Visualizing Network Security Events Using Compound Glyphs from a Service-Oriented Perspective

J. Pearlman and P. Rheingans

Abstract Network security is the complicated field of controlling access within a computer network. One of the difficulties in network security is detecting the presence, severity, and type of a network attack. Knowledge of such an attack is used to mitigate its damage and prevent such attacks from occurring in the future. We present a new visualization of a computer network for security purposes by approaching the problem from a service-oriented perspective. This approach involves a node graph visualization where each node is represented as a compound glyph, which gives details about the network activity for the specific node based upon its service usage. Furthermore, we visualize temporal activity using time slicing techniques in the compound glyph to give more details about the network and allow interactive controls for an administrator to actively monitor a network in order to react to security events quickly. Our resulting visualizations of networks successfully identified and described denial of service (DoS) and compromised network attacks.

1 Introduction

Network security is crucial to maintaining stable networks in order for institutions to continue normal operations. Network attacks are designed to cripple or disable normal functionality of a network, interrupting normal operations. A network administrator's primary task is to enable secure and legitimate communications between machines on a network. A large portion of this task involves both reactive and proactive prevention of attacks. The administrator is concerned with any type of anomaly that could represent an attack or an intrusion. Furthermore,

J. Pearlman and P. Rheingans
University of Maryland Baltimore County, Baltimore, MD 21250, USA, e-mail: jpearl1@cs.umbc.edu, rheingan@cs.umbc.edu

the administrator must recognize signature-based network threats, as most attacks follow some pattern. The network administrator performs three tasks: monitoring, analysis, and response (Komlodi et al., 2004). In the first phase, *monitoring*, the administrator attempts to find something problematic about the network, such as an attack or unauthorized access. Once a problem is found, the administrator will *analyze* the specifics of the problem in order to *respond* by taking steps to correct the issue and prevent the problem from occurring again. Network security visualization aids the network administrator in the first two generalized tasks, monitoring and analyzing. The more specific tasks that a network security administrator must perform in the monitoring and analysis stages include detecting insecurities, detecting intrusion attempts, defending against network attacks, and detecting resource misuse.

The primary goal of network security visualization is to provide a network administrator with visual information that allows the administrator to perform their job-related tasks, including identifying and preventing unauthorized access to resources, attacks on their network, and misuse of resources from within the network. One of the difficulties of this task is handling large amounts of data and filtering the data in such a way that security events stand out. Another difficulty is enabling the visualization to show data for individual nodes while showing data for the entire network to better detect and understand security events. Most of the current network security visualization techniques focus on one of these areas, either displaying data for only one node on a network, or displaying overall network data without going into detail on the particular nodes. Without an understanding of the node's traffic in its significance within the overall network traffic, certain types of attacks are difficult to detect.

The specific problem we address is visually aiding a network system administrator in identifying security events on their network from a service oriented perspective. The major goal of this research is to provide a visual means for a network administrator to take steps to prevent attacks, mitigate damage from attacks, and monitor service traffic. The features that a visualization tool must provide to perform this task is to visually identify anomalous behavior in a network and allow the administrator to gain information about the anomalous behavior. The visualization tool must provide service information about each node on the network to detect anomalous behavior at the service level. Finally, the visualization tool should be able to use temporal data to distinguish between heavy usage and attacks. This visualization will aid the administrator in identifying the presence, severity, and type of network security even present in a network by representing network data at the service or application layer.

This research will visually identify service activity on a per node basis with an emphasis on anomalous service activity. Any inbound network activity occurring on a service that the administrator is unaware of is a cause for concern. Particular services, such as Internet relay chat (IRC), are commonly used for outbound traffic when Trojans or worms are present on a network. Also, common service activity from a set of machines to a machine on another network can lead to the discovery of an attacker's control points. This research will provide a real time system for network administrators to monitor service activity, allowing for early detection of attacks, such as a denial of service (DoS) attack.

We present a new approach to network security visualization by extending existing approaches in order add service and temporal information into the node itself. We begin with a node scatter plot, which is similar to other approaches based upon network traffic data sets. Within each node in the scatter plot, we embed more information than previous approaches by using time slicing and service differentiation visualization techniques. By visualizing different service activity over time on a per-node basis, we are able to differentiate between attacks, discover more details about the attack, and identify different types of attacks not available in previous scatter plot node graph visualizations of network data.

2 Related Work

A cluster of previous research represents the structure of the network as a graph of links and nodes. Becker performed research on visualizing node links; this research is frequently cited because most visualizations contain some type of node link information (Becker et al., 1995). The vertices in the plot represent machines, and the edges in the plot represent network connections. Ball introduced a method that reduced the clutter by focusing in on nodes for a specific network and called this a home-centric approach (Ball et al., 2004). Teoh's work took node links to a higher level of detail, by using focus + context techniques to display time data in addition to node link data (Teoh et al., 2004). In another application of this technique, Goldring experimented with using scatter plots for various visualizations, including network link traffic (Goldring, 2004). Building upon the common node link scatter plot, Erbacher used glyphs instead of dots as nodes, and glyphs instead of lines for links, in order to add more information to the visualization (Erbacher, 2002). Ball uses size to represent amount of traffic and opacity to represent inactivity time (Ball et al., 2004). These explicit approaches have a tendency to be overwhelmed by very large network sizes.

