ARL-TR-7614 • Mar 2016

US Army Research Laboratory

A Proposal for Kelly Criterion–Based Lossy Network Compression

by Sidney C Smith and Robert J Hammell II

NOTICES

Disclaimers

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.

ARL-TR-7614 ● MAR 2016

US Army Research Laboratory

A Proposal for Kelly Criterion–Based Lossy Network Compression

by Sidney C Smith
Computational and Information Sciences Directorate, ARL

Robert J Hammell II
Department of Computer and Information Sciences, Towson University

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED (From - To)
March 2016	Final	August 2014-August 2015

4. TITLE AND SUBTITLE
A Proposal for Kelly Criterion–Based Lossy Network Compression

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Sidney C Smith and Robert J Hammell II

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
US Army Research Laboratory
ATTN: RDRL-CIN-S
Aberdeen Proving Ground, MD 21005-5066

8. PERFORMING ORGANIZATION REPORT NUMBER
ARL-TR-7614

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES
primary author's email: <sidney.c.smith24.civ@mail.mil>.

14. ABSTRACT
This proposal describes the development of a Kelly criterion–inspired compression algorithm to be used in distributed network intrusion detection applications. Most of these applications only send alerts to the central analysis servers. These alerts do not provide the forensic capability that the analysts require to determine if this is an actual or attempted intrusion. Standard lossless compression algorithms do not reduce the size of the traffic enough to prevent negatively impacting the site. Kelly's algorithm instructs a gambler how much to bet based upon the chance of winning and the potential payoff. There has been a significant amount of research into anomaly detection algorithms that will provide some indications of the maliciousness of a network session. We propose to combine expert knowledge, data mining, and best of breed anomaly detection algorithms to determine the likelihood that a session is malicious or the chance of "winning". Further, we propose using a Kelly criterion–inspired algorithm to select the amount of bandwidth or "wealth" for each session. We expect that this will minimize the total amount of traffic we transmit while maximizing the amount of malicious traffic we transmit.

15. SUBJECT TERMS
Kelly criterion, data mining, anomally detection, intrusion, compression, lossy compression

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Sidney C Smith
Unclassified	Unclassified	Unclassified	UU	30	19b. TELEPHONE NUMBER (Include area code) 410-278-6235

Standard Form 298 (Rev. 8/98)
Prescribed by ANSI Std. Z39.18

Contents

List of Figures

1. Introduction

Distributed Network Intrusion Detection Systems (NIDS) allow a relatively small number of highly trained analysts to monitor a much larger number of sites; however, they require information to be transmitted from the remote sensor to the central analysis system (CAS). Unless an expensive dedicated NIDS network is employed, this transmission must use the same channels that the site uses to conduct daily business. This makes it important to reduce the amount of information transmitted back to the CAS to minimize the impact that the NIDS has on daily operations.

One possible solution is to do the processing on the sensor and transmit only alerts. This solution has 3 serious problems. The first is that a sensor with a central processing unit (CPU) overburdened with running analysis algorithms will not be able to capture all of the packets that traverse the network. Smith et al. discussed the impact of this packet loss.[1,2] The second is that the alerts alone seldom provide the analyst with the information necessary to determine if the attack was successful. The third is that signature based NIDS are ineffective at detecting zero-day or advanced persistent threat attacks.

Another possible solution is to do very little processing on the sensor and transmit all of the information captured to the CAS. This frees the sensor CPU and provides the analyst with the required information but doubles the traffic on the network since every packet captured must be transmitted. Lossless compression is a possible solution; however, this introduces additional latency, and the best lossless compression algorithms are still not able to reduce the impact to daily operations enough.

One reasonable alternative is to use lossy compression as a solution. This solution introduces its own set of problems, as it requires a sound method to determine the likelihood that traffic is malicious and the ability to take that likelihood and determine what data must be transmitted and what data are safest to lose. The focus of the proposed work is to solve these problems.

