VisFlowConnect-IP: A Link-Based Visualization of NetFlows for Security Monitoring *

William Yurcik
National Center for Supercomputing Applications (NCSA)
University of Illinois at Urbana-Champaign
byurcik@ncsa.uiuc.edu

Abstract

Network traffic dynamics have become an important behavior-based approach to assist security administrators in protecting networks. In this paper we present VisFlowConnect-IP, a link-based network flow visualization tool that allows operators to detect and investigate anomalous internal and external network traffic. We model the network as a graph with hosts being nodes and traffic being flows on edges. We present a detailed description of VisFlowConnect-IP functionality and demonstrate its application to traffic dynamics in order to monitor, discover, and investigate security-relevant events.

1 Introduction

Complexity is the enemy of security. Networks have become more complex in terms of size, topology, and especially traffic flows. Traffic flows are faster and the number of different applications generating flows grows continuously. While network traffic is an ideal place to monitor for security events since all security events leave some form of network trace, there is a major problem in that security events can also be concealed among the vast amount of legitimate traffic. It is often difficult just to capture and store network traffic, so analyzing and detecting attacks in near-real-time with current command line driven text-based tools can be especially challenging for non-experts.

However, humans excel at visual processing and identifying abnormal visual patterns. Visualization tools can translate the myriads of network logs into animations that capture the patterns of network traffic in a succinct way, thus enabling users to quickly identify abnormal patterns that warrant closer examination. Such visualization tools enable network administrators to sift through gigabytes of daily network traffic more effectively than searching text-based logs.

VisFlowConnect-IP visualizes network traffic as a parallel axes graph with hosts as nodes and traffic flows as lines connecting these nodes. These graphs can then be animated over time to reveal trends. VisFlowConnect-IP has these distinguishing features: (1) it uses animations to visualize network traffic

*This research was supported in part by a grant from the Office of Naval Research (ONR) under the auspices of the National Center for Advanced Secure Systems Research (NCASSR) <http://www.ncassr.org>

dynamics and (2) it provides multi-level views of network traffic including an overview and drill-down views that allow users to query for details on-demand, and (3) it provides filtering capabilities to remove known legitimate traffic so as to focus on potential security events.

In previous papers we have introduced the design and implementation of VisFlowConnect-IP [9, 10, 11, 12]. In this paper we seek to focus on describing the use of this tool in more detail specifically for the unique operational security forum provided by FIRST. The rest of this paper is organized as follows. In Section 2 we briefly describe the design of VisFlowConnect-IP. We describe how to use VisFlowConnect-IP in Section 3. Examples of using VisFlowConnect-IP to detect abnormal behaviors are presented in Section 4. We introduce the related work in Section 5. We end with a summary and future work in Section 6.

2 System Design

The general system architecture of VisFlowConnect-IP is shown in Figure 1. VisFlowConnect-IP has three main components: (1) an input agent that extracts NetFlow records, (2) a NetFlow analyzer that processes the raw data and stores important statistics, and (3) a visualizer that converts statistics into graphical animations. In this section, we briefly describe the design and implementation of each of the 3 components.

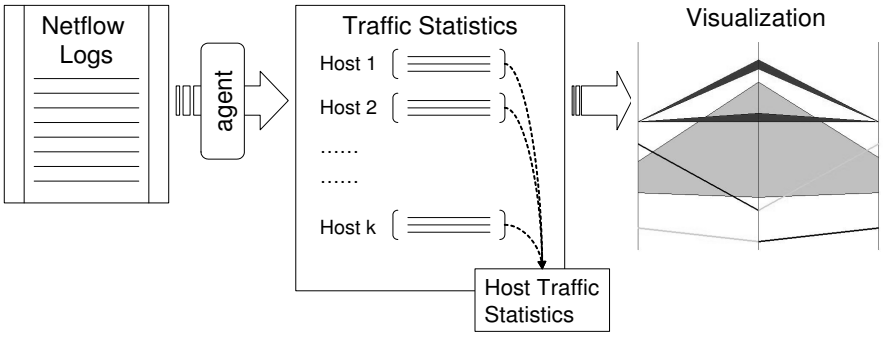

Figure 1. System Overview

2.1 NetFlow Source Data

VisFlowConnect-IP source data is NetFlows in the form of either Cisco format version 5 or Cisco format version 7 or Argus[1] format or NFdump formats. A setting menu within VisFlowConnect allows the user to select which input format to be used. A complementary tool called CANINE[2] has been developed to convert NetFlows between different formats as well as provide multi-level anonymization of NetFlow fields.

