Discrete Mathematical Approaches to Graph-Based Traffic Analysis*

Cliff Joslyn, Wendy Cowley, Emilie Hogan, Bryan Olsen
Pacific Northwest National Latboratory
{cliff.joslyn,wendy.cowley,emilie.hogan,bryan.olsen}@pnnl.gov

Abstract

Modern cyber defense and anlaytics requires general, formal models of cyber systems. Multi-scale network models are prime candidates for such formalisms, using discrete mathematical methods based in hierarchically-structured directed multigraphs which also include rich sets of labels. An exemplar of an application of such an approach is traffic analysis, that is, observing and analyzing connections between clients, servers, hosts, and actors within IP networks, over time, to identify characteristic or suspicious patterns. Towards that end, NetFlow (or more generically, IPFLOW) data are available from routers and servers which summarize coherent groups of IP packets flowing through the network. In this paper, we consider traffic analysis of NetFlow using both basic graph statistics and two new mathematical measures involving labeled degree distributions and time interval overlap measures. We do all of this over the VAST test data set of 96M synthetic NetFlow graph edges, against which we can identify characteristic patterns of simulated ground-truth network attacks.

1 Introduction

Cybersecurity analysts must increasingly manipulate massive-scale, high-resolution *flows* to identify, categorize, and mitigate attacks involving networks spanning institutional and national boundaries. A flow in this context is an aggregation of packet-level communication between two cyber systems over some network protocol between specific ports for a period of time. Flow data can be acquired at all levels of the Internet Protocol (IP) communications hierarchy, starting with bits and packets: each packet passing through a router or switch is inspected for a set of attributes and determined to be unique or associated with packets already seen. All packets with the same source, destination, ports and protocol are grouped into a flow and packets and bytes are aggregated accordingly. We find that this level of aggregation at the flow level is distinctly valuable and complementary to the more detailed packet level for exposing and reasoning about the communication graph patterns with which analysts work.

In this paper, we introduce a network-theoretical anlaysis of NetFlow multigraphs as follows:

- A basic characterization of NetFlow multigraphs and their projections to IP and port subgraphs.
- A description of the VAST cyber challenge test data.
- Some anlaytical results using basic NetFlow statistics.
- Characterizing IP interaction using novel information theoretical labeled graph degree distributions.
- An approach to characterizing temporal flow relations using interval orders and interval mathematics on flows.

While there have been a number of past attempts at IPFLOW graph analytics [3, 5], they have typically been limited in scale or complexity [6, 9], and frequently do not consider flow direction or other attributes [1, 10]. More recently, progress is being made on methods similar to ours, including some use of information-theoretical measures [7, 12]; and in temporal analysis [11], including flow streams with full time-interval overlap among `ssh` sessions [2], and flow collections occurring within a certain "encounter" tolerance [8].

These results apply both simple descriptive statistics, and some descriptive analysis using our two additional novel approaches, against simulated test data. As such, in addition to highlighting the potential significance against real data, they reveal aspects and even artifacts of the simulation itself. This is appropriate for the current stage of development, and indicates the promise for this approach.

2 NetFlow Graphs

Table 1 shows a typical data schema for IPFLOW records.[1] In addition to source and destination IP address and port, the number of packets and bytes comprising the flow, start and stop time, transport protocol (e.g. TCP vs. UDP), and other flags are included. This schema is naturally represented as a directed multigraph with a sample edge shown in Fig. 1, with the following characteristics:

- There is a combinatorial structure on node IDs, each a vector of the four octets of the IP address, together with the port ID. We call these "IPPs".
- # packets and # bytes as quantitative edge attributes.
- An interval attribute for start and stop times.
- A categorical attribute for transport protocol.