Other previous research uses other information visualization techniques to show the multivariate nature of network data. Conti used parallel coordinates to plot external port, internal port, source address, and destination address (Conti and Abdullah, 2004). Yin et al. (2004) used NetFlow data, a format that logs network data transferred from end to end by ignoring intermediate communications, to visualize a network. McPherson et al. (2004) used a color mapping technique to identify interesting ports on a network. Papadopoulos et al. (2004)'s Cyberseer is a unique combination technique using visual and auditory techniques for network monitoring. Girardin (1999) used a self-organizing map to visualize port activity over time. NVisionIP (Lakkaraju et al., 2005) uses a galaxy view as its main view but then allows other more focused views on selected areas of interest.

3 Technical Approach

The method used to solve the problem of identifying anomalous network nodes to determine attacks is a glyph-based visualization technique similar to scatter plots of network traffic developed in previous work. Our method adds more information to the glyphs in order to better identify anomalous behavior by the service activity. In order to identify vulnerable nodes and network attacks, more information than general network activity must be used, such as service information. The service activity information can differentiate between normal usage and potential attacks.

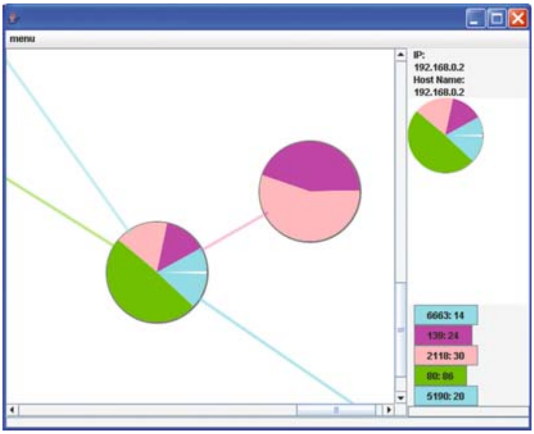
In order to detect DoS attacks and compromised networks, it is important to distinguish traffic based on service type and time. Our method combines several techniques to accomplish the goal of determining the presence, severity, and type of a network attack. Like in Ball's approach, glyph representations and differentiation between managed and unmanaged nodes are techniques used in this method and like in Girardin's technique, we break down difference service activity visually. However, service activity is displayed within each node in order to give more detail about network attacks. Furthermore, temporal data is displayed in a static manner to allow for more analysis on an attack. Finally, adjusting opacity on a per glyph basis is used to compare the network to normal network conditions for detection of anomalies.

3.1 Network Node Glyph

This method maps port information to a glyph representing a node on the managed network. Each open port, or service running on a machine, exposes a potential point of entry, authorized or unauthorized. Each glyph, representing a node on the network, represents the presence and amount of activity for a particular service. Each glyph is a compound representation of services and their activities, with each region of the glyph representing the amount of activity on an open service of a machine. The size of the glyph represents the total amount of activity on the node, while the regions identify what percentage of that total activity belongs to a particular service. The size is scaled with amount of activity, measured by number of packets, with a relative maximum size. The glyph contains a representation color for each different service but reuses colors for services because there are more services than visually distinguishable colors (approximately 65,000 different services). Following Ball's home-centric approach, managed nodes on the network will be rendered differently from unmanaged nodes outside the network (Ball et al., 2004). Finally, node activity links exist between managed nodes to other managed nodes, and unmanaged nodes to managed nodes.

Service mapping. The compound glyph representation is a pie chart. Each service is a region of the pie chart, with its size representing the amount of activity on that service and its color differentiating it from other services. The simple pie chart representation adapts well to adding temporal data. When raw traffic is seen on the wire, several attributes are determined from the network packet including source

Fig. 1 Service mapping of a node. Mouse over highlighting provides a key which details the ports of each service region. Host 192.168.0.2 is performing several different activities during this visualization, including web traffic, chat traffic and file sharing traffic to another managed node



IP:
192.168.0.2
Host Name:
192.168.0.2

6663: 14
139: 24
2118: 30
80: 86
5190: 20

address, destination address, and the type of protocol. The protocol can often be determined by the destination port of the initial packet that establishes a connection (the SYN packet in TCP). However, ports can be customized so looking at the actual data in the packet can more accurately determine the protocol. The protocol then determines the service. In each node representation glyph, an arc is created for each service and filled with a color mapping to that service. Figure 1 shows two managed nodes which are divided up into several services. The managed node 192.168.0.2 is highlighted to show the key which displays ports for each service represented by a colored region. Notice that both nodes in this visualization are participating in various different service activity, and from the key of the highlighted node, we can see that node 192.168.0.2 is conducting approximately two-thirds of its traffic as web traffic. It is also using the instant messenger service (port 5190) and windows rpc service (port 139) in lesser amounts than the web service.