In 2004, Kerry Long described the Interrogator Intrusion Detection System Architecture.[3] In this architecture, remotely deployed sensors, known as Gators, collect network traffic and transmit a subset of the traffic to the analysis level.[3] Interrogator employs “a dynamic network traffic selection algorithm called Snapper.”[3] The proposed effort will build on the work done with Interrogator to add an intelligent

lossy compression algorithm to the Snapper functionality.

This intelligent lossy compression algorithm will employ expert knowledge, data mining, and best of breed anomaly detection algorithms to assign a maliciousness score to each session. It will take this maliciousness score and feed it into a Kelly criterion[4]–inspired selection formula to determine how much traffic from each session to transmit to the CAS.

The remainder of this proposal is organized into the following sections. Section 2 will provide some background. Section 3 provides a preliminary literature review. Section 4 will present a complete definition of the problem. Section 5 will enumerate the objectives of this research. Section 6 will enumerate our research questions. Section 7 will outline the approach chosen to address this problem. Section 9 will explain the results that we expect and how we will analyze them to determine the effectiveness of the algorithm. Finally, Section 10 will provide a brief summary of the proposal.

2. Background

To implement NIDS, we must have some way to bring the relevant traffic back to the CAS. One popular strategy for implementing a distributed NIDS is to do all of the intrusion detection on the sensor and send only alerts to the CAS.[5,6] A second strategy might be to use lossless compression to reduce the size of the data returned to the CAS. A third strategy is to implement some form of lossy compression algorithm to send back relevant portions of traffic.

There are 3 problems with sending only alerts to the CAS. The first is that it has the potential to overburden the sensor's CPU and introduce packet loss. The impact of this packet loss has been discussed by Smith et al.[1,2] The second problem is that the alerts by themselves often do not contain enough information to determine whether the attack was successful. The third problem is that these systems are most often implemented with signature-based intrusion detection engines. Signature-based systems may be tuned to produce few false positives; however, they are ineffective at detecting zero-day and advanced persistent threats.[7]

The second alternative presented several algorithms for lossless compress; however, one of the most widely used is deflation, which is a variation of the LZ77

algorithm described by Ziv and Lempel.[8] Compressing the 2009 Cyber Defense Exercise dataset[9] with gnuzip provides a ratio of 56.4%. To minimize the impact of NIDS on day-to-day operations, compression ratios of less than 10% are required. Lossless compression alone will not provide a reasonable solution.

The concluding reasonable alternative is to use some sort of lossy compression strategy to provide a solution. We may consider network traffic to be composed of sessions that span spectrums from known to unknown and malicious to benign, as illustrated in Fig. 1. Quadrant III, the known malicious quadrant, is the domain of intrusion prevention systems as described by Ierace et al.[10] Specifically, the far lower-left corner of quadrant III, or the intrusion prevention systems, may inflict a denial of service attack upon the systems that they are protecting. We are most interested in quadrant II, the unknown malicious quadrant, because that is the quadrant where we will find evidence of zero-day and advanced persistent threat attacks. We assume that malicious traffic makes up a small amount of the actual traffic on the network. In 2004, Kerry Long described the Interrogator Intrusion Detection System Architecture.[3] In this architecture, remotely deployed sensors, known as Gators, collect network traffic and transmit a subset of the traffic to the analysis level. Interrogator employs "a dynamic network traffic selection algorithm called Snapper."[3] Long and Morgan describe how they used data mining to discover known benign traffic that they excluded from the data transmitted back to the analysis servers.[11]

