Network Forensics Analysis Using Piecewise Polynomials

SEAN MARCUS SANDERS
School of Electrical and Computer Engineering
Georgia Institute of Technology

The information transferred over computer networks is vulnerable to attackers. Network forensics deals with the capture, recording, and analysis of network events to determine the source of security attacks and other network-related problems. Electronic devices send communications across networks by sending network data in the form of packets. Networks are typically represented using discrete statistical models. Discrete statistical models are computationally expensive and utilize a significant amount of memory. A continuous piecewise polynomial model is proposed to address the shortcomings of discrete models and to further aid forensic investigators. Piecewise polynomial approximations are beneficial because sophisticated statistics are easier to perform on smooth continuous data , rather than on unpredictable discrete data. Polynomials, moreover, utilize roughly six times less memory than a collection of individual data points, making this approach storage-friendly. A variety of networks have been modeled, and it is possible to distinguish network traffic using a piecewise polynomial approach.

These preliminary results show that representing network traffic as piecewise polynomials can be applied to the area of network forensics for the purpose of intrusion analysis. This type of analysis will consist of not only identifying an attack, but also discovering details about the attacks and other suspicious network activity by comparing and distinguishing archived piecewise polynomials.

ADVISOR:
HENRY L. OWEN
School of Electrical and Computer Engineering
Georgia Institute of Technology

INTRODUCTION

Problem

Network forensics deals with the capture, recording, and analysis of network events to determine the source of security attacks and other network-related problems (Corey, 2002). One must differentiate malicious traffic from normal traffic based on the patterns in the data transfers. Network communication is ubiquitous, and the information transferred over these networks is vulnerable to attackers who may corrupt systems, steal valuable information, and alter content. Network forensics is a critical area of research because , in the digital age, information security is vital. With sensitive information such as social security numbers, credit card information, and government records stored on a network, the potential threat of identity theft, credit fraud, and national security breaches increases. During July of 2009, North Korea was the main suspect behind a campaign of cyber attacks that paralyzed the websites of US and South Korean government agencies, banks and businesses (Parry, 2009). As many as 10 million Americans a year are victims of identity theft, and it takes anywhere from 3 to 5,840 hours to repair damage done by this crime (Sorkin, 2009). In order to effectively prosecute network attackers, investigators must first identify the attack, and then gather evidence on the attack.

The process of identifying an attack on a network is known as intrusion detection. The two most popular methods of intrusion detection are signature and anomaly detection (Mahoney, 2008). Signature detection is a technique that compares an archive of known attacks on a network with current network traffic to discern whether or not there is malicious traffic. This technique is reliable on known attacks but has a great disadvantage on novel attacks. Although this disadvantage exists, signature detection is well understood and widely applied. Anomaly detection, on the other hand, is a technique that identifies network attacks through abnormal activity, which does not necessarily imply malicious traffic. Anomaly detection is more difficult to implement compared to signature detection because it must flag traffic as abnormal and discern the intent of the traffic. Abnormal traffic does not necessarily imply malicious traffic.

Electronic devices such as notebooks and cellular phones communicate by transferring data across the Internet using packets. A packet is an information block that the Internet uses to transfer data. In most cases, the data being transferred across the Internet must be divided into hundreds, even thousands of packets to be completely transferred. Similar to letters in a postal system, packets have parameters for delivery such as a source address and destination address. Packets include other parameters such as the amount of data being sent in a packet and a checking parameter to ensure that the data sent was not corrupted. The Internet is modeled as a discrete collection of individual data points because the Internet uses individual packets to transfer data. Discrete processes are difficult to model and analyze as opposed to continuous processes because there is not a definite link between two similar events. For example, the concept of a derivative in calculus can only give a logical result if the data is continuous. In many cases, experimental results are given as discrete values. Scientists, engineers, and mathematicians sometimes use the least squares approximation to give a continuous model of the data given. Continuous models that represent discrete data are often preferred because they can be used for different types of analysis such as interpolation and extrapolation.

Many forensic investigators use graphs and statistical methods, such as clustering, to model network traffic (Thonnard, 2008). These graphs and statistics help classify complex networks into patterns. These patterns are typically stored and represented in a discrete fashion because networks transfer data in a discrete manner.

