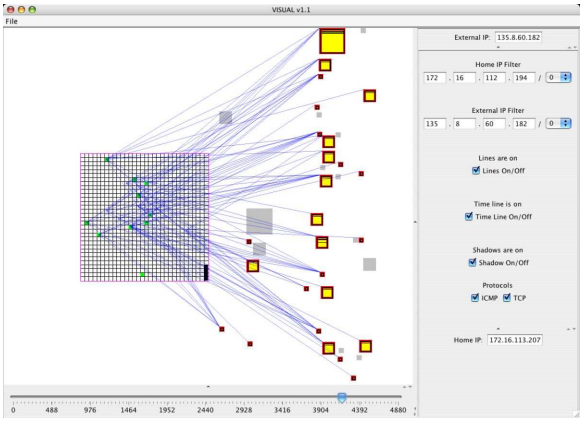


Figure 6: Time line example using shadows. The light-gray squares represent inactive external hosts that have been active in the last 200 seconds. The yellow squares represent active external hosts during the selected second.

the display area to allow the user to view more detail about individual hosts when the overlap option is active.

4.10 Details on Demand

At any time, the user can also get the following details about the computer represented by a marker by selecting it:

- The host's IP address
- The IP addresses of all the computers that the selected computer communicates with
- The TCP/UDP ports (both source and destination), the host uses to communicate
- The percentage of the overall traffic this particular computer contributes to the overall data set within the analysis time period.

5. USABILITY STUDY

We conducted a usability study to determine whether untrained users could perceive network communication patterns by looking at VISUAL's display. Since network administrators are familiar with their own networks and already have a mental model of what the communication patterns *should* look like, they can easily separate normal from abnormal. However, test users, having only a single look at a test network, cannot draw such conclusions. For our usability study, we asked users to compare two or three data sets and identify behavior that was markedly different between the datasets.

We conducted our usability study with eight graduate and undergraduate university students. During each session we had each user answer some biographical questions to determine their experience with computers and network security. The users described themselves on a range from "power users" to "occasional users" of computers. We purposely excluded network or system administrators from our study because we wanted to test untrained users. When asked about the interface, all the users characterized VISUAL as easy to use.

We explained to each user how VISUAL works then allowed them to become comfortable with VISUAL for about five minutes with a training data set that did not have any abnormalities in it. Once they were familiar with VISUAL, we presented them two testing data sets and asked them to:

1. Describe anything striking about this data set.
2. List the IP addresses of four external hosts that appear to be involved in normal (repeated, frequent in time) communication with the home network.
3. List the IP addresses of four external hosts that only communicate from time to time with the home network.
4. List the IP addresses of four home network hosts that make the largest number of connections to external hosts.
5. List the IP addresses of four external hosts that connect to the largest number of different home network hosts.

All data sets came from the same network. [12] The second data set that the users were shown contained large amounts of data compared to the first data set, but was still normal for the network. The third data set had normal network traffic except it contained a ping sweep (see figure 3(a)) and was slightly smaller than the second data set.

The users typed their responses in their own words. Every user was able to quickly find the same set of abnormalities in the data. They all used different words, but their answers agreed. In the second data set the only abnormality was that there were three servers that received most of the traffic. Although this is actually a normal trait for the network that we used, the three servers stood out as different from the first data set. In the third data set every user was able to quickly focus on a ping sweep as abnormal. Not all of the users knew what a ping sweep was, but they were still able to see that it was a different traffic pattern than the rest of the data presented.

Users were also able to quickly identify the following patterns:

- The external hosts that communicated the largest volume of traffic.
- The external hosts that communicated most frequently with the home network.
- The internal computers that communicated the most with one another
- The external hosts that communicated to the most distinct internal hosts (based on number of connections).

Users had difficulty using the timeline to identify external computers that we described as "only communicating from time to time" with the home network in the third question. Part of the problem was with how we framed the question. We did not define "from time to time" quantitatively, and some users were confused. Also, unless there was something striking about the intermittent communication, users had difficulty characterizing the communication and identifying hosts involved in it.

Another problem users experienced was that communication lines occluded some host markers in the overview window. We fixed the occlusion problem by making the lines translucent so that the host markers showed through.

During our usability study, we had not enabled the feature that allowed viewing of traffic strictly between internal hosts. Users and subsequent interviewees have expressed their desire to see this internal traffic. Subject matter experts evaluated VISUAL and determined that their investigation of potential intrusions would require an internal-only view of traffic superior to what VISUAL offered in the study.

One user said VISUAL made it, "easy to make sense of data and see general trends." This user said VISUAL would be less usable for "fine-grained" view of network traffic data. This observation fits our intentions well for VISUAL at this stage and underscores the importance of providing drill down to the packet level for analysis purposes in the future.