Managed and unmanaged nodes. Managed nodes will be visually larger than the majority of unmanaged nodes barring relatively large amounts of activity occurring on an unmanaged node. Furthermore, unmanaged node traffic that does not have one endpoint at a managed node will not be visualized. Unmanaged node traffic (both endpoints are unmanaged nodes) is rarely useful in determining a network attack and is extremely vast, since this effectively means all Internet traffic! Also, having access to capture all traffic on the Internet poses a different problem. By eliminating unmanaged node traffic, the administrator can focus on managed nodes and their communications. Figure 2 is an example of one managed node connecting to three different unmanaged nodes. Each unmanaged node happens to be a web server, the one highlighted by the key being the ACM portal. In this visualization, the larger glyph, representing a managed node, has slightly more than half of its activity occupied by web traffic to three different web sites.

Temporal activity. Another goal of the visualization is to show the change of the amount of service activity over time. Reoccurring network activity on a predictable schedule is an indication of a worm or trojan on a machine and a stronger indication when the reoccurring network activity is of similar amount and using the

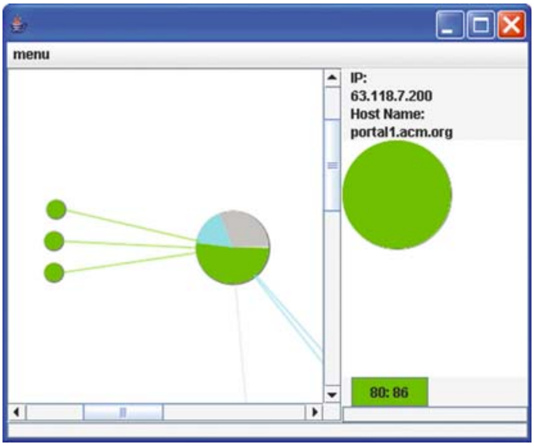


Fig. 2 Managed nodes vs. unmanaged nodes. Managed nodes are represented by larger glyphs and placed in a centralized location of the visualization. This figure shows a managed node connection to three different web sites, one of them being the ACM portal, which is highlighted by a mouse event

same service. Shanbhag et al. (2005) used visual time slices to display changing attributes over time. They present several different methods including rings, slices, and wedges to display temporal data of a region. In order to display the change in activity for a particular service, this technique is applied to each region of the glyph that represents the activity of a particular service. The size of the region represents the amount of activity for the service, the color of the region distinguishes the different services, and temporal slicing is used within the region to show when network activity occurred on the service represented by the region. Ring based temporal slices are used, with the most outer ring representing the most recent time slice. Visually, the slicing is trivial to display by drawing the outer pie chart first and drawing inner pie charts on top of the previous one. The size of each time slice is determined by subtracting a variable step size from the size of the previous radius of the time slice, which eventually hits zero when too many time slices are created. The amount of time each slice represents is configured by the user. However, since the outer time slice represents the most recent data, the oldest time slices will disappear first. Figure 3 shows an example of a display of a node cut into four time slices. In the oldest time slice which is the center of the circle, only two types of traffic are present: IRC represented by yellow, and instant messenger chat represented by cyan. In the second oldest time slice, different service activity occurs including windows file sharing activity represented by pink, email traffic represented by gray, and web traffic represented by green. In the second most recent time slice, only web and instant messenger traffic is present. In the most recent time slice or the most outer ring or the glyph, more than 75% instant messenger traffic occurs with a little web browsing traffic and some IRC chat traffic.

3.2 Layout

The layout of the nodes follows a trivial formula and is not a primary focus of this visualization. The (x, y) coordinates of the node on the two-dimensional plane are

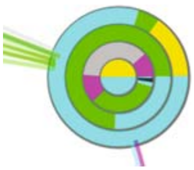
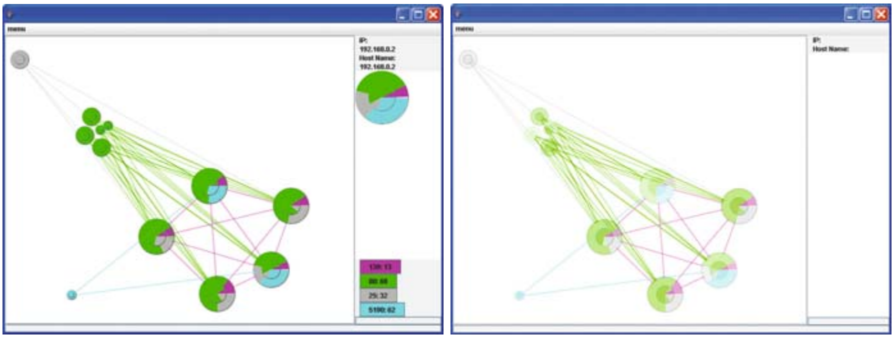