VisFlowConnect-IP works in batch mode, reading NetFlow records from a log file. An agent is used to extract individual NetFlow records sequentially from the log file and feed them into VisFlowConnect-IP. Each record contains the following information: (1) source/destination IP addresses and ports, (2)

[1] http://www.qosient.com/argus/

[2] http://security.ncsa.uiuc.edu/distribution/CanineDownLoad.html/

number of bytes and packets, (3) start and end timestamps, and (4) protocol type. These log files can be dynamically sized to match the the combined size and traffic volume of any network as well as the response interval required. For instance, smaller time intervals for NetFlows logs is appropriate for large networks or networks generating large traffic volumes for data management issues related to log file size. On the flip side, smaller networks or networks with lower traffic volumes may have longer timer intervals for NetFlow logs. Placement of the NetFlows sensors, either at the Internet border router or within the internal network is specific to the environment. For instance, NCSA with a Class B address space uses 5 minute NetFlows log files in the Argus format from the border router as well as 1 hour NetFlows log files in the Cisco format from routers within our internal network as well as on-demand Argus sensors that can be placed at different observation points.

Related to the collection time interval for NetFlows log files is the desired response time. An organization needs to match how they analyze and respond to events with their log file collection time interval. However, there is also the nature of NetFlows which only creates a record for a log file upon connection termination or after a 30-minute cache timeout[3].

2.2 Input Filtering Capability

NetFlow logs contain many different types of traffic with distinct properties. While certain traffic patterns may raise a red flag, most traffic is normal and benign. For example, it is common that a DNS server has connections with every other host on a network, but this same behavior on a workstation may indicate a worm infection. In order to remove noise such as this, VisFlowConnect-IP provides advanced filtering profiles that users can store and load. If this filtering capability is not configured, the default is that no filtering will take place.

Let $F_1, \ldots, F_k$ be a set of user created filters. Table 1 shows filter variables and their value ranges. Each filter has a list of constraints on the variables and a leading label ("+" or "–") that indicates whether to "include" or "exclude" matches. A constraint on a variable takes the form of "$x = v_{min} - v_{max}$", where "x" is a variable and "v_{min}" and "v_{max}" are the lower and upper bounds of "x", and "=" is the only operator defined. Records are passed sequentially through each filter and the last match will determine whether or not to include the record. A record that matches no filter rules is dropped. For example, the following set of filters will include all traffic from domain 141.142.x.x with a source port between 1 and 1000, except tcp traffic involving port 80.

+: (SrcIP=141.142.0.0-141.142.255.255), (SrcPort=1-1000)
-: (SrcPort=80, Protocol=tcp)
-: (DstPort=80, Protocol=tcp)

Variables	Value Ranges
SrcIP, DstIP	0.0.0.0 $\Leftrightarrow$ 255.255.255.255
SrcPort, DstPort	0 $\Leftrightarrow$ 65535
Protocol	tcp, udp, icmp
PacketSize	0 $\Leftrightarrow \infty$

Table 1. Input Filter Language

[3]This 30-minute cache timeout is configurable so it may be unrealistic to expect any incident response benefit from smaller time interval log files without considering the cache timeout setting and long duration flows.