These patterns are used in combination with signature and anomaly detection techniques to identify network attacks (Shah, 2006). In many cases these network patterns are archived and kept for extended periods of time. This storage of packets is needed to compare past network traffic with current network traffic, in order to effectively classify network events. Despite this necessity, the storage of packet captures is not desired because packet captures use a significant amount of memory storage, a limited and costly resource. After a variable amount of time, the archived network data is deleted to free memory for future network patterns to be archived (Haugdahl, 2007). Detailed records of network patterns can be stored for longer periods of time by increasing the amount of free memory or decreasing the amount of archived traffic.

A continuous polynomial representation of a network is preferred to a discrete representation because discrete representations are limited by the types of analysis and statistics that can be performed. Polynomial approximations of data have limitations as well, such as failing to represent exact behavior, which can be vital depending on the system being modeled. In order to effectively differentiate traffic, a continuous polynomial approximation must be robust enough to reveal enough details about network traffic. Polynomial representations of data should require less memory storage than discrete representations. For instance, the polynomial, $y=x^2$, could represent a million data points but take up little memory. This observation is important because, in the area of network forensics, memory storage space is a critical factor.

Related Work

Shah et al. (2008) applied dynamic modeling techniques to detect intrusions using anomaly detection. This particular form of modeling was only used for identifying intrusions and not for analyzing them or conducting a forensic investigation. Ilow et al. (2000) and Wang et al. (2007) both used modeling techniques to try to predict network traffic. Wang et al. took a polynomial approach that utilized Newton's Forward Interpolation method to predict and model the behavior of network traffic. This technique used interpolation polynomials of arbitrary order to approximate the dynamic behavior of previous network traffic. Wang et al.'s technique is useful for modeling general network behavior, but using the polynomial approach for intrusion analysis is another issue. Wang et al.'s technique proved that general network behavior can be predicted and modeled using polynomials, but did not prove whether individual network events can be distinguished and categorized through the use of polynomials.

Proposed Solution

Network data is discrete, scattered, and difficult to approximate; however, approximation and modeling techniques are necessary to define networks and to perform important statistics on the network data. Such statistics include the average amount of data each packet carries, the average rate packets arrive to a computer, and how many packets are lost before delivery. These values are used to adequately classify network traffic as normal or malicious. When a system is approximated as a polynomial, it is faster to perform basic mathematical operations and statistics such as derivatives, integrals, standard deviation, and variance. The ease of the computation of a parameter allows for a more efficient analysis of the data. Networks send an enormous amount of data each day, and precious time is required to process this data. While the polynomial approximation is fairly accurate, forming a long, a complex approximated polynomial is not practical for the purposes of network forensics since a network will seldom have identical behavior in each session. Assuming each of the five segments of points shown in Figure 1 represents network events (i.e., web sites visited), investigators can approximate and classify

the network activity. The network traffic modeled in all plots in this paper represents the same parameters. The x-axis represents the packet capture time, where the unit of time is not represented in seconds (i.e., real time) but rather a time relative to the order the packets were captured. In other words, time represented in the context of this paper does not represent real time, but serves as a parameter for the data being modeled, approximated packet data length. This parameter is referred to as time because the data being modeled is time dependent. The y-axis represents the data length of the packet captured in bytes. Throughout this paper the terms packet, capture time, and time will be used interchangeably. In reality, different network events require different amounts of time and numbers of packets than others. For simplicity, all network events plotted in this paper are scaled so that each network event is modeled by equal time intervals.

If these segments were in a different order (i.e., the same web sites were visited in a different order), then the single polynomial in Figure 2 would not be able to compensate for these changes and would be unable to efficiently classify similar network traffic. Essentially, if this single polynomial method were applied, one would need 120 (5!) different polynomials to represent visiting five different websites in every possible order. To counter this issue, the idea of approximating network traffic by using a piecewise polynomial is proposed. A single polynomial defines one function to represent data for all units of time, while a piecewise polynomial defines separate functions at distinct time intervals and connects these respective pieces to form a single continuous data representation. The property of a piecewise polynomial is important in modeling network traffic as opposed to a single polynomial because many different types of network events can occur. A piecewise polynomial can isolate and model the behavior of a single network event, while a single polynomial is limited