Fig. 3 A glyph representation of a managed node on a network. This node is sliced into four time slices, each representing a different amount and type of service activity



IP:
192.168.0.2
Host Name:
192.168.0.2

139:13
80:48
25:32
5190:62

IP:
Host Name:

(a) **(b)**

Fig. 4 A visualization of a small normal network. Notice that each of the web traffic nodes (*green*), representing web sites the managed nodes are visiting are grouped in the same area. One time slice has been performed on this visualization among all nodes. (**a**) Without comparing to a simple model. (**b**) Comparing to a simple model that detects abnormal behavior. Nearly all nodes are faded because our simple model doesn't detect very anomalous behavior

calculated based upon the type of activity occurring within that node. For managed nodes, their location starts near the center of the viewing area and randomly moved slightly from the center to create some separation from other managed nodes. For unmanaged nodes, the service with the largest amount of traffic is found and used to position the node along the *x*-axis and slight randomization is added to prevent excessive overlap. By performing this positioning, unmanaged nodes with similar characteristics are grouped together, allowing the user to treat such groups as larger entities if needed. In Fig. 4, notice that all of the unmanaged nodes producing web traffic are grouped together. Finally, we allow the user to manually reposition nodes as well.

3.3 Comparing to a Model

Because all networks are different, it is difficult to generically detect normalcy in a network. Because of this, an interface to a model for a network was created to have a visual method for comparing a network to a "normal" network. A "normal" network can be defined by a custom model which follows a limited interface, which includes functionality for rating the abnormality of a node in the actively monitored network. A custom network model is used for this feature because every network has a different definition of normal activity. Using a direct relationship to the anomaly factor given by the network model, the visualization modifies the alpha value of the node. This technique will allow the administrator to fade out nodes that are not considered abnormal by the network model.

The model interface requires the model to return an anomaly factor for a specific node. The node's graphical representation will be altered by opacity based upon the model's anomaly factor for the node. As an example, we use an overly simplified model to demonstrate this visualization technique. The sample model works by starting with an anomaly factor of 0.3 on a zero to one scale. The model is based on college dorm traffic just like the simulation. Activity that we consider normal, such as web and instant messaging traffic, lowers the anomaly factor. Unknown service traffic raises the anomaly factor and ICMP traffic raises the anomaly factor by an even larger step. ICMP traffic is generally used for obtaining detailed information about a network but our sample network only has one administrator and does not expect to see ICMP traffic initiated by other sources. Furthermore, if there is traffic to a select group of common web sites, the anomaly factor is lowered slightly and otherwise raised slightly. A simple host based check raises the anomaly factor when having traffic to or from sites without a common suffix like .com, .net, or .edu. In Fig. 4 we apply the comparison of a normal network to our simple model. Because this visualization is of a normal network, our simple model detects the nodes in the network as fairly normal and therefore fades them out in the right image which is compared to a model.

3.4 Results

These methods were applied to a simulated network consisting of a small set of client users representing college student's dorm computers with some added network servers. Most of the common types of attacks were identifiable by applying these methods and creating a visualization of the network under attack. More specifically, details of the attack can be determined from the visualization down to which computer and which service is under attack. Session hijacking and man in the middle attacks will not be visible using these methods as that different style of attack would require a different visualization approach. This visualization deals with endpoint to endpoint traffic which will not identify changes along the route which occurs in man in the middle attacks. Furthermore, in order to detect session hijacking, unmanaged node to unmanaged traffic must be visualized, which creates large scalability issues.

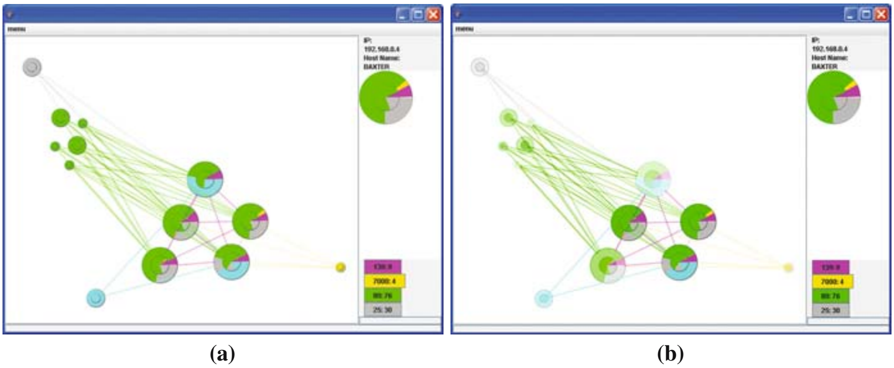
Network packets are captured at each machine on the network and stored in some pcap (packet capture) format. Each packet consists of TCP/IP header information, containing meta information such as the source and destination of the packet. In addition to the source and destination address, the header contains the source and destination port. The source and destination address allows for mapping of activity on a node to node basis. The destination port allows for mapping the availability of services on a node in addition to the amount of activity for a particular service on a node. This application is capable of real time network monitoring and therefore needs the capability to sniff network traffic in real time. JPcap (Java API to the pcap library) is used for real time packet sniffing and creating simulated network packets (Charles, 2001).