*Submitted to the 2014 International Workshop on Engineering Cyber Security and Resilience (ECSaR14)

©ASE 2014 ISBN: 978-1-62561-000-3

Field	Type	Description
EPOCH_TIME	NUMERIC	Flow start time in epoch milliseconds
PROTOCOL	SMALLINT	Transport protocol
SRCADDR	VARCHAR(15)	Source IP address
DSTADDR	VARCHAR(15)	Destination IP address
SRCPORT	INTEGER	Source Port
DSTPORT	INTEGER	Destination Port
MORE_FRAGMENTS	SMALLINT	Indicates more fragments
CONT_FRAGMENTS	SMALLINT	Continuation fragments
DURATION	NUMERIC	Duration in seconds
SRC_PAYLOAD	BIGINT	Source payload bytes
DST_PAYLOAD	BIGINT	Destination payload bytes
SRC_TOTAL_BYTES	BIGINT	Source total bytes
DST_TOTAL_BYTES	BIGINT	Destination total bytes
SRC_PACKETS	SMALLINT	Source packet count
DST_PACKETS	SMALLINT	Destination packet count
RECORDFORCEOUT	SMALLINT	Record force out

Table 1: Spec for input data set.

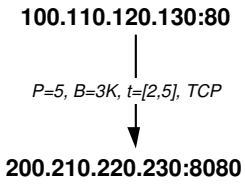

Figure 1: Link in a NetFlow multigraph: nodes are IP:Port, P=# packets, B=# bytes, t=time interval. Transport protocol also shown.

Nodes in NetFlow graphs are most naturally IPPs, which are $\langle \text{IP}, \text{Port} \rangle$ pairs. They thus have two parallel decompositions, or *projections* (*a la* "graph cubes", see [13]) into two sub-graphs, one among IPs, and one among ports. Fig. 2 shows a simplified example where IPs have only two octets, and edges are adorned with a single integer size. Each of the IP and port projections are smaller contractions of the original, and each contracted edge records the number of original edges in red and the aggregated size in black.

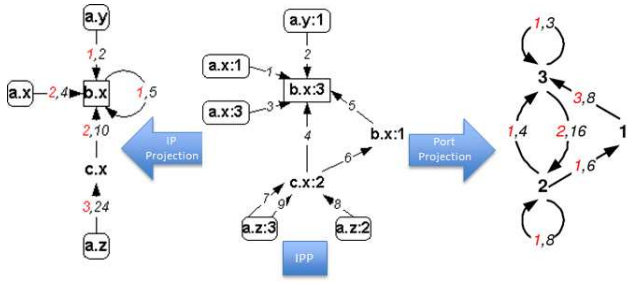

Figure 2: A simplified NetFlow graph and its IP and port projections.

[1]We are limiting our work to IPv4 but consider support for IPv6 within reach of our techniques.

3 VAST Challenge Data Set

Our test data set is from the VAST challenge from the annual Visual Analytics Science and Technology (VAST) conference.[2] One of the 2013 challenges was called "Big Marketing Situational Awareness", and focused on cyber security, representing unusual happenings on the computer network of a simulated marketing company. Participants were provided simulated NetFlow traffic, combined with IPS and BigBrother health monitoring, and were challenged to provide visualizations for situational awareness.

While used for VAST, the test data were actually created at our laboratory on a testbed with a collection of machines on an isolated network. Virtualization was used extensively, but not exclusively. The workstations of the Big Marketing sites were all simulated through custom software supporting behavioral rules to govern normal activities, as during business hours.

Fig. 3 shows the overall architecture. IPs with the 10. prefix are external to Big Marketing (BM), while those with 172. prefix are internal. BM is further divided into three internal sites with prefixes 172.10., 172.20., and 172.30. The overall scenario mostly simulates web servers, including both servers and workstations both internal and external. Fig. 4 shows the timeline for the simulated "ground truth" attacks, including a range of DOS attacks, port scans, and exfiltration events.

