VisFlowConnect: NetFlow Visualizations of Link Relationships for Security Situational Awareness

Xiaoxin Yin
National Center for Supercomputing Applications (NCSA)
University of Illinois at Urbana-Champaign
605 E. Springfield Ave.
Champaign, IL 61821
xiaoxin@ncsa.uiuc.edu

William Yurcik
National Center for Supercomputing Applications (NCSA)
University of Illinois at Urbana-Champaign
605 E. Springfield Ave.
Champaign, IL 61821
byurcik@ncsa.uiuc.edu

Michael Treaster
National Center for Supercomputing Applications (NCSA)
University of Illinois at Urbana-Champaign
605 E. Springfield Ave.
Champaign, IL 61821
treaster@ncsa.uiuc.edu

Yifan Li
National Center for Supercomputing Applications (NCSA)
University of Illinois at Urbana-Champaign
605 E. Springfield Ave.
Champaign, IL 61821
yifan@ncsa.uiuc.edu

Kiran Lakkaraju
National Center for Supercomputing Applications (NCSA)
University of Illinois at Urbana-Champaign
605 E. Springfield Ave.
Champaign, IL 61821
kiran@ncsa.uiuc.edu

ABSTRACT

We present a visualization design to enhance the ability of an administrator to detect and investigate anomalous traffic between a local network and external domains. Central to the design is a parallel axes view which displays NetFlow records as links between two machines or domains while employing a variety of visual cues to assist the user. We describe several filtering options that can be employed to hide uninteresting or innocuous traffic such that the user can focus his or her attention on the more unusual network flows.

This design is implemented in the form of VisFlowConnect, a prototype application which we used to study the effectiveness of our visualization approach. Using VisFlowConnect, we were able to discover a variety of interesting network traffic patterns. Some of these were harmless, normal behavior, but some were malicious attacks against machines on the network.

Categories and Subject Descriptors

C.2.0 [**Computer-Communication Networks**]: General—*security and protection*; K.6.5 [**Management of Computing and Information Systems**]: Security and Protection—*invasive software*

General Terms

Security

Keywords

NetFlows, Security Visualization, Link Relationships, Link Analysis, Parallel Axes, Parallel Coordinates, Security Situational Awareness

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

1. INTRODUCTION

Networks are becoming increasingly complex. The number of users utilizing shared machines is growing, and the number of different applications running on these machines is growing proportionally. It has long passed the time that a system administrator could know about all activities on machines under his or her control.

At the same time, the number of hostile attacks against shared machines is increasing. These attacks conceal themselves among this vast amount of legitimate, yet chaotic, traffic that occurs every day. This makes it difficult for an administrator to detect attacks in time to stop them, especially when he or she is forced to rely on traditional command line tools [16].

The human mind is capable of very fast visual processing, outweighing the data mining capabilities of machines. Tools that visually depict network traffic patterns leverage this capability. They can provide the ability for a user to

drill down into the data to extract more detailed information about potential attacks. Such tools would enable network administrators to sift through the gigabytes of daily network traffic quickly and efficiently in order to identify anomalous patterns and note them for further investigation while ignoring the innocuous traffic.

VisFlowConnect is an example of this type of tool. It uses a parallel axes representation to display network traffic flows between external domains and internal machines. If an unexpected external domain is observed, or if an expected domain shows an unexpected amount of traffic to machines on the internal network, the user can focus on this particular domain to obtain more detailed information on the possible attack. For example, one detail view displays traffic between individual machines on the remote domain and machines on the local network, allowing the user to see exactly which IP addresses are generating the anomalous traffic.

VisFlowConnect is one component of the Security Incident Fusion Tool (SIFT) [31]. SIFT is a component-based approach that divides complex tasks between multiple, highly focused, complementary tools. The ultimate goal of SIFT is to provide network administrators with complete situational awareness of their networks in real time in order to improve their ability to defend against attacks or intrusions. To this end, the tool set focuses on four main areas: visualization, correlation, profiling, and data mining. VisFlowConnect is one of two complementary tools for visualization. The other is NVisionIP [36, 21].

The remainder of this paper is organized as follows: Section 2 describes related work. Section 3 briefly describes the data format used by the application. Section 4 details the goals of the VisFlowConnect application, the interface principles used to accomplish these goals, and the final implementation of the application interface. Section 5 describes several types of security events that can be observed by using this tool. Section 6 highlights some possible areas for extending VisFlowConnect in the future, and Section 7 draws some final conclusions on this work.

2. RELATED WORK

This research intersects three substantial, overlapping areas of research: link analysis, network traffic visualization, and intrusion detection.

2.1 Link Analysis

Link analysis is an subset of the data mining field concerned with extracting useful information from a large data set of associations between entities. This can be accomplished either by having a machine automatically detect the patterns using machine learning or statistical methods, or it can be accomplished by presenting the data visually, allowing a human user to view the data in a way that allows him or her to quickly see the important relationships.

Visualization approaches to link analysis typically represent the set of entity associations as a graph, with nodes representing entities from the data set and edges representing associations between those entities. With an appropriate depiction of the graph, a user can efficiently understand the structure of relationships between the various entities and extract any information he or she requires. [23]

A variety of commercial link analysis tools are available, including Analyst's Notebook [4], PolyAnalyst [30], Clementine [10], NetMap [25], and VisualLinks [33]. These types of commercial packages have proven valuable in extracting data relating to criminal investigations, fraud detection, counterterrorism, national and security, marketing and customer profiling, insurance, and scientific research.