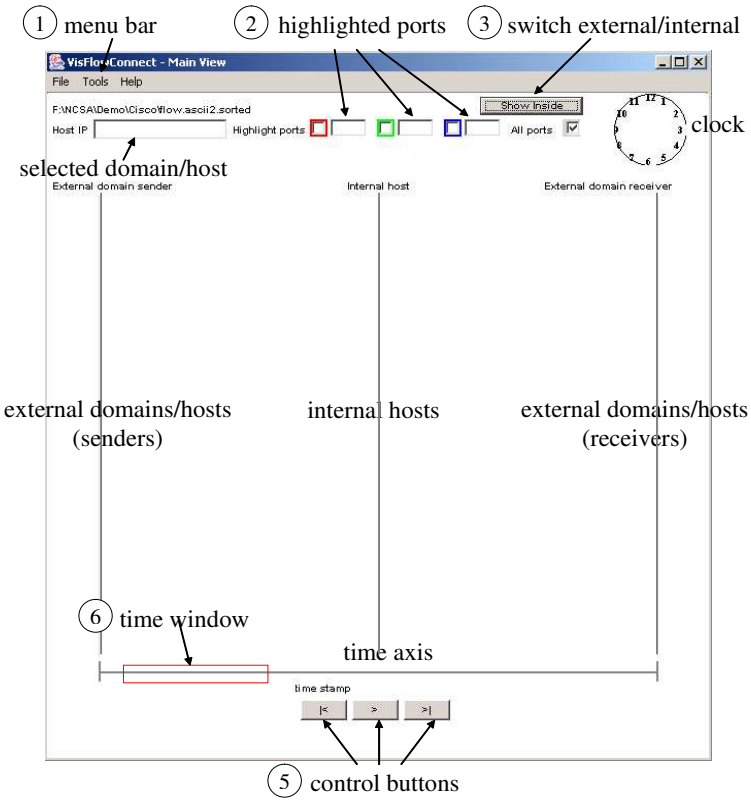

Figure 2. Parallel Axes View

3 How to Use VisFlowConnect-IP

In this section, we describe the visualization interface of VisFlowConnect-IP—which can be downloaded at <**http://security.ncsa.uiuc.edu/distribution/VisFlowConnectDownLoad.html/**>.

In *the Parallel Axes View*, three vertical axes are used to indicate traffic between external domains and internal hosts on the center axis (Figure 3). Points on the left [right] axis represents external domains that are sourcing [receiving] flows to [from] the internal network. Unlike the middle axis where points represent individual hosts, here points represent sets of hosts. The darkness of a line between two points is proportional to the logarithm of traffic volume between the hosts. All points are sorted according to their IP addresses, so that each point will remain at a relatively stable position for a user to track during animation. Figure 2 illustrates the VisFlowConnect-IP GUI with important features labeled.

1. **Menu Bar:** contains items for operations that are less frequently used, including (1) 'Open': open a NetFlow file, (2)'Load Filters': load a file for input filters, (3) 'Settings': bring up the settings dialog box, (4) 'Show Domain': show the domain view of the selected domain (described below), (5) 'Host Statistics': show the traffic statistics of the selected host/domain, and (6) 'Save Screen': save a snapshot of the current view.

2. **Highlighted Ports:** The user may specify up to three ports to highlight in special colors: red, green, or blue (see Figure 3). The user may also click on the check box to show traffic only on the highlighted port.