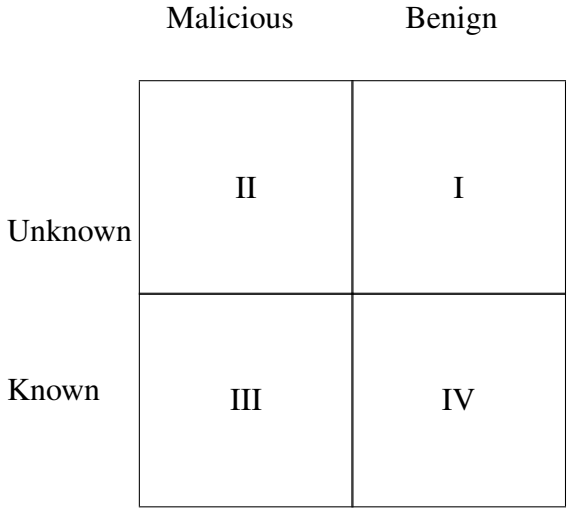

Fig. 1 Network traffic composition

In this research, we propose to combine expert knowledge, data mining, and best of breed anomaly-based NIDS solutions to compute a maliciousness factor. We then propose to feed this malicousness factor into a Kelly criterion[4]–inspired algorithm to compute the amount of traffic in each session that will be transmitted to the CAS. This should produce a lossy compression of the network traffic designed to reduce the amount of benign traffic and maximize the amount of malicious traffic being sent to the CAS.

3. Preliminary Literature Review

This research is broken down into to 2 basic questions: 1) how to rate the maliciousness of traffic and 2) how to use this rating to decide how much of each session to send back to the CAS. We will answer the first question by exploring expert knowledge, data mining, and anomaly detection solutions. We will answer the second question by exploring the application of the Kelly criterion.

3.1 Session Rating

3.1.1 Data Mining

Lee and Stolfo used RIPPER[12] on tcpdump[13] data in their paper, "Data Mining Approaches for Intrusion Detection."[14] The dataset they used from the Information Exploration Shootout[15] contained only the header information for the network traffic and no user data. Lee and Stolofo cooked the network traffic down into records that look very much like Cisco netflow[16] records. Then they were able to feed this information into RIPPER to generate rules. Their initial efforts were unsuccessful; however, once they added a time window into their analysis, they were able to achieve promising results. Since their data only contained Internet Protocol (IP) header information, and the positions of the exploits were not available to them, they were not able to assess the accuracy of their results.

While developing the Intelligent Intrusion Detection System at Mississippi State University, Bridges et al. integrated fuzzy logic, association rules, and frequency episodes data mining techniques to increase the flexibility of the system.[17] Genetic algorithms were employed to tune the membership functions of the fuzzy logic.[18]

Dokas et al. addressed the problem of skewed class distribution in mining data for network intrusion detection that exists because malicious activity compromises less than 2% of the network traffic by applying several boosting strategies to classifica-

tion algorithms for rare classes as part of the data mining in Minnesota Intrusion Detection System (MINDS).[19]

In ARL-TR-4211, *Using Basic Data Mining Techniques to Improve the Efficiency of Intrusion Detection Analysis*,[11] Long and Morgan describe mining the Interrogator database to discover known benign traffic to be excluded from the traffic transmitted to the CAS. Their strategy was to exclude the most common day to day traffic flowing to and from the most popular trusted sites.[11]

3.1.2 Anomaly–Based Network Intrusion Detection

In their history and overview of intrusion detection, Kemmerer and Vigna confirm a long-standing belief that although anomaly detection techniques are capable of detecting unknown attacks, these techniques pay for that capability with a high false positive rate.[7] In traditional NIDS, high false positive rates drain valuable time for the analysts. In this application, false positives simply increase the amount of traffic transmitted. This is a cost to be considered; however, it is a much smaller price to pay than that paid by generating an alert for someone to analyze. This means that a significantly higher false positive rate can be tolerated in this application, making algorithms that would be unusable for detection attractive for rating the likelihood that traffic is malicious. There has been a significant amount of work using anomaly detection in NIDS applications. Garcia-Teodoro et al. reviewed various types of anomaly-based detection techniques, categorizing them as either statistics based, knowledge based, or machine learning based.[20]

In 1994 Mukherjee et al. provided a survey of intrusion detection technology titled "Network Intrusion Detection."[21] By today's standards the title is somewhat deceiving because almost all of the systems they surveyed are what would now be called host-based intrusion detection systems. These systems tend to examine the individual system's audit logs looking for intrusive activity. The notable exception is Network Security Monitor (NSM). NSM employs a System Description Language, which is roughly modeled after a programming language and is used to describe the complex relationship that may be inferred from observable objects. These complex objects are analyzed using behavior-detection functions. NSM implements isolated object analysis and integrated object analysis.[22–24]