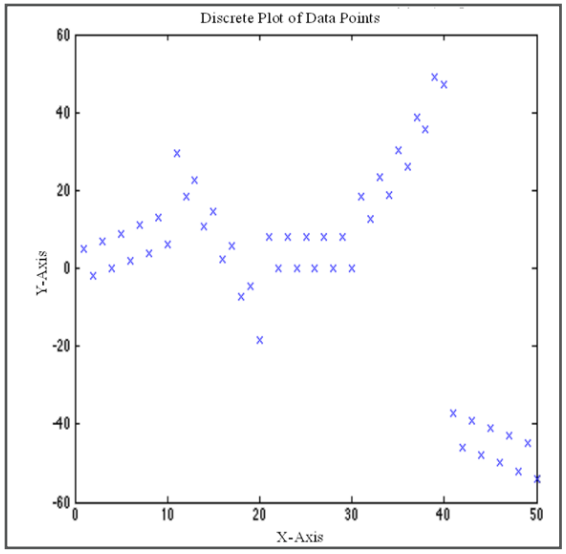

Figure 1. Plot of random discrete data.

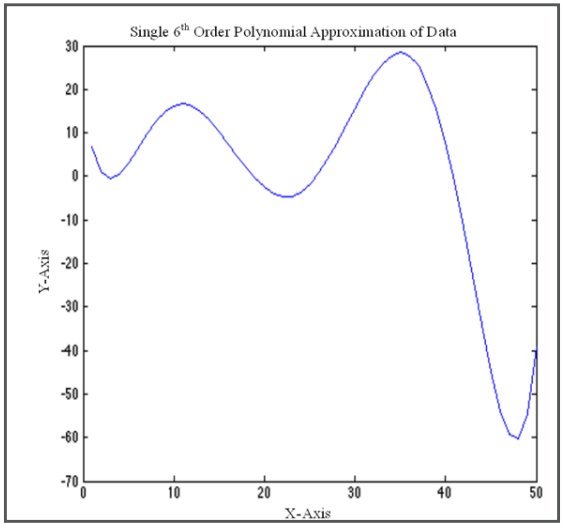

Figure 2. Single polynomial approximation of data represented in Figure 1.

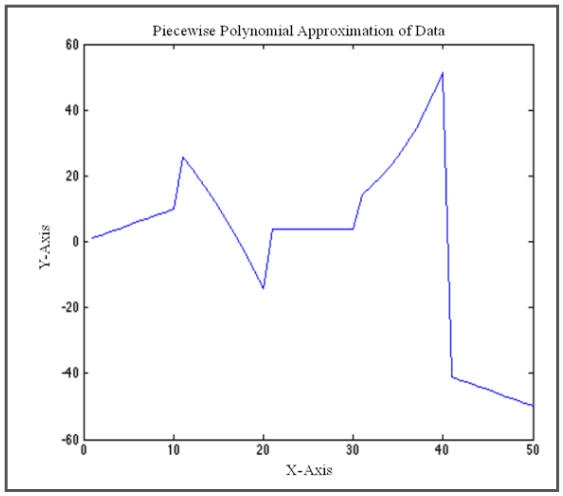

Figure 3. Piecewise polynomial plot of data represented in Figure 1.

to modeling clusters of events. The modeling of event clusters is not desired because it will increase the difficulty in differentiating network traffic based on a single event. Such a scenario will result in a malicious event being clustered with a normal event, which could lead to failure in identifying an attack. A piecewise polynomial approximation should effectively classify every network event that has transpired using a unique piecewise approximation. The piecewise polynomial approximation of the data shown in Figure 1 is shown in Figure 3.

It is clear that while both polynomial approximations in Figure 2 and Figure 3 can model the data represented in Figure 1, the piecewise polynomial (Figure 3) is more accurate and robust than a single polynomial. A single polynomial should not be used to model more than one network event, because it will not be able to represent the individual different network events that it is composed of. This example is meant to emphasize that if a sequence of 100 network events were defined using one single polynomial, it would be difficult to identify which network events behaved in a certain way. A piecewise polynomial model will address this issue by modeling each network event as an individual polynomial. If the order of the network events (segments) were changed, the individual polynomials would just occur at different time intervals, but each segment will remain the same. In other words, in a piecewise polynomial approximation each segment is represented by a distinct polynomial.