3.4.1 Compromised Network

Figure 5 is an example of an IRC trojan which has infected several machines on the managed network. IRC traffic is mapped to a yellow color and covers ports 6667–7000, which are the typical set of ports used for IRC. The small yellow sections indicate a small amount of IRC traffic coming from the managed nodes, in each of the time slices. The small yellow node is the control center the attacker uses to control the IRC trojans. Trojans are difficult to detect from a network administrator view who is not familiar with the network. Certain trojans use fairly uncommon services, for example, Backdoor.IRC.Snyd.A uses IRC as its command protocol. When comparing the infected network to a simple network model, normal traffic can be faded, highlighting potential risks. A very basic model, which does not consider IRC common traffic, can highlight infected nodes in Fig. 5. Even if IRC was commonly used by several users on the managed network, if the administrator is familiar with seeing such traffic, the added IRC traffic when other nodes are infected will appear anomalous. Finally, if the model is created or adapted to consider IRC traffic from particular IRC users to be normal, and even consider the set of IRC servers normally used to be normal, then comparing to the model will still effectively highlight the infected nodes.

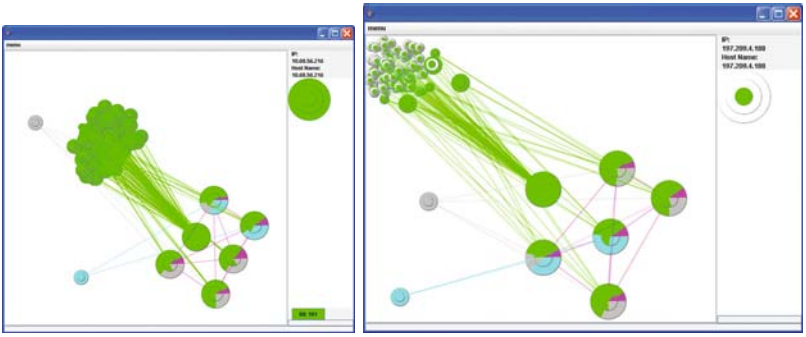
3.4.2 Denial of Service Attacks

Figure 6a is an example of a network receiving an application level distributed DoS attack. This example is a web(port 80) distributed DoS attack. The attackers are



(a) (b)

Fig. 5 A visualization of network infected by an IRC trojan. (**a**) Without comparing to a simple model. (**b**) Comparing to a simple model that detects abnormal behavior. Notice that the nodes emitting abnormal network behavior are darker



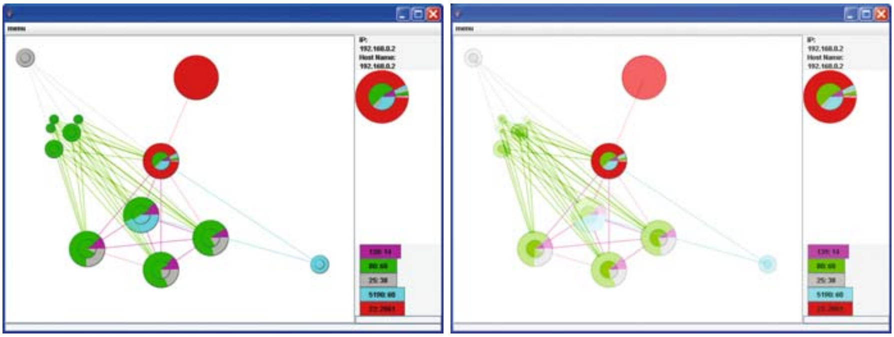
(a) (b)

Fig. 6 Network undergoing heavy web traffic on its web server (**a**) This represents a Distributed DoS attack. Notice that in each timeslice, web traffic is still occurring from each of the nodes contacting the webserver represented by the large green glyph near the center. (**b**) This represents large amounts of web traffic from approximately 100 different clients in the time span of the visualization. Time slicing is used to show that the majority of these clients have activity in a single time slice and are not repeatedly creating connections like in a DoS attack

represented by the large cluster of green colored nodes all connecting to the same managed node. The larger size of the attacking nodes indicates a larger amount of network traffic. A comparison is difficult to find in this network because nearly all represented nodes experience high traffic volume. However, in the northwest region of the web traffic nodes cluster, there is a smaller traffic node in which only about half of the glyph is visible peeking out on the left side. This likely represents normal usage or a weak attacker in the DoS attack.

Time slicing is performed among the web client (or attacking) nodes to show that traffic occurs in each slice. By using time slicing, it makes it more apparent that this is a DoS attack and not a case of large legitimate web traffic. Small time slice intervals will improve the effectiveness of distinguishing between heavy usage and a DoS attack. Notice that web traffic occurs in every time slice of the attacking node, indicating constant web traffic over an extended period of time. Normal usage would have a colored inner ring, indicating the initial connection, but would likely see the traffic trail off toward the outer rings. The number of time slices colored would be proportional to the amount of time the client continues to browse the same web server. Of course, it is extremely unlikely that all clients continue to browse the same web server for even a small amount of time.