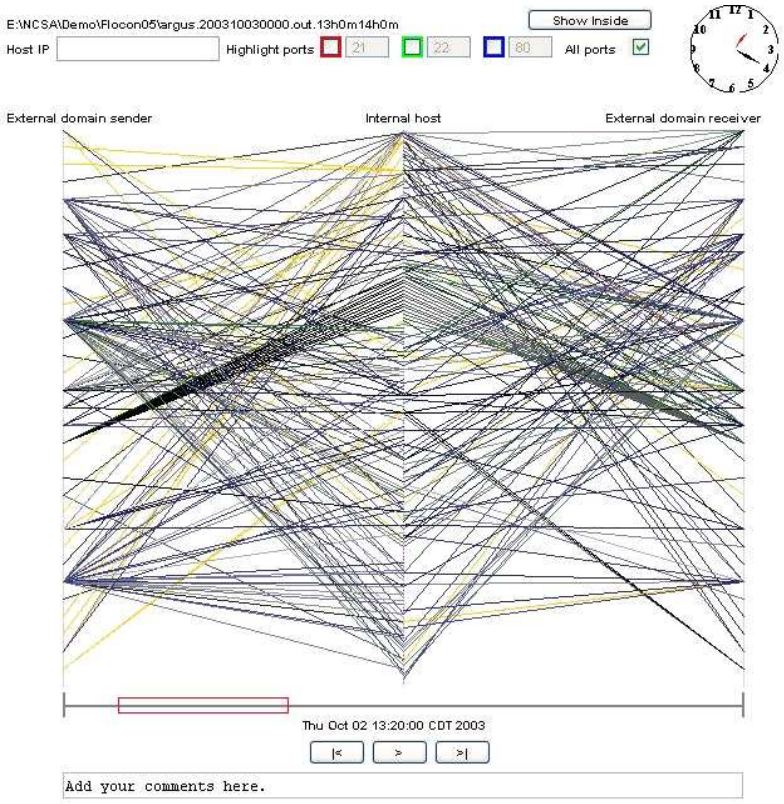

Figure 3. External View

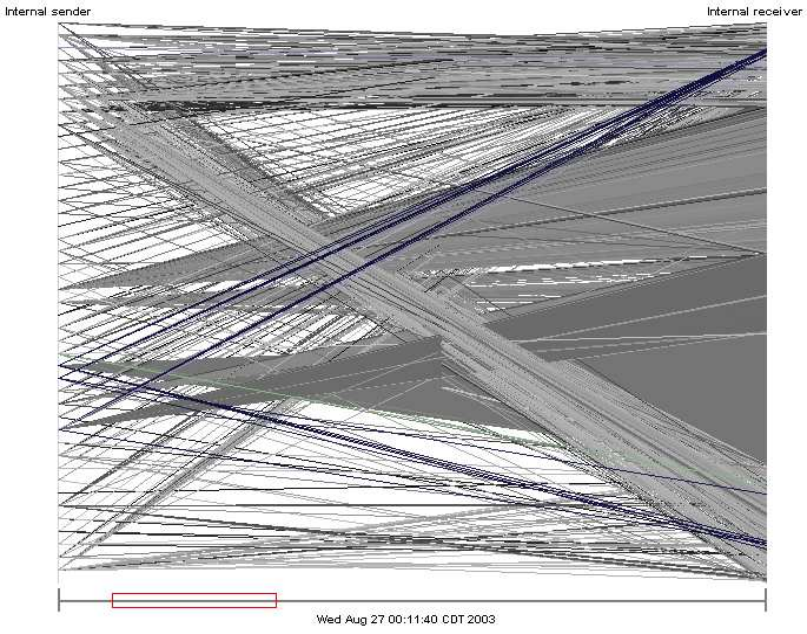

Figure 4. Internal View

Figure 5. Domain View

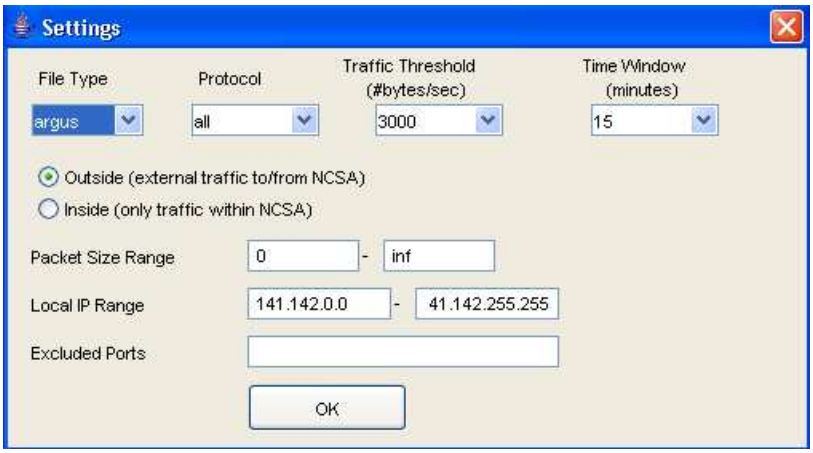

Figure 6. Setting Dialog

3. **External/Internal Switch:** The internal view (See Figure 4) shows traffic between hosts on the internal network. The points on the left [right] axis represent the source [destination] of traffic flows. The user may switch between external and internal views by clicking on the button "Show Inside/Outside".