The World Wide Web is widely studied using link analysis techniques. PageRank [27] is among the most well-known link analysis algorithms. It is used by the highly successful Google search engine to rank the importance of every web page based on an analysis of hyperlinks leading to the site. Despite the success of Google, research in this area continues as researchers strive to improve PageRank, such as by improving stability [26] or by increasing the generality [3] of the algorithm.

Other research has proceeded parallel to PageRank. The HITS [19] algorithm attempts to identify hubs of popularity and nodes of authority in a network of hyperlinked content. The PageCluster [37] algorithm, which counts user traversals of hyperlinks to measure semantic relationships between different web pages in order to cluster conceptually related pages into a certain level of a link hierarchy. The Citeseer [7] application is a popular Web-based research tool which automatically organizes research published online into a single, cross-referenced, searchable, digital library.

2.2 Network Traffic Visualization

Network traffic visualization seeks to represent the traffic of a network graphically such that a user can quickly glean information about the activity or performance of a network. This visualization can focus on different levels of network abstraction. The highest level of abstraction examines the Internet as a whole, while the most detailed level focuses on the individual packets associated with a single machine. Between these two extremes is a wide spectrum of levels of detail, each of which provides valuable insights into the operation of a network. There are a variety of network traffic visualization tools available, each focusing on a different level of network abstraction.

At the highest level, the ELISHA [32] uses Border Gateway Protocol (BGP) data to explore routing behavior on Internet backbones. SeeNet [6] combines geographic information and load data in a set of tools that assist in analyzing performance of an entire switched telephone network. Flowscan [29] focuses at a lower level of abstraction. It examines the total amount of traffic passing through a router regardless of the source or destination of the network flows and categorizes traffic according to the protocol or application generating the flow. [13] describes a visualization technique to represent collections of individual transactions and events on a single machine. At the lowest level, Nam [14] is a tool utilizing animation to display packet-level events for the purpose of protocol design and debugging.

NVisionIP [36, 21], another tool in the SIFT toolkit, spans multiple levels of network abstraction. It shows relationships between events on an entire network, on a subnet, or on a single machine using a graphical matrix representation with drill-down capabilities. As a complementary tool to VisFlowConnect, it focuses on representing activities occurring on machines while VisFlowConnect focuses on network flows between machines.

2.3 Intrusion Detection

Intrusion detection systems (IDS) are a topic of substantial commercial and academic research. They typically em-

```
ipaddrtypesrcaddr;// Source IP Address
ipaddrtypedstaddr;// Destination IP Address
ipaddrtypenexthop;// Next hop router ushort
input;// input interface index ushort output;//
output interface index ulong dPkts;// Packets sent in
Duration ulong dOctets;// Octets sent in Duration ulong
First;// SysUptime at start of flow. ulong
Last;// and of last packet of the flow. ushort
srcport;// TCP/UDP source port number ushort
dstport;// TCP/UDP dest port number uchar
flags;// Shortcut mode uchartcp_flags;// TCP
flags uchar prot;// IP protocol ID
uchartos;// IP Type-of-Service ulong
src_as;// source AS# ulong dst_as;//
destination AS# uchar src_mask;// source subnet mask uchar
dst_mask;// destination subnet mask ushort pad; ipaddrtype
router_sc;// Shortcut router
```

Figure 1: NetFlow Record Structure

ploy signatures of data collected from a variety of sources to detect abnormal activity. These signatures can be automatically acquired using machine learning [15] or data mining [5, 22] techniques, or they can be defined by policies authored by a human administrator. The system attempts to match the signatures to network activity and draw conclusions about intrusions based on the results of the matching.

Traditionally, IDS technology relies on data from only a single source. Recently, however, development has moved in the direction of correlating information from a variety of system data sources in order to find patterns in activity across the entire system [2, 20]. This work has succeeded in detecting intrusions where single-source techniques have failed.

Other recent work involves combining multiple IDS methods into a single, cooperative approach by analyzing correlations between alerts raised by each of the different techniques [11, 12]. This was shown to reduce the number of false positives and reduce the number of duplicate alerts, thereby reducing the noise that must be filtered by a user.

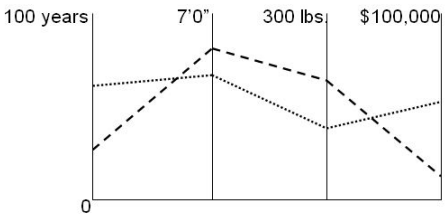
Finally, many IDS solutions are only capable of analyzing past network data. There is much research dedicated to detecting intrusions in real-time as the occur [28, 8]. This is a non-trivial problem that is even more difficult on large networks with high levels of innocuous traffic. Some examples incorporate multiple sources of data or multiple IDS methods, as we have described earlier.

3. NETFLOWS

VisFlowConnect relies on NetFlow [24] data as its input data source, but it is extensible to other data sources. The NetFlow format is a standard log format originally introduced by Cisco and later supported by many commercial routers. NetFlows can also be logged using open source software running independently of a router [9].