Figure 6b is an example of a network receiving large amounts of application level network traffic, but not necessarily a distributed DoS attack. In this network visualization, many web client connections are made to the web server (represented by the cluster of green nodes), which happens in a DoS attack. However, using time slicing, this visualization shows that most of the nodes do not continuously communicate with the web server. Most nodes are represented by a green inner circle, representing an initial connection and traffic to the web server, and clear outer rings. This indicates that after the initial connection and traffic occurs, the user no



(**a**) (**b**)

Fig. 7 A visualization of network under attack by an application level DoS attack against the SSH service. (**a**) Without comparing to a simple model. (**b**) Comparing to a simple model that detects abnormal behavior. Notice that only the nodes involved in the SSH DoS attack still have a strong opacity

longer browses the managed web server in this visualization. This is more typical of normal web usage than Fig. 6a as the user would likely come to a web server, obtain what they need, and move onto to another web server or stop browsing all together.

Figure 7 is an example of a network receiving an application level DoS attack against the ssh service. The large red node indicates the attacker and is large because of the amount of traffic present. Under normal network conditions, no node will become larger than a managed node. Baseline network traffic is also present in this visualization of normal college student traffic. Analyzing the managed node under attack, normal traffic patterns are present in the initial time slice (center) with a mix of web, chat, and file sharing traffic. In the most recent time slice (outer), normal traffic is present but heavily overcome by SSH traffic as indicated by the strong red color presence. In this particular network, the managed node is a normal network user with the SSH service open to allow remote logins to that machine. However, that service is under a DoS attack in this visualization, which is likely inhibiting that user's ability to conduct normal network traffic and certainly inhibiting the use of the SSH remote login service. The SSH DoS attack network visualization is also compared to a normal usage network model and is easily able to identify the managed node under attack.

Figure 8 shows one of the managed nodes in the college student network getting ping flooded. Note that this visualization does not look much different from the SSH application level DoS attack. Underneath, the attack is similar, lots of network packets directed at the same network service. For this visualization, ICMP packets are regarded as port 0, since ICMP as a protocol does not have ports. Also notice that the attack occurred in the second time slice of the managed network. There is no black coloring, indicating ICMP traffic, in the first time slice represented by the center of ring in the managed node representation of the machine with the host name "HEAP". The difference between this network attack and an application level network attack such as the SSH DoS attack above is that the ICMP layer is harder to

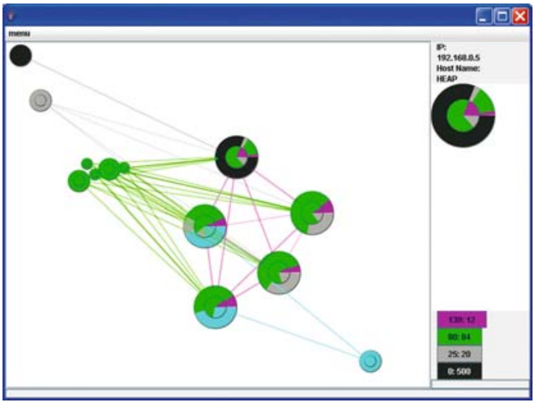


Fig. 8 Visualization of a student's machine under attack from an ping flood

defend. The main reason for this is that services such as SSH must be enabled by a user and are not active by default while ICMP services (such as ping) are generally enabled by default and often require changes to the operating system's kernel to disable or change behavior. However, ICMP services are often more lightweight than application level services, requiring less communication to perform their function, making it more difficult to use these services for a DoS attack. It is important for an administrator to see these attacks because although the user may not be able to disable such services, the network administrator can prevent malicious traffic from getting to the user at all.

3.4.3 Evaluation

This visualization was evaluated by surveying network administrators about effectiveness. The survey consisted of questions asking the network administrator to draw conclusions from the visualization. Network administrators of varying experience and ability were used in the survey to better evaluate who the visualization is useful for. The survey displays different visualizations of the same network with varying network conditions, including different types of network attacks. The administrator was asked to identify the anomaly by comparing the different visualizations of the same network. One of the choices we give is a network attack that this visualization technique does not support and is not visually present in our resulting figures. This evaluation was designed to address the feasibility of this visualization technique assisting network administrators in detecting security events but is not a formal validation of the effectiveness of the technique.

Five network administrators were able to identify which type of network attacks were taking place in the visualization with high accuracy. Even when they were incorrect, they were able to identify the features shown in the visualization and describe them. The survey questions were asked for the figure representing the normal network (Fig. 4) and each of the five analysts produce similar answers. Each participant was easily able to identify managed nodes vs. unmanaged nodes. Each participant was also able to identify that two machines were using instant messaging traffic, four machines have sent email, and machines were accessing web servers. One participant thought it was odd that all of the students access the same set of web servers which is fairly unlikely and more a problem with my simulation. Although some analysts were skeptical of this being a normal network, when asked to choose between the choices listed above, each participant chose normal network. This image was shown to the participants first in order to give them some base knowledge of the network. All of the attacks occur on this same network, so this at least gives the participants of the survey some knowledge of the base network because a network administrator would certainly have knowledge of their network.