The basic concept is that while the network will not behave the same all the time, it will behave the same in certain pieces. If network traffic can be quantified using piecewise polynomials, investigators can apply signature and anomaly detection techniques to identify and investigate events from a forensics perspective. Piecewise polynomial approximations will be effective because they should approximate the behavior pattern of a network with enough resolution to differentiate network traffic.

The primary goal is to test whether or not a piecewise polynomial approach can approximate network data with enough precision to distinguish network traffic. If there are no distinct differences in piecewise polynomial approximated network traffic then this approach will not be valid for this application. Conversely, if a piecewise polynomial approximation can effectively differentiate network traffic then it can be applied to intrusion analysis, because intrusion analysis is primarily focused on classifying traffic. This application is beneficial because polynomial-represented data should occupy less memory storage than discrete data, and polynomial data have lessfewer limitations on the type of analysis that can be performed.

METHODOLOGY

Tools and Algorithms

Wireshark was used to capture network traffic in packet

capture files. A packet capture is a collection of the network traffic that has made contact with a computer and is stored in a packet capture file (.pcap file). Wireshark is an effective tool for capturing and filtering network traffic, but does not allow for a custom analysis of network traffic. The Libpcap library, which is used by Wireshark, was investigated in order to use the captured network traffic as an input to a custom parsing algorithm. This algorithm opens .pcap files that were saved using Wireshark and extracts the source address, destination address, packet data length, and packet protocol into a format that can be used for custom processing. After these aspects of the packet were extracted they were saved in a .csv file (comma separated files) for processing in MATLAB. Although the parameters initially extracted (source address, destination address, packet data length, and packet protocol) are not sufficient to analyze and detect all malicious activity, these parameters are a good starting point for a proof of concept implementation and analysis of this approach.

MATLAB was chosen for its versatility, variety of functions, and computing speed in processing large vectors. MATLAB has two built-in functions called Polyfit() and Polyval() that respectively compute polynomial coefficients and evaluate polynomials by using input data. In MATLAB, the input and output data of polyfit() and polyval() are represented as vectors. Polyfit() uses the least squares approximation to approximate the coefficients of a best fit, Nth-order polynomial for the given vectors of data: X and B. In statistics, the least squares approximation is used for estimating polynomial coefficients based on discrete parameters. Polyval() can best be viewed as a support function for Polyfit(), which gives the approximated numerical values of the polynomial approximated in Polyfit(),Y. A clearer example of how Polyfit() and Polyval() are related is shown in equation 1.

(1)

$$P = \text{polyfit}(X, B, N)$$
$$Y = \text{polyval}(P, X)$$

Piecewise.m is a custom-developed script, written in MATLAB. Essentially, Piecewise.m uses Polyfit() and Polyval() to create piecewise polynomials. This script was designed to use packet data lengths as the parameter on the y-axis, and packet capture time as the parameter on the x-axis.

IMPORTANT DECISIONS AND CAUSES FOR ERROR

An important parameter used to approximate the data is the order of the polynomial. Typically, the higher the order of the polynomial, the more accurate the approximation; in an approximation of network behavior/patterns, though, modeling exact behavior is unnecessarily complex whereas approximating behavior is more useful. Thus, the orders of the piecewise polynomials are manually chosen based on the predicted complexity of the network traffic. More complex traffic should be approximated with a higher order polynomial than less complex traffic. This is an assumption that will be used to designate the order of a polynomial given the type of network being modeled. Network traffic was also modeled using different orders to determine the effect(s) that changing orders have on the approximation of traffic.

When approximating polynomials, ensuring that there are enough data points to create a reliable approximation is important. For example, if there is one data point a first order polynomial would give an inaccurate approximation, because at least two points are needed to approximate a line. The general rule is that the accuracy of the polynomial approximation depends directly on the order of the polynomial, and number of data

points used to define the polynomial. The number of data points must be at least one more than the desired order to yield an accurate polynomial approximation. In most cases, the higher the order of the polynomial, the more accurate the approximation is. On the other hand, a polynomial of too high of an order may yield unrealistic results. Thus finding a balance of polynomial order that yield both of approximate and realistic results is important.