4. **Domain View:** As shown in Figure 5, VisFlowConnect-IP has a drill-down Domain View that allows a user to visualize traffic between hosts in a specific external network domain to/from hosts in the internal network. The Domain View shows all traffic between individual hosts in the corresponding external network domain and the internal network.

5. **Control Buttons:** A user can control the animation with three buttons: ($|<$) rewind back to start, ($>$) play forward a defined time unit (default is 10 minutes), and ($>|$) play forward to the end of the data set.

6. **Time Window:** This key setting is user configurable for longer or shorter intervals depending on the volume and features of network traffic being studied. Only the flows within a time window are shown as opposed to a cumulative view. A sliding rectangle along a horizontal time axis is shown at the bottom of the GUI to indicate the time window in view.

7. **Settings Dialog:** Figure 6 shows the settings dialog, which allows the user to change the input file format (Cisco or Argus) and to select protocols of interest (e.g., tcp, udp or icmp). Here, the user may also adjust the traffic threshold, so that only domains whose aggregate traffic volume is lower (or higher) than this threshold are ignored. It also allows the user to change the time window, to restrict investigation to flows whose sizes are within a user-defined range, and to ignore flows over certain ports. This is also where the user sets the "Local IP Range" in order to distinguish internal hosts from external hosts.

4 Example Anomaly Detection

Here, we show an example of how we can detect the Blaster worm with VisFlowConnect-IP. The Blaster worm spreads quickly and has a common characteristic that infected computers send out packets to an abnormally large number of hosts within a short time period. In Figure 7, one can see that there is one domain which connects to almost every host in the local network. This indicates that some hosts in that domain may be infected by Blaster. This is verified when we see the uniform payload size and port usage on all of these flows that match the Blaster signature. At this point we can filter on those characteristics, and by digging deeper with the domain view, we can begin to identify specific hosts that have been infected. This exact procedure was used to identify the source of an internal Blaster worm infiltration at NCSA (despite perimeter protection).

Another example is non-malicious anomaly detection as shown in Figure 8. From the VisFlowConnect-IP main view we find that a particular domain (64.68.x.x) has intensive http traffic to our network. When we visualize traffic flows involving that domain we find that 9 hosts in that domain have intense and steady http traffic specifically to our web servers. Using the "whois" command we confirm our suspicion that these flows are from Google.com web crawlers retrieving web pages from our web servers for indexing.

5 Related Work

Linkages between different hosts and events in a computer network contain important information for traffic analysis and intrusion detection. Approaches for link analysis are proposed in [2, 3, 8]. [2] and [8] focus on visualizing linkages in a network, and [3] focuses on detecting attacks based on fingerprints. Link analysis can illustrate interactions between different hosts either inside or outside a network system, thus providing abundant information for detecting intrusions.

In [6], the authors present a visualization of network routing information that can be used to detect inter-domain routing attacks and routing misconfigurations. In [7], they go further and propose different