Sekar et al. describe their experiences with specification-based intrusion detection. They created a behavioral monitoring specification language and compiled it into

detection engines,[25–27] validating their approach using the Defense Advanced Research Project Agency (DARPA) dataset.[28]

Eskin et al. describe an unsupervised anomaly detection framework where network connections are mapped to a feature space and either cluster-based, k-nearest, or support vector machine–based algorithms are used to find anomalies in the sparse spaces. One of the key advantages to their approach is that it does not required labeled or known normal data to train the engine.[29]

Kruegel et al. developed a service-specific anomaly detection engine.[30] This engine contained a packet processing unit and a statistical processing unit. The packet processing unit pulled packets from the network and reassembled them into service requests. The statistical processing unit measured the type of request, length of request, and content of the request. It then computed values that ranged from 1 to 15 for each of these aspects, such that greater deviation translated into higher numbers. These values were then combined to provide an anomaly score. This score was compared against a standard that the author suggested should be set so that the system produces no more than 15 false positives a day. Because the deviation in type, length, and content varies significantly between services and even the types of requests, the statistical data must be partitioned by service and the length and content by type; however, the algorithms may be used without change by any service. However, the packet processing unit may need to be adjusted per service.[30]

Ertoz et al. describe the MINDS.[31–33] MINDS uses Cisco Netflow[16] data to collect statics for 16 different features, half observed and half computed for each session. For each session the local outlier factor is computed. Sessions with features that contain very large local outlier factors are considered anomalous. Sessions then undergo associated pattern analysis, which provides a summary of highly anomalous traffic for the security analyst.[31]

Munz et al. describe anomaly detection using K-means clustering.[34] Similar to Mukherjee et al., they separate the analysis for each service or port. Similar to Ertoz et al., they work with Cisco Netflow data.[16] Unlike the solutions mentioned above, this one requires both normal and attack training data to establish initial clusters. New traffic is then compared to the established clusters.[34]

Yassin et al. describe an approach that combines K-means clustering and naive

Bayes classification called KMC+NBC. They were able to validate their algorithm against the ISCX 2012 Intrusion Detection Evaluation Dataset[35] with strong positive results.[36]

3.2 Session Selecting

In 1956 while working for Bell Telephone Laboratories, Kelly was developing a way to assign a value measure to a communication channel.[4] He described a hypothetical illustration of a gambler who received advanced notice about the outcome of an event through a communication channel with a non-negligible error rate. By doing this, Kelly was able to assign a cost value to the communication, achieving his original goal. At the same time, he developed a formula based upon the probability of winning and the rate of payoff that would provide an amount to bet l that, if bet consistently over time, would achieve and maintain greater wealth than any other value of l. We saw this in Eq. 1, where l is the fraction of wealth to bet, p is the probability of winning, and b is the net odds of the wager.[4]

$$l = \frac{bp - q}{b} = \frac{p(b+1) - 1}{b}. \tag{1}$$

Breiman uses the Kelly's work while discussing optimal gambling systems.[37] He considers the problem of how much to bet on a series of biased coin tosses. To maximize returns on each toss, one would bet his or her entire fortune; however, this will ultimately ensure ruin. To maximize winning and avoid ruin, some fixed fraction of wealth will be bet at each iteration. He uses Kelly's work to discover that fixed fraction.[37]

Thorp first wrote about applying mathematical theory to the game of Blackjack in the 1960 paper "Fortune's Formula: The Game of Blackjack".[38] Later Thorp published the book *Beat the Dealer*, where he referred to what he called "The Kelly Gambling System".[39] Although he mentions using the Kelly criterion as the optimal way to bet in his research for *Beat the Dealer* in his later work,[40] he mentions it only once in passing in this book.[39] The bulk of his book discusses the rules of Blackjack and methods to determine when one has an advantage over the dealer and how great that advantage might be. The Kelly criterion would be used to calculate how large of a bet to place based upon the size of the advantage. Instead of directly using the Kelly criterion, he talks about placing big bets and little bets.[39] In his paper "Under-