We feel that our usability test successfully demonstrated that a traffic pattern visualizer such as VISUAL can provide insight into network traffic data without requiring any training. The participants also provided many useful comments that we will use to improve VISUAL and work based on it in the future.

6. FUTURE WORK

VISUAL is an proof-of-concept program that has allowed us to test the premise that visualizations of network traffic data help users rapidly form accurate mental models of network events with little or no training. VISUAL is actually a prototype for part of our end-to-end security monitoring application concept we call Network Eye.

6.1 VISUAL's Future

VISUAL currently relies on a preprocessor to digest network packet traces. We accepted this limitation early in development to save time, but it has always been our intention to allow VISUAL to accept tcpdump [19] format packet traces and the like. VISUAL would be much more useful to system administrators as a real-time network traffic visualizer.

We believe that the view of purely internal traffic VISUAL provides is a good approach since it uses the same presentation that our usability study confirmed as good for internal–external traffic. However, this mode and the interactions that allow switching between modes must be tested. We envision doing an expert review and another usability test with this feature.

A user-suggested idea for the timeline is to change the alpha channel for external host markers so that they are fully opaque when active and slowly fade away (increasing transparency) until they disappear over time similar to [14]. It would also be helpful to place a histogram over the timeline's scale to show the distribution of the relative amount of activity at each moment as in [11]. This feature would enable users to more rapidly locate areas of interest when viewing in timeline mode.

Currently, VISUAL is useful for small networks of fewer than about 12,500 nodes (of which 2,500 are internal). We believe the concept could be scaled to larger networks of perhaps a few hundred thousand hosts. One technique that may be useful is shrinking the host markers to single pixels and implementing a distorted lens (fisheye) to view and select individual computer markers. Our forthcoming project, Network Eye, is designed to explore larger-scale visualization techniques.

6.2 Network Eye Overview

We are currently completing Network Eye's architecture definition stage. The final architecture has been derived from interviewing system administrators, testing existing security awareness tools, and researching visualization techniques. Our architecture consists of two tightly integrated visualizations: the Network Communication Visualization (which VISUAL is a prototype of) and the Host Resource Visualization, that together provide an end-to-end view of applications, computers, and networks in context (see Figure 7).

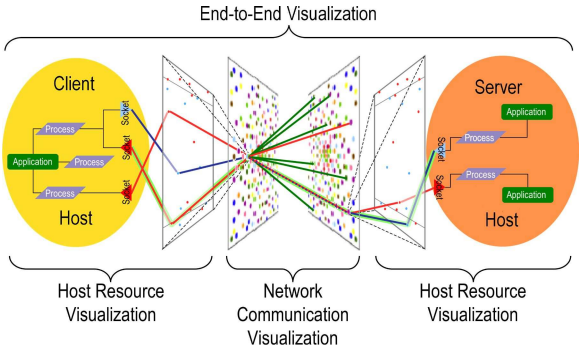


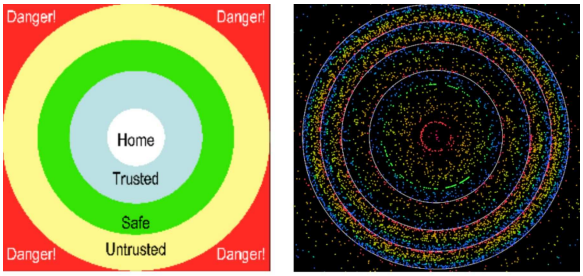
Figure 7: Network Eye's end-to-end view of application communicating across the network.

The first visualization, the network communication diagram, displays network data (packet traces, etc.) gathered from passive listeners on the network. The second visualization, the host-application resource map, provides a view of host data (processes, sockets, ports, and files in use, etc.). The two are joined to give an integrated, end-to-end system security picture that can show detailed information about both ends of a TCP connection simultaneously.

6.2.1 *The Network Communication Visualization*

The network communication visualization's high-density, three-dimensional scene is made of two network pixel maps facing each other as if in a mirror. A network pixel map is a pixel-oriented overview visualization [9] that can display up to about 100,000 host markers in a single 1,000 by 1,000 pixel window (Figure 8). We will use focus+context (fish-eye distortion) to enable selecting a single host from the myriads displayed. Markers are colored in spectral order from red to violet where red indicates lower-values (near 0.0.0.0) and violet indicates higher values (near 255.255.255.255). We will use pulsating illumination to indicate recent activity of hosts in this view.