NetFlows record information on unidirectional end-to-end transactions between two machines, aggregating packets into larger "flows" of data. For TCP, this aggregation is accomplished using the sequence number on each packet. For UDP, the aggregation is accomplished by assuming all packets transmitted from machine A to machine B on the same port within a particular time threshold are part of the same flow of data between A and B.

VisFlowConnect can read either from past recorded log files or, in the near future, from a stream of NetFlow records



Joe	- - -	27	6'0"	200 lbs	$1000
Adam	.....	56	5'2"	138 lbs	$50000

Figure 2: Example of a Parallel Axes Representation

coming from a streaming socket in real time. Each record represents a distinct network flow and contains a variety of characteristics about the that flow, as shown in Figure 1. Our tool makes use of the following information: (1) source IP address and port number, (2) destination IP address and port number, (3) number of bytes transferred in the flow, (4) number of packets transferred in the flow, (5) protocol used, and (6) start and end times of the transmission.

Preliminary results describing the design of VisFlowConnect are published in [1, 35]. A more detailed description of the internal implementation of VisFlowConnect, including a performance and scalability analysis, has been performed and will be published in the near future.

4. VISUALIZATION DESIGN

VisFlowConnect was implemented as a demonstration of an efficient and effective visualization of network flows into and out of a network. The goal of the visualization technique is to display relationships between internal hosts and external machines, including the direction and volume of traffic. The visualization of the data enhances a network administrator's ability to detect intrusions or attacks on the network by improving his or her situational awareness of current and recent network events.

It has been claimed that some networks have too much traffic for all network flows to be viewed at once. Our visualization approach supports a high-level overview of the data, but it also allows the user to drill down into interesting or anomalous regions of the data in order to get more detail. This is accomplished by maintaining the individual NetFlow records without sampling or summarization even when the user is viewing aggregated data. The view at any level of detail fits on a single screen in order to obviate the need for arduous panning across the data by the user. Additionally, the tool highlights unusual patterns in the network flows, such as asymmetric traffic volume between two hosts, in order to draw the user's attention to these events. Finally, since network traffic patterns change over time, we provide an interface that allows the user to view temporal characteristics of the data using animation techniques.

4.1 Parallel Axes

In order to best view the link relationships, we used a parallel axes view [17, 34, 18]. This visual representation plots data from an arbitrary number of axes onto a two dimensional view. Figure 2 shows a trivial example of a parallel axes representation. It uses a set of parallel lines

with each line corresponding to an axis of the data. Each data point is then represented by a chain of line segments across these axes such that the point at which the segments cross each axis corresponds with the value of the data point on that axis.

This representation is valuable because it spreads out nodes across the axis so each one can be individually discerned while also drawing the user's attention to nodes of unusual activity, including unexpected intensity, unexpected IP address space, unexpected ports, or unexpected protocol interactions. It also groups similar links together to highlight outliers and emphasize baseline traffic patterns.

4.2 Animation

We use an animation mechanism to display temporal aspects of the data. The user can "replay" the network events recorded in a NetFlows log file to observe events as they occurred.

Data are filtered by a user-configurable time window. The time window shows only data with a timestamp within a certain range around the "present". Setting the time window size to infinite would show all traffic flows from the input data in the same view simultaneously, regardless of their temporal location or ordering. On a network of any significant size, this much traffic viewed at once would be overwhelming, and the user would have difficulty gleaning any useful information because individual flows would be indistinguishable. By shrinking the time window to a smaller, finite length of time, the application shows only network flows that occur within that length of time from the current time. The time window size is interactive to allow the user to find an optimal setting for the data.

The "present" time stamp and the size of the time window are visually represented using a time axis below the parallel axes view. As time advances in the replaying of the log, the time window, represented by a rectangle, slides from left to right along this axis. The motion of the rectangle represents the forward motion of time. New flows move into the time window from the future and old flows move out of the time window to the past. As data enters the time window it is added to the view, and as it leaves the time window it is removed from the view.

In VisFlowConnect, the current time and the size of the window are displayed in the main view to indicate these variables to the user. The horizontal axis at the bottom of each view represents the temporal axis, and the rectangle represents the span of time included in the current time window.

The time stamp associated with the NetFlow log record is also indicated by a wall clock representation with hands that move as the animation progresses, as well as with a digital date and time display accurate to 1/100th of a second. This allows the user to correlate events in the animation to an actual, real world time at a level appropriate for human senses while also allowing for machine-level accuracy when necessary. A time indication is important to provide the user with temporal awareness, since an event that might be normal at one time might be very conspicuous when observed at another time.

4.3 Interface Views

Our visualization approach combines a variety of types of views. Each view assists the user in extracting particular types of information from the NetFlow data and presents the data in particular ways to facilitate these insights.

4.3.1 Global View

Figure 3 shows the VisFlowConnect's Global View, the basic parallel axes view that acts as a starting point for our visualization approach. The figure shows the three parallel lines representing the different variables plotted by the parallel axes representation. The center vertical line represents machines on the internal network. The left vertical line corresponds to the originating domain of network traffic coming into the internal network, and the right vertical line represents the destination domain of outgoing traffic. Points on these axes are ordered by IP address of the machine or domain, with the lowest numbered IP address at the top of the axis. All traffic flows from left to right.