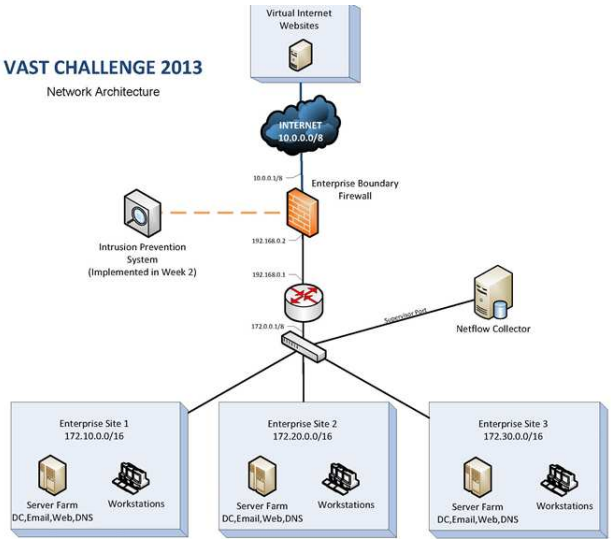

Figure 3: VAST challenge.

4 Basic NetFlow Analysis

Table 2 shows fundamental statistics about the VAST NetFlow multigraph, including the base IPP graph and the IP and port projections. There are 96.3M flows, divided over 10.1M IPPs, but only 1,440 IPs and (of course) 65,536 ports. "Leaves" and "roots" (aka "sources" and "sinks") have no children (resp. parents), as shown in Fig. 2 as square and round nodes. This yields only 1.25M internal IPP nodes, while retaining 1,329 internal IPs.

[2]http://vacommunity.org/VAST+Challenge+2013

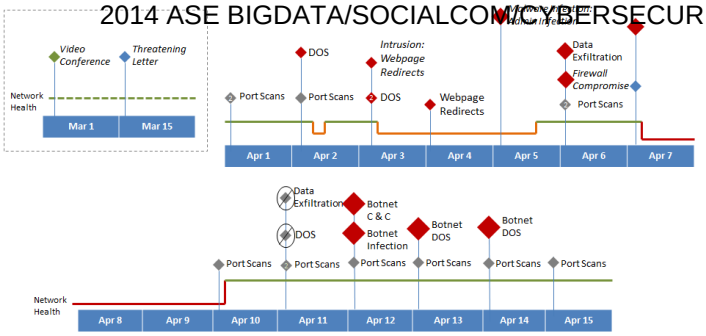

Figure 4: Simulated attacks and other activity in the BM scenario.

	IP		IPP		Port	
Nodes	1,440		10.1M		65,536	
Flows/node	48,192		6.89		1,059	
Leaves	16	1.1%	1.28M	12.7%	1,035	1.6%
Roots	95	6.6%	7.53M	74.8%	-	0.0%
Internals	1,329	92.3%	1.25M	12.4%	64,501	98.4%
Density	1.57%		0.0000646%		0.023%	

Table 2: VAST NetFlow basic statistics.

The NetFlow structure is relatively simple, and yet completely characterizing the potentially multi-variate statistical relationships between the components of flows, IPPs, IPs, ports, start times, finish times, durations, sizes, and protocols is still a daunting exercise. Yet before involving our novel methods, we can glean specific insights about the BM scenario from some of this initial analysis.

Fig. 5 shows # flows in vs. out for each IP. The multi-colored point collection in the upper right below the diagonal are BMs primary servers (e.g. 172.30.0.4, 172.20.0.15), showing massive flows, but more in than out, since web requests initiate from the ultimate receiver. Yet the blue points in the upper center are above the diagonal, also indicating massive flows, but more out than in. These are virtually all external attackers, showing their denial of service (DOS) attack activity.

Fig. 6 shows the same statistics for the port projection of the NetFlow graph, colored by ephemeral vs. non-ephemeral ports. Note the extreme outlier of port 80 in this log-log plot. This highlights the synthetic nature of this data being primarily focused on the generation of web traffic for a set of marketing companies. Real world data would be represented by different compositional features not represented in this data set. For our purposes, we are able to distinguish intent and usage of the network utilizing typical analysis of NetFlow information. Other ports which are prominent within the simulated BM environment include 25 (SMTP), 123 (NTP Network Time Protocol), 137 (NETBIOS Name Service), and 138 (NETBIOS Datagram Service).

Fig. 7 highlights a malicious attack on April 6 involving exfiltration of data leaving the organization destined for attacker 10.7.5.5, showing the details surrounding the aggregate communications per destination IP. Outgoing payload is on the horizontal, # flows on the vertical, and points are sized by incoming payload. When drilled down to the detailed data, this volume was transmitted during one flow making it an extreme outlier.