standing the Kelly Criterion", Thorp mentions the application of the Kelly criterion to the stock market and his previous book *Beat the Market*[40]; however, the Kelly criterion is not mentioned at all in *Beat the Market*. Instead, Thorp concentrates on how the market works, what short selling and warrants are all about, and how to determine the relative value of a stock or a warrant.[41] Thorp goes into greater detail about how the Kelly criterion would be used in Blackjack and the stock market in his paper "Optimal Gambling Systems for Favorable Games".[42] Thorp goes into even greater detail in his later work "The Kelly Criterion in Blackjack, Sports Betting, and the Stock Market", where he graphically illustrates how the log of wealth is maximized to maximize the growth of wealth over time.[43] He specifically applies the criterion to the stock market in "The Kelly Criterion and the Stock Market".[44]

Nekrasov created a formula for implementing the Kelly criterion in multivariate portfolios, as seen in Eq. 2.[45] Consider a market with n correlated stocks S_k with stochastic return r_k and a riskless bond with return r. An investor puts a fraction u_k of his capital in S_k and the rest is invested in bonds. The following formula may be used to compute the optimum investments, where $\vec{r}$ and $\hat{\Sigma}$ are the vector of the means and the matrix of second mixed noncentral moments of the excess returns.[45]

$$\vec{u^*} = (1 + r)(\hat{\Sigma})^{-1}(\vec{(r)} - r). \tag{2}$$

4. Problem Definition

The scope of this proposal involves developing a packet capture tool that will intelligently select portions of packets in an effort to return more data in the sessions most likely to contain malicious data and less data in sessions most likely to be benign, while effectively using a limited amount of bandwidth. This entails developing an algorithm to maximize the efficient use of the available bandwidth and selecting and refining a suite of algorithms to determine the likelihood that a flow is malicious.

The following requirements must also be met within this scope:

- While exploring sensor-based packet loss, Smith et al.[1] discuss the ways in which packet loss was decreased by reducing the processing load of the CPU.[46–50] Since there is a direct relationship between the CPU load and the

number of packets dropped, it is important to reduce the load on the CPU as much as practical.

- Since the sensor will be using the same network that the monitored site uses for daily operation, the amount of data transmitted must be kept relatively small; 10% of the monitored traffic is considered a reasonable maximum.

- As sensors employing this technology may be deployed in Department of Defense networks, it is important that the software developed complies with the joint information environment[51–55] wherever applicable.

- The data that we collect will need to be analyzed by tools contributed by other organizations; therefore, it is important that the output format complies with relevant standards. In this field the packet capture (PCAP) data format used by Tcpdump[13] and implemented by Libpcap[56] is the de facto standard. The packet capture tool developed must support those standards as much as practical.

- Since the tool we develop will implement the e-box of a network intrusion detection system, it is important that it be resistant to insertion, evasion, and denial of service attacks as described by Ptacek and Newsham.[57]

- The Kelly criterion assumes that one is able to bet an arbitrary amount of one's total wealth; however, the amount of data that may be transmitted is limited by the content of the network traffic. It will be necessary to account for situations where the amount of traffic available for transmission is less than the amount that the algorithm indicates.

- The Kelly criterion assumes that one gets paid in the same currency with which one bets, implying that winning increases the wealth. This is not the case; rather, there is a steady income that does not increase with each win. It is quite possible that the algorithm may need to be significantly adjusted to account for this. Kelly himself stated that the formula would be significantly different if the gambler were to be on a fixed budget and unable to reinvest his winnings.[4]

- We will need to discover the optimal collection window. Comparing the problem of selecting the amount of traffic to transmit to the CAS to how much to bet on horses in a race or how much to invest in stocks in a portfolio, we

can see that the problem is basically given a set of choices and predicted outcomes regarding how much should we invest in each choice. In the horse racing problem, the choices are limited by the number of horses in the race. In the portfolio problem, the choices are limited by the number of stocks in the portfolio. In this case, the choices are limited by the size and duration of the window we use. If the window is small and short enough, then the choices will be limited to a single session. If the window is large and long enough, then the choices could expand significantly. The size and duration of the window also affects the amount of information that may be included in the maliciousness score. If the window is very short, then the only information that we may be able to access is the source and target IP address and ports. If the window is longer, we may be able to include a test of the entropy of the data to exclude encrypted traffic. If the window is long enough, we may be able to include the results of static NID systems.