Our design displays domains rather than individual machines on the left and right axes because there are too many IP addresses to fit onto a single axis. During experimentation, we measured 100,000 addresses during one test period. Unless the tool is running on an extremely high resolution display, external machines would end up stacking on the same pixel, obfuscating the administrator's view of the data. By aggregating machines into domains for a high level view, the user is able to get general information about the origin and destination of traffic. Additionally, this aggregation serves to cluster related flows to assist the user in visualizing and examining network flows from a particular domain.

The multitude of lines between the three axes represents network flows. A darker or thicker line represents a larger amount of traffic. In cases where the traffic into a host is much less than or much greater than the traffic leaving the host, the flows are automatically highlighted in yellow. A user can color the flows from a selected external domain in order to highlight them from the other displayed traffic.

4.3.2 Domain View

A user can select a particular domain and zoom in on the data to see the Domain View. This view displays a visualization showing the data traveling between only the selected external domain and the local network. The center axis still represents hosts on the local network, but the left and right axes represent individual machines on the selected domain, instead of representing separate domains as in the Global View. As in the global view, traffic flows from left to right. The Domain View inherits the visualization features of the Global View, including selection highlighting, display of the IP addresses, and alert of asymmetric traffic volume.

The Domain View is crucial to understanding the network activity. Although the Global View provides a high-level overview of the data, it is usually inadequate for assessing the exact nature of an anomalous flow pattern. The Domain View allows the user to see whether traffic is originating from one machine in the remote domain or from many. If the traffic originates from many machines, the Domain View also allows the user to see patterns in IP addresses. For example, it might show unusual activity emanating from a block of sequential IP addresses, or it might show an interesting correlation of links between two sets of machines. These types of clues can be critical to understanding a security situation.

4.3.3 Internal View

Figure 4 shows another possible view of the data, the In-

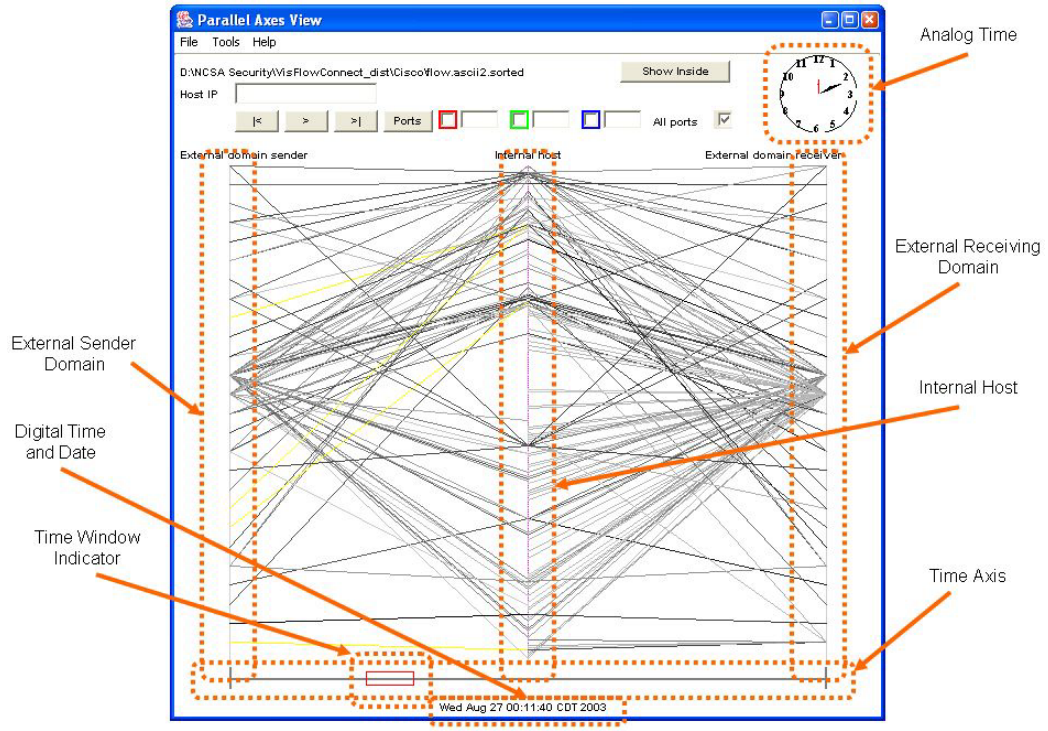


Figure 3: Global View

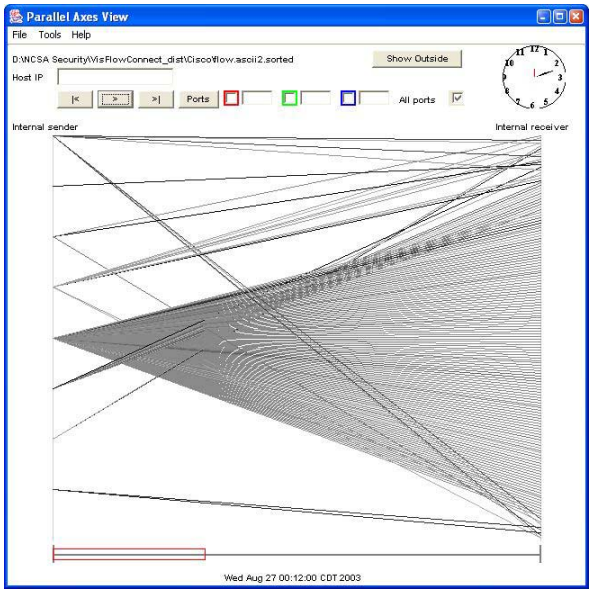


Figure 4: Internal Network View

ternal Network View. This depicts traffic flows that are entirely internal to the local network. Only two axes are shown in this view because no indication of whether the flow is entering or leaving the network is required.