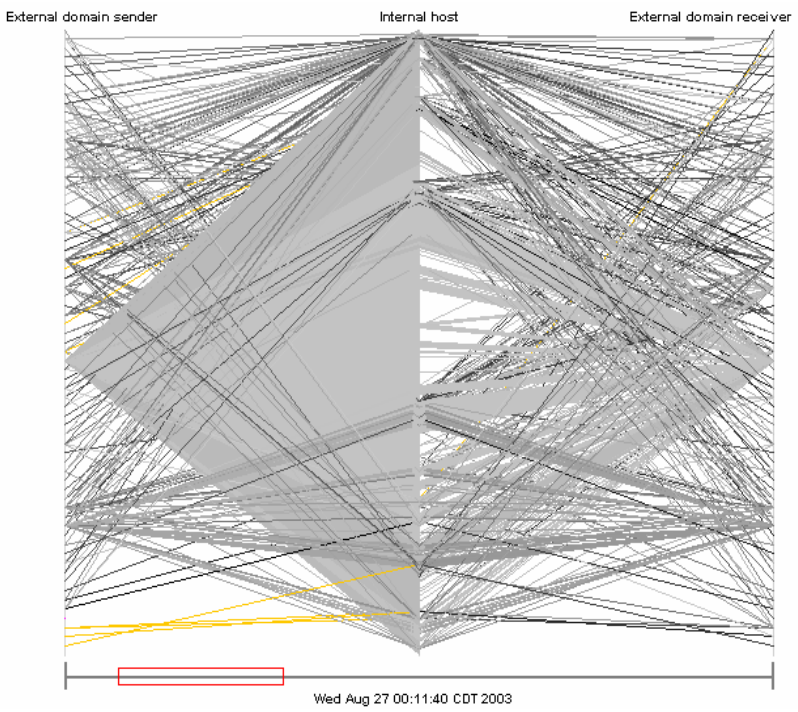

Figure 7. External view of blaster attacks

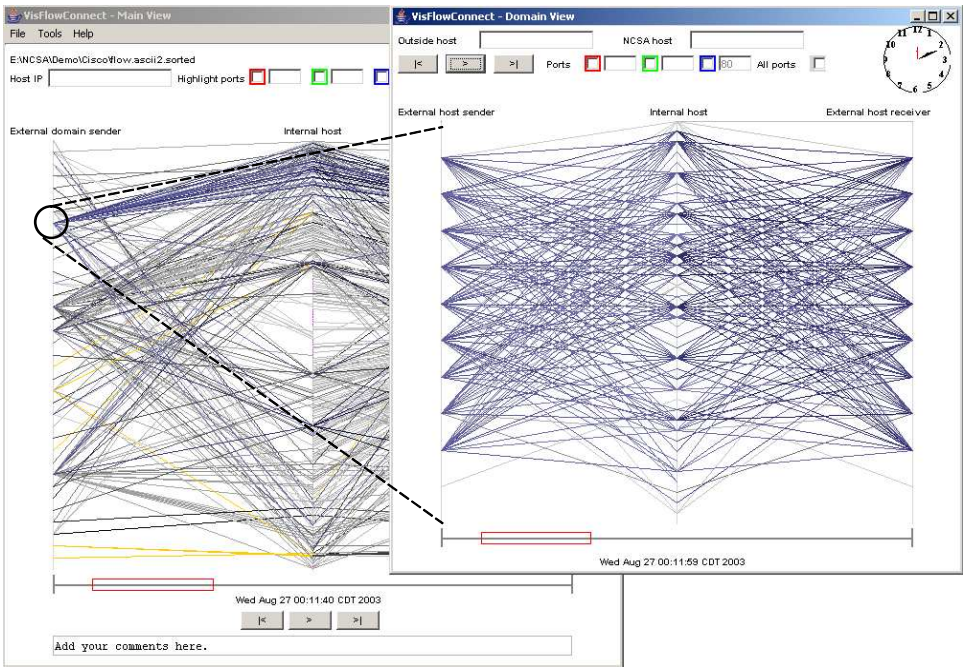

Figure 8. Visualizing detailed traffic involving a domain – traffic pattern of web crawlers

ways of visualizing routing data in order to detect intrusions. An approach for comprehensively visualizing computer network security is presented in [4], where Erbacher et al. visualize the overall behavioral

characteristics of users for intrusion detection. [1] focuses on visualizing log data from a web server in order to identify find patterns of malicious activity caused by worms.

The most closely related work is EtherApe[4]. EtherApe is a graphical network monitor which displays network traffic graphically in a ring. EtherApe is focused on the local area network environment by supporting level two network protocols including Ethernet, FDDI, token ring, ISDN, PPP, and SLIP. However, in the larger level three network environment of IP address spaces EtherApe does not scale to monitoring traffic on Class C or particularly Class B or Class A address spaces without the use of filtering to limit the number of IP addresses to be represented – the graphical ring convention is limited in size and traffic within the ring becomes obscured as traffic volumes increase. In contrast, VisFlowConnect-IP is designed to monitor a Class B IP address space and thus can also monitor smaller Class C address spaces as well as multiple Class B address spaces with multiple windows of VisFlowConnect-IP executing simultaneously.