EXPERIMENTS

Closed/Controlled Network Behavior

The first step to determine whether a polynomial can accurately approximate and differentiate network behavior is to analyze the behavior of a closed/controlled network. As opposed to open networks, closed networks are not connected to the Internet. The designed closed network was composed of two Macbooks, with four virtual machines operating on the separate Macbooks. Figure 4 gives a visual representation of the designed closed network.

A virtual machine is a software implementation of a machine that executes programs like a physical machine. Virtual machines operate on a separate partition of a computer and utilize their own operating system. Due to the hardware limitations of physical machines, virtual machines and physical machines do not execute commands simultaneously. From a networking perspective the execution of commands is not a problem, because once connected, networks utilize protocols to send and sometimes regulate the flow of network traffic. In other words, the network does not know that there is a virtual machine operating on a physical machine and thus supports multiple simultaneous network connections.

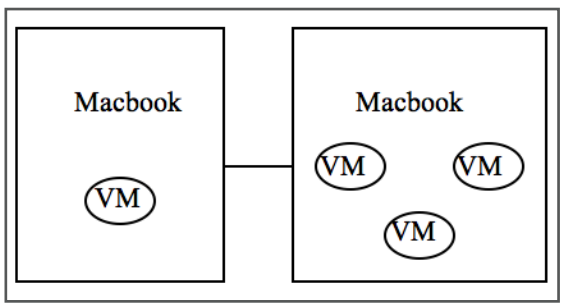

Figure 4. Visual representation of designed closed network with virtual machines. VMS circled.

Packet captures were performed using Wireshark on the Macbook operating with three virtual machines on the ethernet interface. A variety of packet captures were made to compare and contrast network behavior using web pages. If the resulting piecewise polynomials could effectively compare and contrast network traffic based on various behaviors, then the polynomial approximation will be considered a success. The descriptions of these packet capture files are listed below.

- Idleclosed.pcap— a .pcap file that captures the random noise that is captured when the network is idle.
- Icmpclose.pcap— a .pcap file that is composed primarily of ping commands from one Macbook to the other. Ping commands are used to test whether a particular computer is reachable across a network. This test is performed by sending packets of the same length to a computer, and waiting to receive a reply from that computer.
- Httpclose.pcap— a .pcap file that includes a brief ping command being sent from one Macbook to the other Macbook, but is dominated by HTTP traffic (basic website traffic). This file also includes a period of idle behavior where the network is at rest.
- Packet Capture A— a .pcap file that contains the network data for visiting a specific site hosted on one Macbook.

• Packet Capture B— a separate .pcap file that contains the network data for visiting the same site visited in Packet Capture A hosted on the same Macbook at a different time.

Idleclosed.pcap and Icmpclose.pcap yield piecewise polynomials that model the behavior of idle and ping traffic respectively. These piecewise polynomials should identify both the idle and ping behavior found in Httpclose.pcap. The piecewise polynomials that model two separate .pcap files going to the same pages (i.e. Packet Capture A and Packet Capture B) should resemble each other in behavior. A second order piecewise polynomial is used for the closed network analysis because it is assumed that closed network events should not be extremely complex. Higher orders are avoided wherever possible due to reasons explained in Important Decisions and Causes for Error.

Open/Internet Network Behavior

While experimenting with a controlled network is useful, a network that is connected to the Internet will behave differently from one that is not. To investigate a more realistic scenario, one Macbook was utilized to make different packet captures under similar conditions to those in Closed/Controlled Network Behavior, but with contact to the Internet. The details of the packet capture files are listed below.

• Internet.pcap— a .pcap file that contains network data captured while actively browsing the Internet.

• Packet Capture C— a .pcap file that contains the network data for visiting a sequence of three web sites on the Internet in a particular order (google.com, gatech.edu, and facebook.com).