As in the other views, traffic flows from left to right, so the left axis represents the source of the traffic and the right axis represents the destination machine. These axes are sorted by IP address. A particular machine can be selected to highlight in magenta the flows that enter or leave it, and the IP address of the selected machine is displayed in the upper left of the window.

This view is important because, despite emphasis on perimeter security, most security events are internal. For example, a worm or virus blocked by a firewall can still enter an internal network in many ways. For example, a laptop computer could acquire a virus while outside the firewall, then carry it past the firewall when the user next plugs in to the local network. This type of scenario can cause problems which can go undetected until disaster. We have used our prototype tool to detect employees violating security policies and to track internal virus infections.

4.3.4 Host Statistics View

Figure 5 depicts the Host Statistics View, which shows detailed data about traffic for a specific machine. It shows the total number of bytes flowing into and out of the machine in the current time window. Additionally, this total is subdivided to show the number of bytes transferred to and from each other individual machine with which the selected host communicated. This view allows the user to obtain more specific data about flows that is otherwise represented imprecisely using only by the darkness or thickness of a link in a parallel axis view.

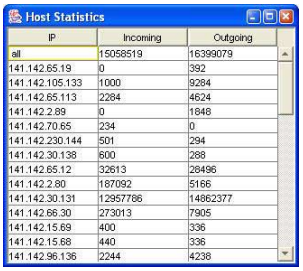


Figure 5: Host Statistics View

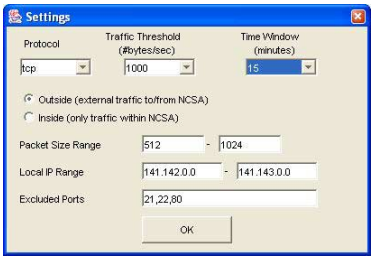


Figure 6: VisFlowConnect Filtering Options

4.4 Filtering

NetFlow logs for networks of any significant size typically contain a large amount of data. Filtering of this input data set by showing or hiding data with certain specified attributes can reduce the amount of visual noise the administrator must sift through. The user can employ such filters to focus an investigation in particular regions of the data by excluding the data he deems unrelated to his current task. This allows him to finish his task more easily and efficiently, since the tool displays only the important information. Additionally, by hiding data known to be innocuous, the administrator may see details he might have overlooked.

Multiple filters can be combined to produce more complex masking effects. Filter combinations result in more data being hidden from the view so a narrower class of data is shown. This provides the user with greater flexibility to hide normal traffic while viewing anomalous data.

A variety of filtering options can be conceived. We have selected four different filter types as being the most useful for our visualization approach: port number, protocol type, transfer rate, and packet size.

4.4.1 Port Filtering

Port filtering shows or hides data based on the network port over which traffic is flowing. We identify two useful types of port filtering: inclusive and exclusive. Inclusive port filtering restricts the input data to show only traffic flows using the selected ports. This allows the administrator view traffic that might represent an attempt to exploit an known vulnerability on a certain port.

Exclusive filtering shows data for all except the specified ports. This type of filtering allows the administrator to easily hide network flows corresponding to a large volume of known traffic that might obscure other, anomalous network flows in the data.

VisFlowConnect allows a user to select up to three separate port numbers for inclusive filtering, and the flows for each port highlighted in a different color. Any number of ports can be selected for external filtering, and the network flows that are not hidden are displayed normally.

4.4.2 Protocol Filtering

Protocol filtering shows only data corresponding to a specified network protocol. If the administrator learns of a new type of attack that uses a particular protocol, he or she can use this to get a clearer view of the suspect traffic by masking traffic known to be unrelated.

To demonstrate this feature, our application supports five types of protocol filtering. It can restrict the data to display only network flows using the TCP, UDP, or ICMP protocols, all of these, or all other protocols. Although NetFlows can identify all protocols based on the protocol ID, the ability to isolate ICMP scans and UDP packet flood attacks as separate from normal TCP flows is especially useful for security purposes. It would be a trivial extension to allow a user to filter according to any protocol that concerned him.

4.4.3 Transfer Rate Filtering

Another key attribute of a network flow is the transfer rate of the data. Transfer rate filtering causes the tool to show network flows based on the transfer rate of the flow. If an administrator knows that a particular attack employs very high or very low transfer rates, he can filter for only the appropriate data to help him detect the existence of such an attack more quickly.