6 Conclusions

In this paper we describe a framework for visualizing flow patterns – specificly we introduce the tool – VisFlowConnect-IP – for visualizing traffic flows on a network using NetFlow source data. This visualization framework is extensible to other domains beyond IP networks – we are currently modifying it to monitor access flows in storage systems and communication flows in high performance computing (HPC) clusters.

VisFlowConnect-IP enhances the situational awareness of a security administrator by providing an intuitive view of IP flows using link analysis. The primary graphical user interface to VisFlowConnect is a parallel axes view which is used to represent the source and destination of network traffic flows. A high-level overview is provided first with the user able to drill down to lower level views for more resolution and details on demand. Filtering mechanisms are provided to facilitate either extracting known legitimate traffic patterns or highlighting flows of special interest.

VisFlowConnect-IP is available for Internet download at this URL:
http://security.ncsa.uiuc.edu/distribution/VisFlowConnectDownLoad.html/
The code is still under development and when it is stable we will seek to release it as open source (SourceForge). There is an associated archived mailing list for questions and interaction with developers and other users, see information about subscribing to this majordomo mailing list on the VisFlowConnect download webpage. Since 2005 we have had hundreds of downloads from this website. We currently have a usability study with human subjects in order to compare the utility of VisFlowConnect-IP to current security administration tools. Preliminary results from this study are promising.

[4] see URL <etherape.sourceforge.net>

References

[1] S. Axelsson. Visualisation for Intrusion Detection - Hooking the Worm. *Eighth European Symposium on Research in Computer Security (ESORICS)*, Lecture Notes in Computer Science (LNCS) , Springer, 2003.

[2] R. Ball, G. A. Fink, C. North. Home-Centric Visualization of Network Traffic for Security Administration. *CCS Workshop on Visualization and Data Mining for Computer Security (VizSEC/DMSEC)*, 2004.

[3] G. Conti, K. Abdullah. Passive Visual Fingerprinting of Network Attack Tools. *CCS Workshop on Visualization and Data Mining for Computer Security (VizSEC/DMSEC)*, 2004.

[4] R. Erbacher, K. Walker, D. Frincke. Intrusion and Misuse Detection in Large-Scale Systems. *IEEE Comp. Graphics and Applications*, 22(1):38–48, 2002.

[5] R. Henning, K. Fox. The Network Vulnerability Tool (NVT) – A System Vulnerability Visualization Architecture. *NISSC*, 2000.

[6] S. T. Teoh et al. Elisha: a Visual-based Anomaly Detection System. *RAID*, 2002.

[7] S. T. Teoh, K. Ma, S. F. Wu. A Visual Exploration Process for the Analysis of Internet Routing Data. *IEEE Visualization*, 2003.

[8] S. T. Teoh, K. Zhang, S. Tseng, K. Ma, S. F. Wu. Combining Visual and Automated Data Mining for Near-Real-Time Anomaly Detection and Analysis in BGP. *CCS Workshop on Visualization and Data Mining for Computer Security (VizSEC/DMSEC)*, 2004.

[9] X. Yin, W. Yurcik, Y. Li, K. Lakkaraju, C. Abad. VisFlowConnect: Providing Security Situational Awareness by Visualizing Network Traffic Flows. *23rd IEEE Int'l. Performance Computing and Communications Conference (IPCCC)*, 2004.

[10] X. Yin, W. Yurcik, A. Slagell. The Design of VisFlowConnect-IP: a Link Analysis System for IP Security Situational Awareness. *3rd IEEE Int'l. Workshop on Information Assurance (IWIA)*, 2005.

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

[12] W. Yurcik. The Design of an Imaging Application for Computer Network Security Based on Visual Information Processing. *SPIE Defense and Security Symposium / Visual Information Processing XIII*, 2004.