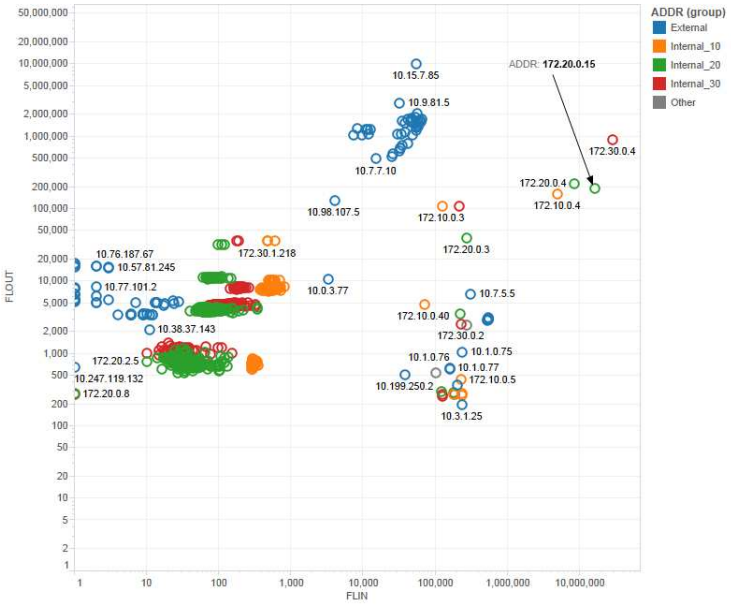

Figure 5: # flows in vs. out by IP, color coded by group.

Figure 6: # flows in vs. out by IP, color coded by port type.

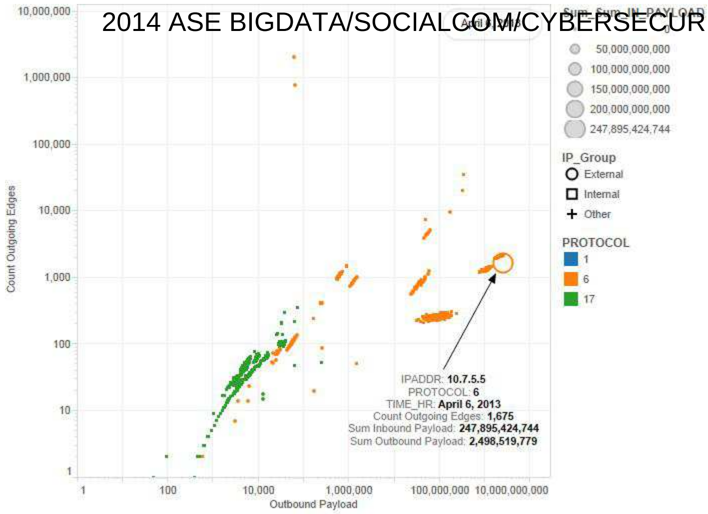

Figure 7: Outbound payload vs. # flows in, points sized by incoming payload.

5 Labeled Degree Distributions

Beyond the basic NetFlow graph statistics, we are interested in assessing the relations among NetFlow components: which IPs, ports, and their combinations have interesting, significant, or unusual relationships with each other? Does one IP talk to many, few? Among those, are they grouped or spread out?

In graph analysis, degree distributions are a standard measure, counting the number of incoming and outgoing edges over a set of nodes. We extend these methods to *labeled degree distributions* by additionally counting how those edges are distributed over a set of labels, for example the incoming or outgoing IPs or ports.

Fig. 8 illustrates an example for $N = 10$ flows. On the left, they're distributed into $m = 5$ IPs, with six flows in one IP and one in each of four other IPs. We notate this as a sorted count vector $\vec{C} = \langle C_l \rangle_{i=1}^m = \langle 6, 1, 1, 1, 1 \rangle$ of length $m = 5$. On the far right we show the case where those same 10 flows are distributed as $\langle 6, 4 \rangle$ over $m = 2$ IPs, while in the center it's $\langle 2, 2, 2, 2, 2 \rangle$ distributed uniformly over again $m = 5$ IPs. In all cases $\sum_{i=1}^m C_i = 10$, our number of flows.