Marker (host) arrangement is critical to user insight in high-density displays as noted in [9]. Each marker (which may be as small as a single pixel) represents a unique IP address observed in the network traffic. Marker placement indicates the trust level and IP address of the host. While VISUAL placed markers into two trust groups, *us* and *them*, the network pixel map will accommodate up to five trust levels: Home, Trusted, Safe, Unknown, and Danger. Another way to think of trust is organizational proximity. These lev-



(a) (b)

Figure 8: The network pixel map: (a) The layout of trust levels, (b) Example network pixel map showing 8,513 IP addresses observed from a single host.

els are arranged, target-style, with the most trusted hosts' markers (Home) placed in the center ring and the rest of the markers placed further from the center as trust level decreases. Trust levels may be assigned from network configuration information and via experience. Dangerous hosts will be assigned from local IP blacklists or can be moved to the outskirts as they are discovered.

This arrangement of markers causes groups of hosts that are equally trusted and in the same class B networks to appear in rings. In Figure 8, the broken green ring near the center is the set of home hosts in the local network.

The true power of the network pixel map becomes clearer when two of them are placed in 3-dimensional space facing each other and lines representing communications are drawn between them. This visualization, called the *network communication visualization* shows network traffic flow from the TCP/IP client, on the left, to the server, on the right (see Figure 9). As with VISUAL, elementary communication patterns such as point-to-point interconnections, and fan-out/fan-in become readily apparent in this view.

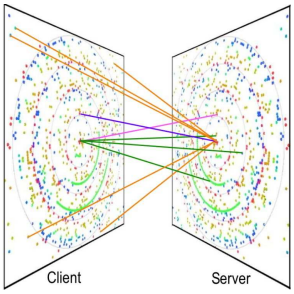


Figure 9: An illustration of the network communication visualization, a high-density, 3D scene depicting connections between clients and servers.

We can allow the user to encode meaning into line thickness, pattern, color, pulsation, and luminosity in a variety of ways as desired. To assist the users in making sense of the tremendous amount of data, we will provide three kinds of filtering tools: (1) selection via pointing and clicking, (2) graphical and textual packet filtering based on Berkeley Packet Filters (BPF), and (3) rate-based filtering for selecting according to the level of traffic experienced. We will provide tools to zoom in, filter out distracting features, and drill down to the packet-level on demand. By filtering out typical traffic, analysts may identify unusual flows as highly likely intrusion paths.

Another sense-making strategy is aggregation of markers and communication lines. We can reduce the complexity of the picture dramatically by aggregating the host markers according to the amount of traffic they generate, thus only showing the "heavy hitters" as Estan, Savage, and Varghese do in their AutoFocus tool. [4] We are collaborating with Dr. Estan to investigating effective ways to visually aggregate host markers and flows.

6.2.2 *The Host Resource Visualization*

The second type of visualization in the Network Eye design displays host processes and their communication resources. It meshes closely with the network communication visualization. The host resource visualization shows the interaction of applications in the overall network (see Figure 10). An analyst may spot some communication patterns of interest, select a communication link, and drill down to find out what hosts, ports, sockets, processes, applications, and users are responsible. Host drill-down enables quick investigation and reaction to potential intrusions.

An application can be seen as a set of processes that use sockets and other resources to operate. The sockets are bound to some port, and these ports may be communicating with remote hosts. The ability to trace a port back to its owning application greatly helps in determining why the communication is happening. Both client-server and peer-to-peer relationships can be discovered in this way, regardless of encryption.

Obtaining detailed information from a host requires a tattletale client to be installed there. This is reasonable since remote administration tools are common. Network Eye's tattletale client will use secure shell authentication and encryption for security. Our users requested that the host-application resource visualization also be able to show files related to each application to make it easier to locate and eradicate unwanted or malicious programs.

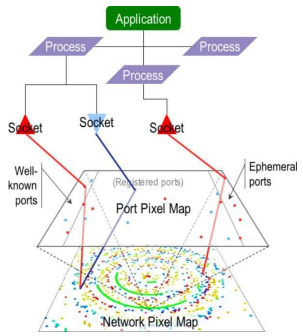


Figure 10: Host-application resource map showing an application tied into a network pixel map.

7. CONCLUSIONS

In this paper we discussed our contributions to network security visualization which include the following:

- A new visualization of network traffic that enables rapid perception.

- Usability results showing how the tool helps untrained users understand network state.
- Network visualization scalability to over 10,000 nodes and plans for up to 100,000 nodes.
- Temporal visualization of traffic.

Network data analysis is a very important task from an administrator's point of view. A significant amount of time and manpower is devoted to sift through text-only log files and packet traces generated by tools used to secure networks. VISUAL has demonstrated that visualization considerably reduces the time and training required for data analysis of network traffic and at the same time provides insights which might otherwise be missed during textual analysis.