5. Objective

The objective of this research is to build a network capture tool that captures malicous traffic. It must outperform the existing Vsnap network capture tool, which is the successor to the Snapper network capture tool described by Long.[3] The Vsnap tool uses a dynamic algorithm with limited intelligence for choosing which packet and how much of each packet to collect. The tool we develop will combine expert knowlege, data mining, and best of breed anomaly detection to create a maliciousness score for each session. This score will be fed into a Kelly criterion–inspired algorithm to decide how much of each session to transmit.

6. Research Questions

To complete our objectives, 3 primary research questions must addressed:

1. What is the optimal way to combine expert knowledge, data mining, and anomaly detection into a single maliciousness rating?
2. What is the optimal strategy for selecting how much traffic should be captured for a given session based upon its maliciousness rating?
3. What is the optimal window of selection?

(a) How long is the average network session?

(b) What is the tolerable latency between collection and transmission to the CAS?

(c) How computationally expensive may the algorithm be before it causes the system to lose packets?

(d) What level of packet loss is acceptable?

7. Approach

This research effort is broken down into 3 research questions and 3 phases. The first question, which will be addressed in phase 1, is how to determine what traffic is most likely to contain malicious activity. The second question, which will be addressed in phase 2, is how to select the traffic most likely to contain malicious activity for transmission to the analysis servers. The third research question and its subquestions, which will be addressed in phase 1, is to determine the optimal window of selection. In phase 3 the prototype developed in phase 2 will be incorporated into the Interrogator NIDS Architecture.

7.1 Phase 1

In phase 1, we plan to combine expert knowledge, data mining, and best of breed intrusion detection to compute a maliciousness rating. The first step of this research will be to discover the relevant facts that may be gleaned from expert knowledge. For example, when the Heart Bleed vulnerability was discovered, an expert could have caused the system to rate secure socket layer traffic higher; and when a known malicious IP address or domain is discovered, an expert could cause the system to rate traffic, including that IP or domain, higher. The second step of this research will be to discover the relevant facts that may be mined from the Interrogator data store. For example, Long and Morgan mined Interrogator to develop a white list of web servers to be excluded and instances of new servers to be included.[11] This could be expanded to rate traffic more malicious, which contains addresses and ports associated with alerts or incidents. The third step of this research will combine best of bread anomaly detection algorithms to form a maliciousness rating. For example, MINDS collected, computed, and assigned a local outlier factor to 16 different features.[31–33] KMC+NBC uses K-Means clustering and Naive Bayes Classification to detect anomalies in network traffic.[36] Again, a measure of abnormality could factor

into the session rating. The fourth step of this research will be to develop a formula to combine all of these into a single score. Phase 1 corresponds to the top half of Fig. 2, where unrated sessions are captured by the sensor and flow into the session rater, which uses expert knowledge, mined data, and anomaly algorithms to rate each session. The green sessions are known benign, the red sessions are known malicious, and the colors in between are meant to represent the continuum in between.

7.2 Phase 2

In phase 2, we plan to develop a Kelly criterion–based formula that takes the scores generated from phase 1 as input and produces as output a fraction of the available network traffic that should be invested in each session. Kelly proved that there exists an amount to bet l being some portion of the total wealth G, that if the gambler bets it consistently, G will obtain and maintain a level greater than any other possible value for l.[4] This may be seen in Eq. 1, where l is the fraction of wealth to bet, p is the probability of winning, and b is the net odds of the wager. Thorp applied the Kelly criterion to the game of Blackjack.[39] Smoczynski and Tomkins applied the Kelly criterion to horse racing.[58] Separately, Thorp and Nekrasov applied the Kelly criterion to the stock market.[41,45] Using this generalization, one would consider network flows to be stocks and rate of return to be the maliciousness score of the session. Phase 2 corresponds to the bottom half of Fig. 2, where the rated sessions flow into the algorithm and the session selector feeds those ratings into the Kelly criterion[4]–inspired formula to determine how much traffic to invest in each session. The fatter sessions represent more traffic being invested in the session, and the skinnier sessions represent less traffic being invested in the session.