The SSH DoS attack (Fig. 7) produced consistent results among the participants of the survey. When asked to choose an identifier for this particular figure, all participants chose DoS attack and were able to identify it was against the SSH service. One of the participants noted that it could simply be a large file transfer over SSH, using a tool like sftp. Two other participants noted differences in instant messaging traffic over time, which was present, but was just a result of the randomness of the simulation. Two of the participants noted that the SSH traffic only occurred in the most recent time slice, although the other three participants were not explicitly asked about when the attack occurred. One participant was considering this to a possible session hijacking, but thought it out and chose DoS. This figure, especially compared with the normal network figure, provides an easily identifiable increase in traffic on the SSH service between two nodes and each participant was immediately able to identify it.

The compromised network (Fig. 5) produced good results among the participants of the survey. Each participant was able to identify light traffic occurring across port 7000, or IRC traffic. When comparing this model to the network, each participant tried to figure out the anomalous behavior in each of the darker nodes. Three of the five participants were able to quickly identify this network as network compromised by trojans that communicate via IRC. However, two of these administrators were aware that IRC is a commonly used protocol for trojans. The other two participants eliminated most of the choices listed (DoS, distributed DoS, etc.) but were unable to decide between normal network, compromised network with trojans, or session hijacking. However, these two participants, when asked, did not know that IRC is a commonly used protocol for trojans. One of the two participants that did not choose compromised network was wondering how much IRC traffic is a normal amount of IRC traffic for a particular host but assumed the small number of packets using the IRC service were normal and therefore chose normal network. The compromised network is one of the types of attacks that other visualizations do not focus upon and our method is designed to identify. With network security knowledge that IRC is a commonly used protocol for trojans, our participants were able to use this visualization to identify the attack.

The distributed DoS attack (Fig. 6a) was quickly and easily identified by each participant as a distributed DoS attack. The addition of multiple unmanaged nodes, each using the same service and grouped together, made this an easily identifiable attack. Four of the five participants looked at the time slicing on the unmanaged web traffic nodes to conclude that this was a distributed DoS attack. Those participants were able to assume that continuous traffic over time slices from all unmanaged web traffic nodes would only occur in an attack. Two of the five participants were considering this to be normal traffic to a web server assuming that maybe the web server just got turned on or got linked from a popular news site but still chose distributed DoS when presented with the options.

The normal heavy web usage (Fig. 6b) took the most thought for each of the five participants. Each of their initial thoughts was a potential distributed DoS attack. Three of the five participants considered it normal network usage and the other two participants considered it a possible distributed DoS attack. One of the participants that considered this figure a DDoS attack said it was the end of a DDoS which occurred in the first time slice. The other participant that considered this a DDoS, said it was possible a quick DDoS which stopped, or just heavy usage. Each of the five participants viewed time slicing on the unmanaged web traffic nodes to identify that the amount of web traffic went to zero over time. One of the participants noted that some of the unmanaged web nodes produced constant web traffic over multiple time slices which was a result of the randomization of the simulation. Each participant was able to view time slicing to differentiate this figure from the web DDoS figure which is the goal of time slicing in this context.

The ping flood (Fig. 8) proved to be another figure where the attack was quickly and easily identifiable by each of the five participants. Each participant identified some type of DoS attack occurring over ICMP. Three of the five participants identified this as a "ping of death" or ping flood attack, which was the attack we attempted to simulate in this figure. All participants noted that this attack occurred in the most recent time slice against one of our managed nodes that was behaving normally in the previous time slice.

Each participant was asked after each of the different figures if using Ethereal, which is a common practice of a network security administrator, would help or be better than using our visualization to identify an attack. Four of the five participants felt that this visualization method makes it significantly easier to identify attacks. All five participants felt that using this visualization tool to identify a starting point for analyzing anomalous behavior and then further investigating using ethereal would be useful. One of the five participants preferred to use Ethereal setup with various filters for each type of attack to actively monitor a network, but noted that it would only be effective if the administrator was able to actively see everything in Ethereal meaning that the amount of traffic was low enough to be managed on a line by line basis by the administrator.

4 Future Work

In our comparison to network model section, we used an overly simplistic model to demonstrate the opacity effect. Our future work should include creating several models designed for defined networks to give a more realistic feel impressive of the comparison to model section. Furthermore, using better models in the evaluation section will improve the conclusiveness of the results of the visualization technique based upon the model. Along with using more realistic data models, using real network data instead of simulated network data would improve the practicality of the evaluation section. Creating a good network model may also prove to be a difficult task for a network administrator. We will consider using machine learning techniques using user actions as attributes to attempt to build a network model based upon what the administrator does interactively. While our layout technique is effective in clustering nodes experiencing similar network behavior, we lose details due to occlusion as the node set gets larger. More work can be done using more interactive controls, grouping, and self-organizing maps to address this issue.