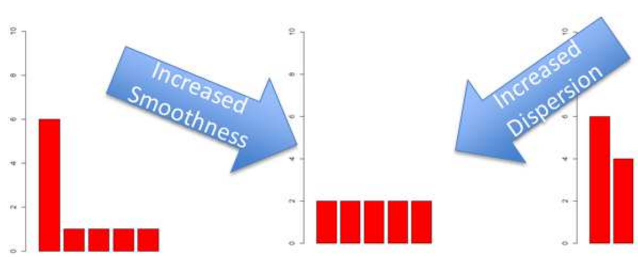

Figure 8: Three count vectors differently distributing $N = 10$ flows over IPs. From left to right $\kappa = 0.70, G = 0.76$, then $\kappa = 0.70, G = 1$, and finally $\kappa = 0.30, G = 0.97$.

Mathematically, our count vectors are partitions of the integer $N = 10$, whose properties we can measure by introducing two quantities. The **dispersion**

$$\kappa(\vec{C}) \; := \frac{\log_2(m)}{\log_2(N)}$$

measures the amount of coverage the N flows has over the m IPs. When $\kappa(\vec{C}) = 0$ then there is only $m = 1$ IP, while when $\kappa(\vec{C}) = 1$ then there a maximal $m = N$ number of IPs. But, given that N flows group into m IPs, they can do so either smoothly of "lumpily". The **smoothness**

$$G(\vec{C}) \; := \frac{\mathbf{H}(f(\vec{C}))}{\log_2(m)} = \frac{-\sum_{l=1}^m \frac{C_l}{N} \log_2\left(\frac{C_l}{N}\right)}{\log_2(m)}$$

is a group-relative entropy, with $G(\vec{C}) = 0$ when all flows go to $m = 1$ IP, while $G(\vec{C}) = 1$ when flows are distributed uniformly over any number m of IPs. Different values of κ, G are seen in Fig. 8. While $\kappa = 0 \leftrightarrow G = 0$ and $\kappa = 1 \leftrightarrow G = 1$, otherwise these measures are relatively independent.

These measures are useful in characterizing attacks and other simulation activity in the VAST NetFlow multigraph. Fig. 9 shows the dispersion and smoothness of outgoing flows over IPs, with coloring by group, and size showing the # of flows outbound. We can observe a number of significant features called out in the figure:

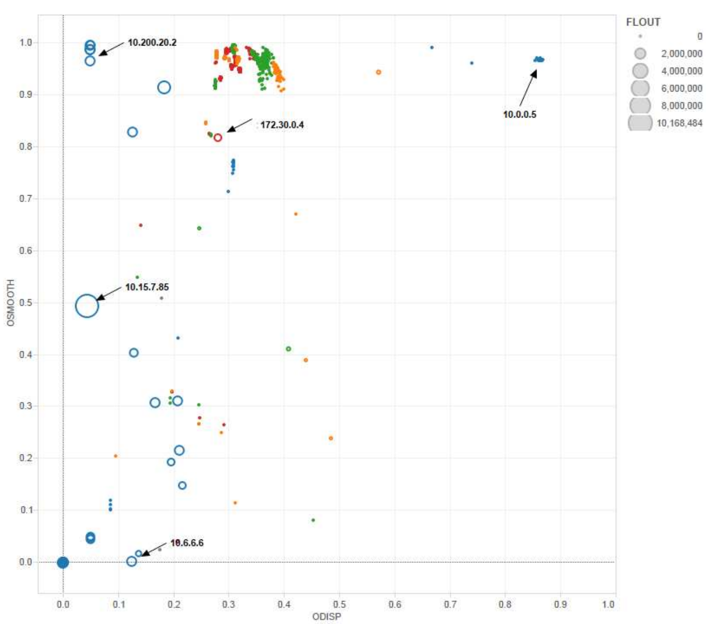

Figure 9: Dispersion (ODISP=κ and smoothness (OSMOOTH=G) of outgoing flows over IPs.