• Packet Capture D— a separate .pcap file that contains the network data for visiting the same web sites as Packet Capture C but in a different order (gatech.edu, facebook.com, and google.com)

Internet.pcap was used to show the effect the order of a polynomial has on the approximation because it contains the most complex network traffic. Packet Capture C and Packet Capture D were used to determine if different web sites exhibit distinguishable behavior by using by piecewise and single polynomials. These models will test the theory of the benefit of piecewise polynomials over single polynomials similar to the example inthe Proposed Solution. Fourth order piecewise and single polynomials are used for the open network analysis, as opposed to second order, because it is assumed that open network events should be more complex than closed networks.

RESULTS

Closed Network Analysis

Ping Analysis In the closed network case, as defined in the Closed/Controlled Network Behavior, Httpclose.pcap and Icmpclose.pcap both contained the same type of ping traffic going through the network but in different packet captures and different times. The resulting piecewise polynomial that described this traffic in both packet captures was the constant 98. This constant value of 98 represents that every packet captured had a packet data length of 98 bytes. A constant piecewise polynomial is an acceptable value because the ping command constantly sends packets of identical lengths to a single destination.

Traffic Analysis Packet Capture A and Packet Capture B are two different .pcap files that were capturing the same network activity at approximately the same time interval, and are represented as second order piecewise polynomials. According to Figure 5, the two packet captures are represented in a very similar manner. This result is interesting because, while the results are similar, they are not exact. This mismatch is not damaging as Figure 5 shows the relationship of the two data files. The

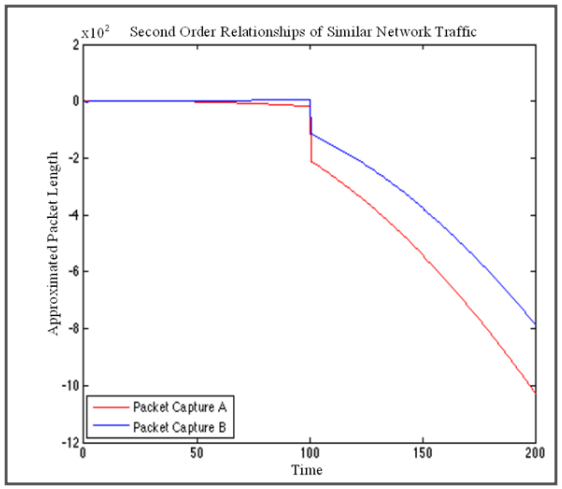

Figure 5. Second order relationship of similar packet capture files.

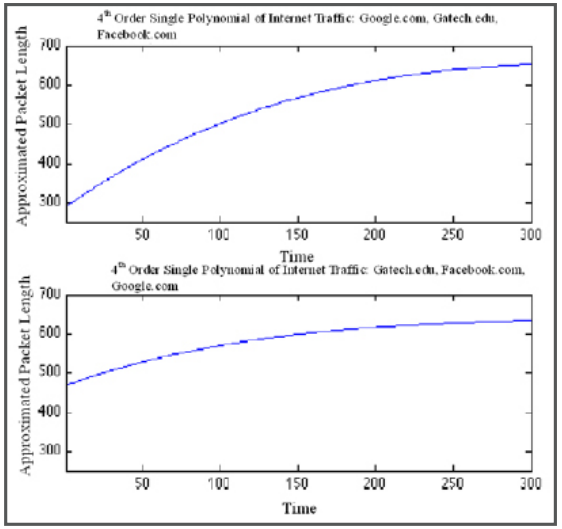

Figure 6. Single polynomial comparison plots of similar out of order traffic.

relationship of the first segments of data is that they are constant around the same value, while the second segments of the data are both decreasing, concave down, and share similar values.

Traffic Analysis of Open Networks

The similar packet capture files, Packet Capture C (the upper plot) and Packet Capture D (the lower plot), were plotted in Figure 6 using a fourth order single polynomial . Figure 7 shows the plot of Packet Captures C and D using a fourth order piecewise polynomial.

Packet Capture C visits google.com first, followed by gatech.edu, and ends with facebook.com, while Packet Capture D visits gatech.edu first, followed by facebook.com, and ends with google.com. Figure 7 shows that each piecewise polynomial gives each website visited a unique behavior that can be identified with visual inspection.