We will use Nekrasov's formula in Eq. 2 to illustrate how this might work. To apply this to our problem, we will substitute the returns for the maliciousness score and the investment for the amount of available traffic to assign to each session. Since a riskless bond makes no sense in our problem, we will set the value to zero, simplifying the equation as in Eq. 3. This leaves us with only one variable because the second noncentral moment is a function of the maliciousness rating over time. Remember, it is unlikely that Nekrasov's formula will work as given. We need to start from the same starting point that Kelly did to retrace his steps to construct a formula for this specific application.

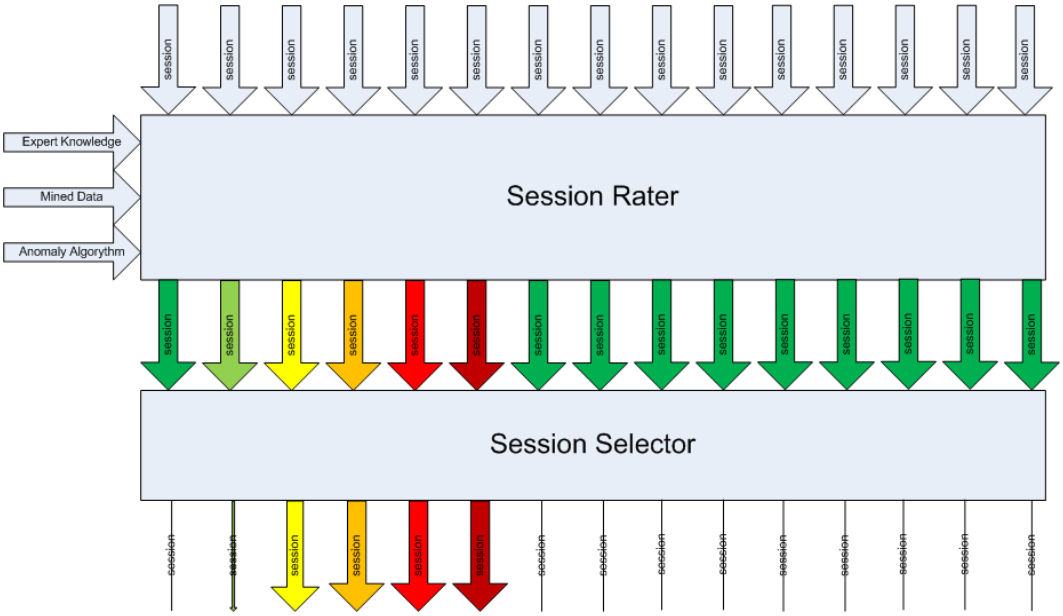

Fig. 2 Kelly compressor diagram

$$\vec{u^*} = (\hat{\Sigma})^{-1}(\vec{(r)}). \tag{3}$$

Once the session rater and session selector algorithms are developed, they will be incorporated into a prototype that will be tested against open source datasets to include those used by Smith et al. in their theoretical exploration[1] and data collected by the Army Research Laboratory (ARL) Computer Network Defense Service Provider (CNDSP).

7.3 Phase 3

In phase 3, the prototype developed in phase 2 will be developed into a production application that addresses all of the requirements from Section 4 and may be incorporated into the Interrogator NIDS Architecture.[3]

8. Milestones

Milestones for phase 1 include the following:

- Produce a validated list of facts that may be discovered through expert knowledge.

- Select and evaluate several techniques for mining the Interrogator data store.
- Produce a validated list of facts that may be discerned by mining the Interrogator data store.
- Select and evaluate several anomaly detection algorithms against open source datasets and data collected from the ARL CNDSP.
- Produce a validated list of facts that may be discerned by executing these anomaly detection algorithms against network traffic.
- Combine all of these into a maliciousness rating.