8. ADDITIONAL AUTHORS

Additional authors: Anand Rathi and Sumit Shah (Virginia Tech, email: `arathi,sshah@vt.edu`) helped implement parts of VISUAL. Ricardo Correa (University of Texas at El Paso, email: `rcorrea@utep.edu`) conducted several system administrator interviews.

9. REFERENCES

[1] B. Cheswick, H. Burch, and S. Branigan. Mapping and visualizing the internet. In *Proceedings of the 2000 USENIX Annual Technical Conference*, pages 1–12. USENIX Assoc., 2000.

[2] J. T. Cota. Implementacion de un monitor y analizador grafico de red en el entorno gnome, July 2001.

[3] R. F. Erbacher. Intrusion behavior detection through visualization. In *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, pages 2507–2513. IEEE, IEEE Computer Society, 2003.

[4] C. Estan, S. Savage, and G. Varghese. Automatically inferring patterns of resource consumption in network traffic. In *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 137–148. ACM Press, New York, NY, USA, 2003.

[5] D. Frincke. Balacing cooperation and risk in intrusion detection. *ACM Transactions on Information and System Security (TISSEC)*, 3(1):1–29, February 2000.

[6] D. A. Frincke, D. Tobin, J. C. McConnell, J. Marconi, and D. Polla. A framework for cooperative intrusion detection. In *Proc. 21st NIST-NCSC National Information Systems Security Conference*, pages 361–373. NIST, 1998.

[7] L. Girardin. An eye on network intruder-administrator shootouts. In *Proceedings of the 1st USENIX Workshop on Intrusion Detection and Network Monitoring*, pages 1–12. USENIX Assoc., 1999.

[8] I. Herman, G. Melancon, and M. S. Marshall. Graph visualization and navigation in information visualization: a survey. *IEEE Transactions on Visualization and Computer Graphics*, 6(1):24–43, 2000.

[9] D. A. Keim. Designing pixel-oriented visualization techniques: theory and applications. *Visualization and Computer Graphics, IEEE Transactions on*, 6(1):59–78, 2000.

[10] D. A. Keim and A. Herrmann. The gridfit algorithm: An efficient and effective approach to visualizing large amounts of spatial data. In *Proceedings of the Conference on Visualization '98*, pages 181–188. IEEE Visualization, 1998.

[11] Q. Li and C. North. Empirical comparison of dynamic query sliders and brushing histograms. In *Proceedings of the IEEE Symposium on Information Visualization 2003*, pages 147–153. IEEE Computer Society, 2003.

[12] R. Lippmann, J. Haines, D. J. Fried, J. Korba, and K. Das. The 1999 darpa off-line intrusion detection evaluation. *Computer Networks*, 34(4):579–595, October 2000.

[13] D. McGuire. Study: Online crime costs rising, http://www.washingtonpost.com/wp-dyn/articles/a53042-52004may53024.html, 2004.

[14] C. North, U. Farooq, and D. Akhter. Datawear: Revealing trends of dynamic data in visualizations. In *LBHT Proc. IEEE Symposium on InfoVis 2001*, pages 8–11. IEEE, IEEE computer Society, October 2001.

[15] L. Nowell, E. Hetzler, and T. Tanasse. Change blindness in information visualization: A case study. In *Proceedings of 2001 Information Visualization*, pages 15–22. IEEE Computer Society, 2001.

[16] K. Nyarko, T. Capers, C. Scott, and K. Ladeji-Osias. Network intrusion visualization with niva, an instrusion detection visual analyzer with haptic integration. In *Proceedings of the 15th annual ACM symposium on User interface software and technology*, pages 277–284. IEEE, Palgrave Macmillan, March 2002.

[17] G. Robertson, M. Czerwinski, K. Larson, D. C. Robbins, D. Thiel, and M. van Dantzich. Data mountain: Using spatial memory for document management. In *Proceedings of the 11th annual ACM symposium on User interface software and technology*, pages 153–162. ACM, ACM Press, 1998.

[18] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *Proceedings of the IEEE Symposium on Visual Languages '96*, pages 336–343. IEEE, IEEE Computer Society, 1996.

[19] Tcpdump public repository, June 2004.

[20] S. T. Teoh, K.-L. Ma, and S. F. Wu. A visual exploration process for the analysis of internet routing data. In *Proceedings of the IEEE Conference on Visualization 2003*, pages 523–530. IEEE Computer Society, 2003.

[21] B. Tversky. Distortions in cognitive maps. *Geoforum*, 23(2):131–138, 1992.

[22] H. Venter and J. Eloff. A taxonomy for information security technologies. *Computers and Security*, 22:299–307, May 2003.

[23] C. Wickens, D. Sandry, and M. Vidulich. Compatibility and resource competition between modalities of input, central processing, and output. *Human Factors*, 25(2):227–248, 1983.