Google.com behaves in a sinusoidal type manner, Gatech.edu is represented as concave down parabola, and Facebook.com exhibits a strong linear behavior with a small positive slope. Although, the three web sites visited can be clearly identified in Figure 7, it does not seem to be the case in Figure 6.

In the single polynomial approximation the data looks relatively similar , and it is difficult to discern which part of the polynomial represents which website. This result shows that different network events can be approximated and distinguished using a piecewise polynomial approach, whereas a single polynomial approximation is not sufficient to distinguish network events.

Significance of Order

Internet.pcap was plotted using zero, second, and fifth orders to discern the effect order has in the approximation of a polynomial.

Figure 8 shows that the higher the order of the polyno-

mial, the more detail is shown about the network. Despite revealing more details of a network, Figure 8 does not show which order of the polynomial yields better results. Figure 8 is shown to illustrate the effect order has on the approximation of network traffic. More details are not necessarily better, because too many details may not yield an approximation that is robust enough to identify similar future network traffic and is difficult to interpret.

Memory Savings

Internet.pcap was saved in two separate files. One file was saved using Internet.pcap's polynomial representation, and a separate file was saved using Internet.pcap's representation as a collection of individual data points (i.e., packets). The polynomial file was 12Kb large while the collection of individual data point file was 72Kb large. This size difference indicates that saving network traffic as polynomials instead of a collection of individual points saves memory.

DISCUSSION OF RESULTS

The plots in Results are intended to show whether piecewise polynomials can effectively differentiate and link network traffic. The ping traffic analyzed in Ping Analysis was approximated by piecewise polynomials that exhibited constant behavior. Although this result is desired, ping traffic is the simplest type of network traffic and is not sufficient enough to prove the validity of a piecewise polynomial approach. Traffic analysis of the closed network yielded similar results to the ping analysis, by successfully differentiating and linking network traffic. Although the closed network analysis was a success, in reality most network traffic occurs on the Internet. Thus the open network results are of primary interest.

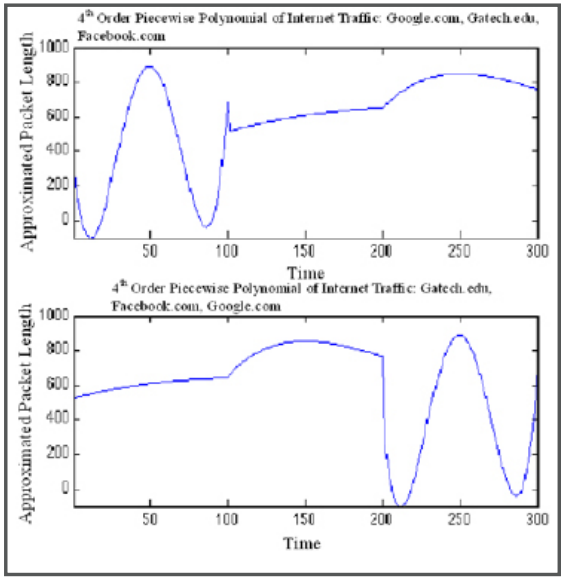

Figure 7. Piecewise polynomial comparison of similar out of order traffic.

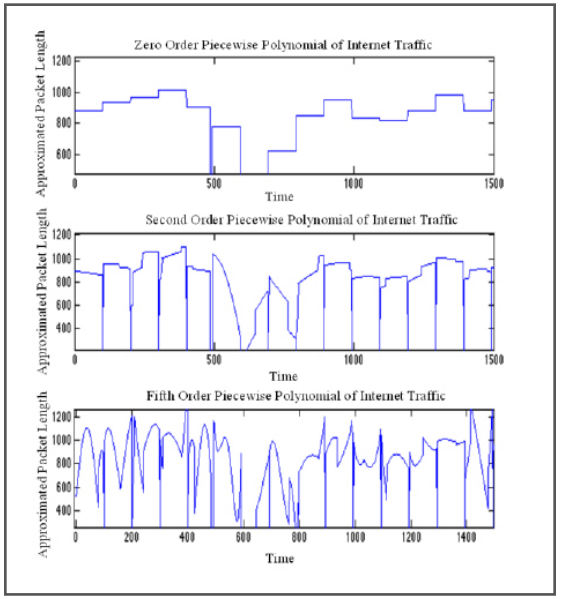

Figure 8. Internet pcap plots of varying orders.