5 Conclusions

This approach aids a network administrator in identifying network attacks and intrusions. By visualizing the network from a service perspective, more specific types of attacks can be detected. The contribution of this research is to provide an application using a combination of existing visualization techniques applied a network traffic data set in order to better detect the type, severity, and presence of a network attack. Furthermore, this research visualizes more information than previous network traffic maps by approaching the data from a service oriented perspective and embedding multivariate information into the glyph representing a node. Based upon the results of our evaluation survey, network administrators feel that they gather more information in order to successfully detect attacks using this visualization technique. Finally, the temporal data present in the compound glyph via time slicing provides information for an administrator to distinguish between high volume cases and distributed DoS attacks.

There are some limitations to this approach including visual scalability and custom models. The comparison feature of this visualization is only available and effective when used with a good model of an existing network. In our results we use a very basic and simple model, which covers a variety of anomalous behavior, but does not get into enough detail about the normal activity of the network in order to be more useful. Basically, our model assumes all networks are the same, and things such as web and instant messaging chat should be the large majority of traffic. Such a model would not compare well with a cluster of SSH servers. Also, scalability to large networks is a limitation.

References

Ball, R., Fink, G.A., North, C.: Home-centric visualization of network traffic for security administration. In: VizSEC/DMSEC '04: Proceedings of the 2004 ACM Workshop on Visualization and Data Mining for Computer Security, pp. 55–64. ACM Press, New York (2004)

Becker, R.A., Eick, S.G., Wilks, A.R.: Visualizing network data. IEEE Transactions on Visualization and Computer Graphics **1**(1), 16–28 (1995)

Charles, P.: Jpcap: Network Packet Capture Facility for Java. http://sourceforge.net/projects/jpcap (2001)

Conti, G., Abdullah, K.: Passive visual fingerprinting of network attack tools. In: VizSEC/DMSEC '04: Proceedings of the 2004 ACM Workshop on Visualization and Data Mining for Computer Security, pp. 45–54. ACM Press, New York (2004)

Erbacher, R.F.: Glyph-based generic network visualization. In: Proceedings of the SPIE '2002 Conference on Visualization and Data Analysis, pp. 228–237 (2002)

Girardin, L.: An eye on network intruder-administrator shootouts. In: Proceedings of the Workshop on Intrusion Detection and Network Monitoring (ID'99), pp. 19–28. USENIX Association, Berkeley, CA, USA (1999)

Goldring, T.: Scatter (and other) plots for visualizing user profiling data and network traffic. In: VizSEC/DMSEC '04: Proceedings of the 2004 ACM Workshop on Visualization and Data Mining for Computer Security, pp. 119–123. ACM Press, New York (2004)

Komlodi, A., Goodall, J.R., Lutters, W.G.: An information visualization framework for intrusion detection. In: CHI '04: CHI '04 extended abstracts on Human Factors in Computing Systems, pp. 1743–1746. ACM Press, New York (2004)

Lakkaraju, K., Bearavolu, R., Slagell, A., Yurcik, W., North, S.: Closing-the-loop in nvisionip: Integrating discovery and search in security visualizations. In: VIZSEC '05: Proceedings of the IEEE Workshops on Visualization for Computer Security, p. 9. IEEE Computer Society, Washington, DC, USA (2005)

McPherson, J., Ma, K.L., Krystosk, P., Bartoletti, T., Christensen, M.: Portvis: a tool for port-based detection of security events. In: VizSEC/DMSEC '04: Proceedings of the 2004 ACM workshop on Visualization and Data Mining for Computer Security, pp. 73–81. ACM Press, New York, NY, USA (2004)

Papadopoulos, C., Kyriakakis, C., Sawchuk, A., He, X.: Cyberseer: 3d audio-visual immersion for network security and management. In: VizSEC/DMSEC '04: Proceedings of the 2004 ACM Workshop on Visualization and Data Mining for Computer Security, pp. 90–98. ACM Press, New York (2004)

Shanbhag, P., Rheingans, P., desJardins, M.: Temporal visualization of planning polygons for efficient partitioning of geo-spatial data. In: INFOVIS '05: Proceedings of the 2005 IEEE Symposium on Information Visualization, pp. 28–36. IEEE Computer Society, Washington, DC, USA (2005)

Teoh, S.T., Ma, K.L., Wu, S.F., Jankun-Kelly, T.J.: Detecting flaws and intruders with visual data analysis. IEEE Computer Graphics and Applications **24**(5), 27–35 (2004)

Yin, X., Yurcik, W., Treaster, M., Li, Y., Lakkaraju, K.: Visflowconnect: netflow visualizations of link relationships for security situational awareness. In: VizSEC/DMSEC '04: Proceedings of the 2004 ACM Workshop on Visualization and Data Mining for Computer Security, pp. 26–34. ACM Press, New York (2004)