- BM's servers are somewhat visible by size, e.g. 172.30.0.4.
- The group in the far upper right (e.g. 10.0.0.5) are external servers in the simulated internet environment. Their maximal κ and G values, coupled with small flows, would indicate flows initiated from the servers back to the internal workstations, one per workstation. In fact, this analysis reveals an artifact of the simulation related to reversed time orders on flows, which would not otherwise be identified.

©ASE 2014 ISBN: 978-1-62561-000-3

- But most significantly are the external IPs scaled out on the left side. The group in the far upper left (e.g. 10.200.20.2) has high smoothness, 10.15.7.85 moderate, and 10.6.6.6 low. But these all have both a high number of flows and a low dispersion. These are, in fact, all attackers, and this signature seems to typify their activity, hitting a small number of targets (low dispersion) relative to the large number flows. But these attacks are distinguished by their smoothness: the lowest IP 10.6.6.6 attacks a single IP disproportionately, while the 10.200.20.2 participates in multiple attacks relatively evenly, although still vastly fewer than the millions of flows, so retaining a small dispersion. 10.15.7.85 is an intermediate case.

Fig. 10 shows the dispersion of input flows against the number of input flows for a particular day of a DOS attack. The attack against the three nodes at the top is clearly visible, with a massive number of incoming flows (log plot). But in addition to the large number of flows, the small dispersion indicates that they come from a small number of IPs, a signature of a DOS attack.

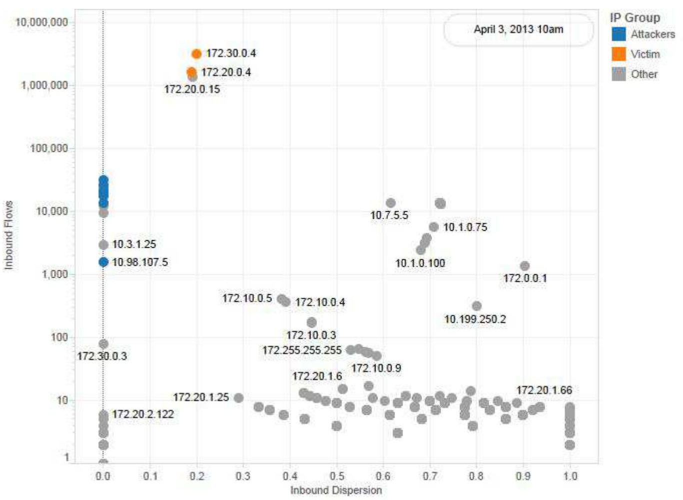

Figure 10: Dispersion (DISP_IN=κ) against # flows in by IP for a single day.

6 Time Interval Analysis

Our second novel approach to NetFlow traffic analayis is in the use of interval orders [4] and interval analysis to understand the temporal relationships between events. Let $\bar{x} = [x_*, x^*], \bar{y} = [y_*, y^*]$ be the time intervals for two NetFlows, so that $x^* - x_*$ and $y^* - y_*$ are their **durations**. We can also measure their **separation** $\|\bar{x}, \bar{y}\|$, not as a single number, but rather also as an interval between the minimal and maximal distance possible between any two points in either interval. $\bar{x}$ and $\bar{y}$ always sit in one of three basic relationships $\bar{x} \leq \bar{y}$,[3] which also determine how to calculate their separations (see Fig. 11):

Disjoint: We say $\bar{x} \leq_S \bar{y}$ when $x_* < x^* < y_* < y^*$. Then the separation ranges from $\alpha_* = y_* - x^*$ to $\alpha^* = y^* - x_*$.

[3]For simplicity, we assume that no two of the four endpoints involved are ever equal. Note also that there are in total six relationships, when also including the reverse $\bar{x} \geq \bar{y}$ of the three shown.