Presently in VisFlowConnect, the user can select from several hard-coded threshold values the minimum transfer rate that is required for a flow to be displayed. However, this could easily be extended to allow the user to set one or more user-specified transfer rate ranges to display or hide, allowing the user to more closely examine very high speed of very low speed transmissions.

4.4.4 Packet Size Filtering

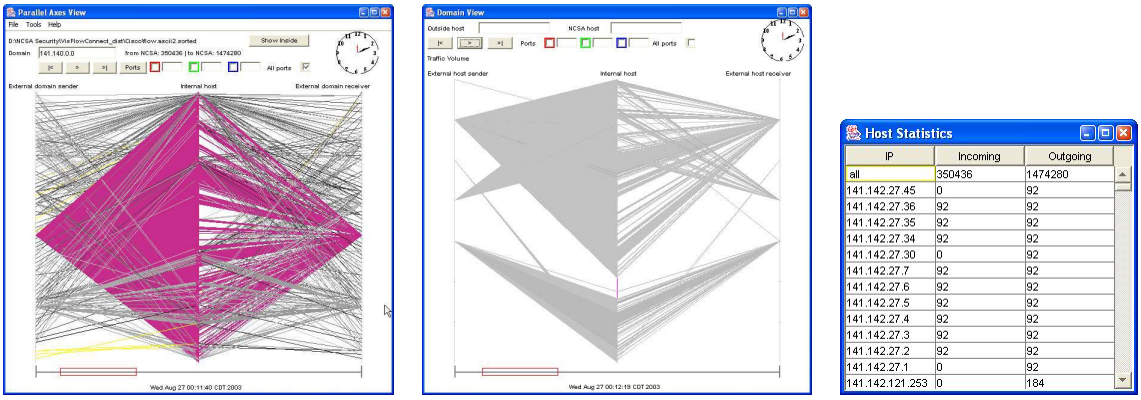
The user has the option of restricting the view to flows using packet sizes that fall within a particular user-specified range. If a known exploit employs packets of a particular size or range of sizes, the user can select only this range in order to block all other traffic and focus on packets of the suspect size. Although legitimate applications can use packets of any size in their communications, the appearance of packets of the suspect size on the network would be a strong indication that more investigation is needed. Conversely, if no packets of the suspect size are observed, the administrator can be confident that this type of attack had not occurred.

5. RESULTS

We use VisFlowConnect to demonstrate the ability of our visualization approach to detect a wide variety of interesting security events. Here we describe how the various features of our approach combine to produce a powerful security tool, using three examples of its success as case studies.

5.1 Virus Outbreak

Figure 7A depicts a global view in VisFlowConnect. This example shows an anomalous traffic pattern on a network, characterized by a single external domain generating so much traffic to so many machines on the local network that it is



(A) (B) (C)

Figure 7: Virus Outbreak

difficult to see other network activity clearly. An administrator familiar with the characteristics of this network would immediately notice this anomaly. This type of pattern is a strong indication that some kind of security event is in progress.

The domain view in Figure 7B shows a more detailed view of the domain responsible for the transmissions. The traffic initiates from multiple, unrelated hosts on this domain. Using this view, the administrator can gather the exact IP addresses that are involved in the network transactions.

Using the host statistics view, shown in Figure 7C, the administrator can see that the majority of these network flows use packets 92 bytes in size. With this information as a starting point, additional investigation would reveal that this network traffic is the result of the Blaster virus.

5.2 Denial of Service Attack

Figure 8A depicts another global view showing suspicious traffic pattern on a network. In this case, an unfamiliar domain is generating a significant amount of traffic to multiple hosts on the local network. Figure 8B is generated by filtering out all traffic except that on port 80. This reveals that a single domain is generating a large amount of traffic to many internal hosts on this port, suggesting that this external domain is contacting many local web servers. The domain view for this suspicious domain, displayed in Figure 8C, shows that many machines on the external network are heavily accessing a set of hosts on the local network. This type of pattern strongly suggests the occurrence of an attempted denial of service attack in progress on local web servers.

5.3 Grid Computing

Figure 9A shows a global view with another unusual traffic pattern. An external domain can be observed accessing a block of sequential IP address on the local network. This could indicate a port scan or other type of hostile reconnaissance.

Drilling down to the domain view, displayed in Figure 9B, reveals that there is a one-to-one relationship between a small number of machines on the internal network and some machines from the external domain. With this additional insight, the traffic pattern no longer looks like reconnaissance, but instead resembles grid computing activity with communication and cooperation between a remote cluster system and a cluster system on the local network. Although this is a false alarm from a security perspective, the tool successfully provides information and awareness of activity on the network.