The open network single polynomial approximation was unable to differentiate and link network events, as shown in Figure 6. The plot given in Figure 6 shows two similar curves of different ordered network traffic. Although this result is not desired, it was expected that a single polynomial approximation would not be able to classify out of order traffic effectively. Conversely, Figure 7 shows that a piecewise polynomial approximation was able to distinguish each section of the network traffic that was captured. These results show that a piecewise polynomial approximation can be used to classify and differentiate network traffic.

Memory storage is also of primary concern when modeling network data. The Internet packet capture shows that the discrete representation of data utilized 72Kb of memory storage, while the polynomial representation utilized 12Kb of memory storage. This result shows that polynomial processes utilize roughly six times less memory storage than discrete processes. This size difference indicates that storing network traffic as polynomials instead of a collection of individual points significantly saves memory. This outcome is important in network forensics because network events can be archived for a longer amount of time than before. This extra storage allows for more extensive and detailed investigations.

CONCLUSION

Networks can be approximated using piecewise polynomials with enough detail to aid forensic investigators. The precision of the approximation depends directly on the order of the polynomial used to approximate the data. In general, the higher the order the more details are revealed. Networks behave differently and therefore every network analyzed needs its own set of polynomials to approximate their respective network events. The use of piecewise polynomials is also beneficial because polynomials use roughly six times less memory than individual data points.

FUTURE WORK

Piecewise polynomials will be applied to the area of network forensics for intrusion analysis. This analysis will require collection of known data that are classified as either malicious or normal. Also, more information about packets will have to be quantified, to further classify and to distinguish network traffic because approximating packet length and protocols are not sufficient to perform a thorough analysis. The malicious data will be modeled as piecewise polynomials and used for signature detection. The normal network traffic will also be modeled as piecewise polynomials and used for anomaly detection.

Future research also includes identifying what certain traffic patterns represent, such as web browsing traffic, video streaming traffic, or file downloading traffic. This classification of network events will enhance a forensics investigator's ability to quickly determine what events have transpired on a network.

ACKNOWLEDGEMENTS

This research was conducted with the guidance of Kevin Fairbanks and Henry Owen and supported in part by a Georgia Tech President's Undergraduate Research Award as a part of the Undergraduate Research Opportunities Program. This research was also supported in part by the Georgia Tech Department of Electrical and Computer Engineering's Opportunity Research Scholars Program.

REFERENCES

Vicka C, Peterman C, Shearin S, Greenberg MS, Bokkelen J, (2002)Network Forensics Analysis, IEEE Internet Computing, http://computer .org/internet/, December 2002

Scott HJ, (2007)Network Forensics: Methods, Requirements, and Tools, www.Bitcricket.com, November 2007

Ilow J, (2000)Forecasting Network traffic using FARIMA models with heavy tailed innovations, Proceedings of the Acoustics, Speech, and Signals Processing, 2000. On the IEEE International Conference-Volume 6, IEEE Computer Society, Washington DC, pp. 3814-3817

Mahoney MV, Chan PK, (2008) Learning Models of Network Traffic for Detecting Novel Attacks, Florida Institute of Technology, www.cs.fit.edu/~mmah oney/paper5.pdf

Parry RL, (2009) North Korea Launches Massive Cyber Attack on Seoul, The Times, http://www.timesonline.co.uk/tol/news/world/asia/article6667440.ece, July 2009

Shah K, Jonckheere E, Bohacek S, (2006) Dynamic Modeling of Internet Traffic for Intrusion Detection, EURASIP Journal on Advances in Signal Processing Volume 2007, Hindawi Publishing Corporation, May 2006

Sorkin, (2009) Identity Theft Statistics, spamlaws.com

Thonnard O, Dacier M, (2008) A framework for attack pattern discovery in honeynet data, The International Journal of Digital Forensics and Incidence Response, Science Direct, Baltimore, Maryland, August 2008

Wang J, (2007)A Novel Associative Memory System Based Modeling and Prediction of TCP Network Traffic, Advances in Neural Networks, Springer Berlin, July 2007