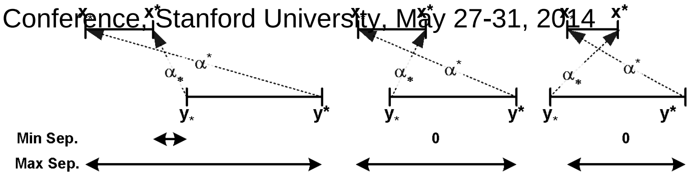

Figure 11: The three possible relationships between two intervals, and their minimum and maximum separation. (Left) Disjoint, $\bar{x} \leq_S \bar{y}$; (Center) Properly overlapping from the left, $\bar{x} \circ_{\leq} \bar{y}$; (Right) Contained, $\bar{x} \subseteq \bar{y}$.

Properly Overlapping From the Left: We say $\bar{x} \circ_{\leq} \bar{y}$ when $x_* < y_* < x^* < y^*$. Then the separation ranges from 0 to $\alpha^* = y^* - x_*$.

Contained: We say $\bar{x} \subseteq \bar{y}$ when $y_* < x_* < x^* < y^*$. Separation ranges from 0 to the maximum of $\alpha^* = y^* - x_*$ and $-\alpha_* = x^* - y_*$.

Fig. 12 shows a time line for one of the simulated attacks in the VAST NetFlow database occurring between 6 AM and 8 AM. In this case, a single external server 10.0.0.8 (blue) is DOS attacked by a collection of botnet-infected internal workstations (other colors). The top two graphs show the spikes in the number of flows incoming to the victim and outgoing from the attackers. The bottom two graphs show the durations of the flows coming from the attackers and their maximum separation.

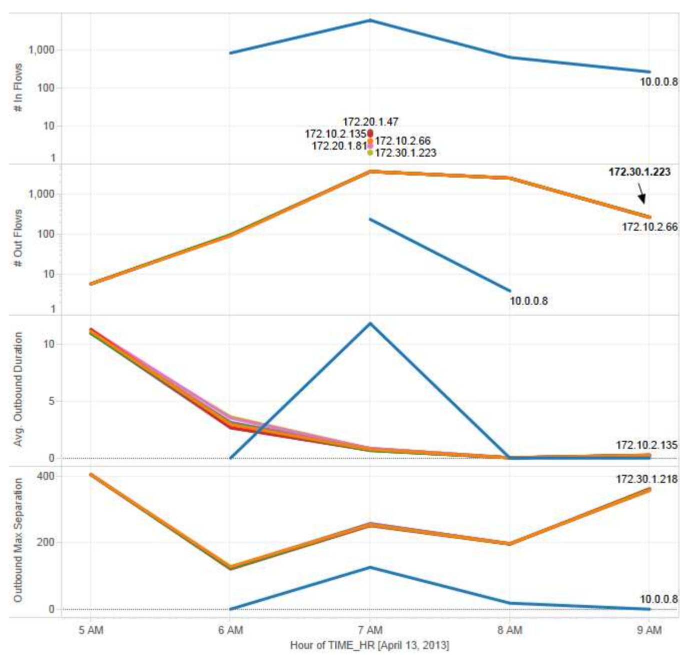

Figure 12: Time intervals of a DOS attack on 10.0.0.8. (Top) # incoming flows. (Top Center) # outgoing flows. (Bottom Center) Outgoing average duration. (Bottom) Average maximum time separation between outgoing flows.

Notice how attacker activity is generaly synchronized: other than the top plot, the lines of multiple infected workstations overlay each other. But also note a marked decrease

in the durations as the attack commences, and simultaneously a marked decrease in their average pairwise maximum separation, as shown qualitatively in Fig. 13. If these outgoing flows were unrelated, then simply reducing their widths may or may not change their separations, and *vice versa*. But the combination of decreasing duration and separation shows the increasingly coordinated nature of the attack: the botnet infection triggered the same processes in the infected workstations, leading to an even more *synchronized* attack.

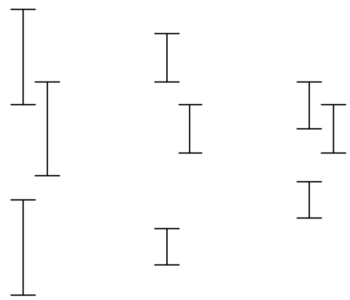

Figure 13: Qualitative representation of widths and separations of NetFlow time intervals of attackers. (Left) Before attack. (Center) If only durations were reduced, separations may increase. (Right) During attack: both durations and separations reduced, showing increased synchrony.