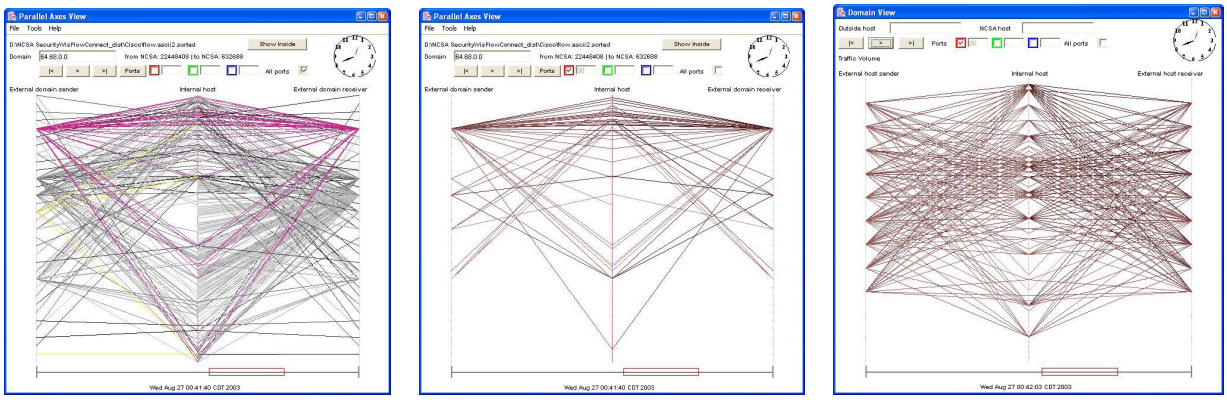
6. FUTURE WORK

Our visualization approach opens many opportunities for future work involving integration with other existing IDS technologies. Although it provides a strong visualization paradigm for network traffic flows, it does not alert the user to any possible suspicious activity within those flows, nor does it attempt to automatically reduce the amount of noise that an administrator must filter through when attempting to find evidence of an intrusion.

This is not necessarily a negative, however. In situational awareness, by not diverting attention via alarms the user is able to use his or her expertise and intuition to focus his or her attention. Alarms have well-known problems with false positives. Even when accurate, the number of alarms diminishes user attention and also diverts attention from other significant activity. It can often focus attention on the symptoms of a problem while drawing attention away from the more subtle causes.

Our visualization approach is inadequate in several key areas. Although we have developed an efficient representation of NetFlows to enable a user to see relationships between flows, it does not provide a sufficient amount of drill-down depth when a user requires additional detail. Data of interest include, but is not limited to, the port number, protocol, and transfer rate of the flow. The user can filter data according to these attributes, but flows cannot be queried for these attributes. It would be very useful to not only provide access to this data, but to also develop a consistent extension to our existing visualization techniques to allow a user to observe patterns between these characteristics.

Once this data is made available to the user, another avenue of improvement for VisFlowConnect is to correlate the NetFlow data with other external network or system events. This would provide the user with easy access to other sources of information when attempting to assess the nature of an unusual pattern in the network traffic. Ideally, more visualizations would be developed to display this information as well.



(A) (B) (C)

Figure 8: Denial of Service Attack

(A) (B)

Figure 9: Grid Computing

7. CONCLUSIONS

We have presented a novel approach to the visualization of traffic flows and patterns on a network. It focuses on enhancing an administrator's situational awareness by providing an easy-to-use, intuitive view of NetFlow data using link analysis.

The central aspect of this interface is the parallel axes view, used to represent the origin and destination of network traffic. A high-level overview of the data, fitting all on one screen, is provided first, and the user is provided with several methods to drill down into the data to find additional detail. A variety of filtering mechanisms are provided in order to assist the user in extracting interesting or important traffic patterns.

Using VisFlowConnect, we have shown that our visualization approach is capable of revealing a variety of network traffic patterns relevant to security. Some anomalous patterns were eventually found to be benign (in the case of the grid computations), but others demanded further investigation and defense using other tools. This demonstrates that these visualization techniques serve as a powerful tool for situational awareness of network security events.

8. ACKNOWLEDGMENTS

We would like to acknowledge the significant intellectual input of our SIFT colleagues, whose work and insights indirectly contributed to this paper: Cristina Abad, Ratna Bearavolu, Adam Lee, and Adam Slagell.

We would also like to acknowledge our NCSA colleagues Jim Barlow, Tim Brooks, Jeff Rosendale, and Ashish Sharma of the NCSA security operations group.

9. REFERENCES

[1] C. Abad, Y. Li, K. Lakkaraju, X. Yin, and W. Yurcik. Correlation Between NetFlow System and Network Views for Intrusion Detection In *Workshop on Link Analysis, Counter-terrorism, and Privacy held in conjunction with the SIAM International Conference on Data Mining (ICDM)*, 2004.

[2] C. Abad, J. Taylor, C. Sengul, W. Yurcik, Y. Zhou, and K. Rowe. Log correlation for intrusion detection: A proof of concept. In *Annual Computer Security Applications Conference (ACSAC)*, 2003.

[3] S. Acharyya and J. Ghosh. A maximum entropy framework for higher order link analysis on directed graphs. In *Workshop on Link Analysis for Detecting Complex Behavior (LinkKDD)*, August 2003.

[4] Analyst's notebook software. `www.i2inc.com/Products/Analysts_Notebook/default.asp`.

[5] D. Barbara, J. Couto, S. Jajodia, L. Popyack, , and N. Wu. Adam: Detecting intrusions by data mining. In *Proceedings of the IEEE Workshop on Information Assurance and Security*, June 2001.

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

[7] K. Bollacker, S. Lawrence, and C. L. Giles. CiteSeer: An autonomous web agent for automatic retrieval and identification of interesting publications. In *Proceedings of the Second International Conference on Autonomous Agents*, pages 116–123, New York, 1998. ACM Press.

[8] J. Brutlag. Aberrant behavior detection in time series for network monitoring. In *Proceedings of the USENIX Fourteenth System Administration Conference LISA XIV*, New Orleans, LA, December 2000.

[9] C. Bullard. Audit record generation and utilization system (argus). `http://www.qosient.com/argus/` and `ftp://ftp.andrew.cmu.edu/pub/argus`.

[10] Clementine software. `www.spss.com/clementine/`.

[11] F. Cuppens and A. Mige. Alert correlation in a cooperative intrusion detection framework. In *Proceedings of the 2002 IEEE Symposium on Security and Privacy*, page 202. IEEE Computer Society, 2002.

[12] H. Debar and A. Wespi. Aggregation and correlation of intrusion-detection alerts. In *Proceedings of the 4th International Symposium on Recent Advances in Intrusion Detection*, pages 85–103. Springer-Verlag, 2001.

[13] R. Erbacher. Visual behavior characterization for intrusion detection in large scale systems, September 2001.

[14] D. Estrin, M. Handley, J. Heidemann, S. McCanne, Y. Xu, and H. Yu. Network visualization with nam, the vint network animator. *Computer*, 33(11):63–68, 2000.

[15] K. Fox, R. Henning, J. Reed, and R. Simonian. A neural network approach towards intrusion detection. Technical report, Harris Corporation, July 1990.

[16] J. G. Goodall, A. Komlodi, and W. G. Lutters. Information visualization for intrusion detection analysis: A needs assessment of systems and network security experts. In *Workshop on Statistical and Machine Learning Techniques in Computer Intrusion Detection*, Fairfax, VA, 2003.

[17] A. Inselberg. Parallel coordinates for multidimensional displays. *In Spatial Information Technologies for Remote Sensing Today and Tomorrow*, pages 318–322, 1984.

[18] A. Inselberg and B. Dimsdale. Parallel coordinates: a tool for visualizing multidimensional geometry. In *IEEE Visualization '90 Proceedings*, pages 361–378. IEEE Computer Society, October 1990.

[19] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5):604–632, 1999.

[20] C. Kruegel, T. Toth, and C. Kerer. Decentralized event correlation for intrusion detection. In *International Conference on Information Security and Cryptology (ICISC)*, Lecture Notes in Computer Science. Springer Verlag, December 2001.

[21] K. Lakkaraju, W. Yurcik, A. Lee, R. Bearavolu, Y. Li, and X. Yin. NVisionIP: NetFlow Visualizations of System State for Security Situational Awareness. In *ACM CCS Workshop on Visualization and Data Mining for Computer Security (VizSEC/DMSEC) held in conjunction with the 11th ACM Conference on Computer and Communications Security*, 2004.

[22] W. Lee, S. J. Stolfo, and K. W. Mok. A data mining framework for building intrusion detection models. In *IEEE Symposium on Security and Privacy*, pages 120–132, 1999.

[23] J. Mena. *Investigative Data Mining for Security and Criminal Detection*. Butterworth Heinemann, 2003.

[24] Netflow services and applications. Technical report, Cisco Systems, 1999.

[25] Netmap software. `www.netmapanalytics.com`.

[26] A. Y. Ng, A. X. Zheng, and M. I. Jordan. Stable algorithms for link analysis. In *Proc. 24th Annual Intl. ACM SIGIR Conference*. ACM, 2001.

[27] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.

[28] V. Paxson. Bro: a system for detecting network intruders in real-time. *Computer Networks (Amsterdam, Netherlands: 1999)*, 31(23–24):2435–2463, 1999.

[29] D. Plonka. Flowscan: A network traffic flow reporting and visualization tool. In *Proceedings of the USENIX Fourteenth System Administration Conference LISA XIV*, December 2000.

[30] Polyanalyst software. `www.megaputer.com/products/pa/index.php3`.

[31] Security incident fusion tools (sift). `www.ncassr.org/projects/sift/`.

[32] S. T. Teoh, K.-L. Ma, S. F. Wu, , and X. Zhao. Case study: Interactive visualization for internet security. In *Proceedings of 13th IEEE Visualization Conference*, 2002.

[33] Visuallinks suite. `www.visualanalytics.com/Products/VL3-0Features.cfm`.

[34] E. Wegman. Hyperdimensional data analysis using parallel coordinates. *Journal of the American Statistical Association*, 85:664–675, 1990.

[35] X. Yin, W. Yurcik, Y. Li, K. Lakkaraju, and C. Abad. VisFlowConnect: Providing Security Situational Awareness by Visualizing Network Traffic Flows. In *Workshop on Information Assurance (WIA04) held in conjunction with the 23rd IEEE International Performance Computing and Communications Conference (IPCCC)*, 2004.

[36] W. Yurcik, J. Barlow, K. Lakkaraju, and J. Rosendale. A prototype tool for visual data mining of network traffic for intrusion detection. In *3rd IEEE International Conference on Data Mining (ICDM), Workshop on Data Mining for Computer Security (DMSEC)*, 2003.

[37] J. Zhu, J. Hong, and J. G. Hughes. Pagecluster: Mining conceptual link hierarchies from web log files for adaptive web site navigation. *ACM Trans. Inter. Tech.*, 4(2):185–208, 2004.