7 Computational Environment

We processed the VAST NetFlow data set in an environment built around a high performance Netezza TwinFin parallel SQL databases appliance, a unique asymmetric massively parallel processing (AMPPTM) architecture that combines open, blade-based servers and disk storage with data filtering using FPGAs. The Netezza allowed us to distribute flow data across all of the disks in the storage array, providing better query performance than a traditional relational database management system. This was complemented with SQL code, R, and the Tableau tool for exploratory data analysis.

8 Conclusion

As noted above, this paper presents our first results for these methods against simulated test data sets, revealing aspects of the simulated behaviors and features of the simulation itself. We are proceeding to build out both the mathematical, engineering, and application capability of this technology in a number of ways:

- Extensions to graph theoretical representations of cyber systems, other than NetFlow, at other scalar levels.
- Further mathematical and engineering development for both interval calculations and other approaches to combinatoric information theoretical measures.
- Examining data reduction methods, including more severe graph cube projections and Metcalf's encounter graph representations [8].
- Bringing these methods to bear on real data sets from operational situations.
- Deploying these measures to be used as features to be combined with others in machine learning applications.

References

[1] Baris Coskun, Sven Dietrich, and Nasir Memon. Friends of an enemy: Identifying local members of peer-to-peer botnets using mutual contacts. In *Proc. ACSAC 2010*, pages 131–140, 2010.

[2] Hristo Djidjev, Gary Sandine, Curtis Storlie, and Scott Vander Wiel. Graph based statistical analysis of network traffic. In *MLG 2011*, 2011.

[3] Nick Duffield, Patrick Haffner, B Krishnamurth, and Haakon Ringberg. Rule-based anomaly detection on ip flows. In *IEEE Infocom*, 2009.

[4] Peter C Fishburn. *Interval Orders and Interval Graphs*. Wiley, New York, 1985.

[5] Cliff Joslyn, Sutanay Choudhury, David Haglin, Bill Nickless, Bryan Olsen, and B Howe. Massive scale cyber traffic analysis: A driver for graph database research. In *Proc. 1st Int. Wshop. on GRAph Data Management Experiences and Systems (GRADES 2013)*, 2013.

[6] Thomas Karagiannis, Konstantin Papagiannaki, and Michalis Faloutsos. Blinc: Multilevel traffic calssification in the dark. In *SIGCOMM 05*, 2005.

[7] Alex Kent, Mike Fisk, and Eugene Gavrliov. Network host classification using statistical analysis of flow data. In *FLOCON 2010*, 2010.

[8] Leigh Metcalf. Analyzing flow using encounter complexes. In *FLOCON 2014*, 2014.

[9] David Moore, Geoffrey M Voelker, and Stefan Savage. Inferring internet denial-of-service activity. In *ACM Transactions on Computing Systems*, volume 24:2, pages 115–139, 2006.

[10] Shishir Nagaraja, Prateek Mittal, Chi-Yao Hong, M. Caesar and Nikita Borisov Botgrep: Finding p2p bots with structured graph analysis. In *Proc. 19th USENIX Conf. on Security*, 2010.

[11] Florian Tegeler, Xiaoming Fu, Giovanni Vigna, and Christophe Kruegel. Botfinder: Finding bots in network traffic without deep packet inspection. In *Proc. Co-NEXT 12*, pages 349–360, 2012.

[12] Arno Wagner and Bernhard Plattner. Entropy based worm and anomaly detection in fast ip networks. In *Proc. 14th IEEE Int. Wshop. on Enabling Technologies (WETOCE 05)*, pages 172–177, 2005.

[13] Peixiang Zhao, Xiaolei Li, Dong Xin, and Jiawei Han. Graph cube: On warehousing and olap multidimensional networks. In *SIGMOD 2011*, 2011.

©ASE 2014 ISBN: 978-1-62561-000-3