Milestones for phase 2 include the following:

- Develop a Kelly criterion–inspired formula that takes as input the facts discovered in phase 1.
- Develop a prototype network compression tool that uses our collection of facts and our modified Kelly criterion to produce compressed network traffic.
- Assess this prototype based upon its ability to compress the traffic and the amount of malicious activity in the original data but not in the compressed data.

Milestones for phase 3 include the following:

- Incorporate prototype from phase 2 into the Interrogator Network Intrusion Detection Architecture.[3]
- Evaluate its performance in a relevant environment.

9. Expected Results

We have access to several public datasets of network traffic. We also have access to the Vsnap network capture tool, which is the successor of the Snapper network capture tool describe by Long.[3] We will be able to process our datasets with both Vsnap and the Kelly compressor and compare which captures more malicious traffic. We will also be able to measure the amount of benign traffic each tool captures and compare their relative densities.

10. Conclusion

In a distributed NIDS environment, it is necessary to transmit the right data back to the central analysis servers to provide analysts with the information necessary to detect and report malicious activity. Bringing back all of the data would double the bandwidth requirements of the site and require that the analysis servers have massive bandwidth available to receive it all. Standard lossless compression is not sufficient to reduce this traffic to an acceptable level. The goal of this research is to develop a lossy compression algorithm that will ensure that the traffic lost is the least likely to contain malicious activity. The approach is to use an algorithm based upon the Kelly criterion to allocate the limited bandwidth available, coupled with best of breed anomaly detection, to assess the maliciousness of the traffic. These 2 technologies will be combined into a packet capture tool that will produce data compliant with the standards used by existing NIDS tools.

11. References

1. Smith S, Hammell R, Parker T, Marvel L. A theoretical exploration of the impact of packet loss on network intrusion detection. International Journal of Networked and Distributed Computing. 2016;4(1):1–10.

2. Smith SC, Hammell RJ. An experimental exploration of the impact of sensor-level packet loss on network intrusion detection. Aberdeen Proving Ground (MD): Army Research Laboratory (US); 2015 Jul. Report No.: ARL-TR-7353.

3. Long KS. Catching the cyber spy: ARL's interrogator. Aberdeen Proving Ground (MD): Army Research Laboratory; 2004. DTIC Document No.: ADA432198.

4. Kelly JL. A new interpretation of information rate. Information Theory, IRE Transactions on. 1956;2(3):185–189.

5. Roesch M. Snort: lightweight intrusion detection for networks. In: Proceedings of the 13th System Administration Conference (LISA '99); Vol. 99; 1999 Nov 7–12; Seattle, WA. p. 229–238.

6. Paxson V. Bro: a system for detecting network intruders in real-time. Computer Networks. 1999;31(23):2435–2463.

7. Kemmerer RA, Vigna G. Intrusion detection: a brief history and overview (supplement to Computer magazine). Computer. 2002;35(4):27–30.

8. Ziv J, Lempel A. A universal algorithm for sequential data compression. IEEE Transactions on Information Theory. 1977;23(3):337–343.

9. Sangster B, O'Connor T, Cook T, Fanelli R, Dean E, Adams WJ, Morrell C, Conti G. Toward instrumenting network warfare competitions to generate labeled datasets. In: Proc. of the 2nd Workshop on Cyber Security Experimentation and Test (CSET09); 2009 Aug; Montreal, Canada.

10. Ierace N, Urrutia C, Bassett R. Intrusion prevention systems. Ubiquity. 2005;2005(June):2–2.

11. Long KS, Morgan JB. Using data mining to improve the efficiency of intrusion detection analsysis. Aberdeen Proving Ground (MD): Army Research Laboratory (US); 2007. Report No.: ARL-TR-4211.

12. Cohen WW. Fast effective rule induction. In: Proceedings of the twelfth international conference on machine learning; 1995 Jul 9–12; Lake Tahoe, CA. Morgan Kaufmann; 1995. p. 115–123.

13. Jacobson V, Leres C, McCanne S. tcpdump – dump traffic on a network. 2015 Sep 17 [accessed 2016 Feb 3]. http://www.tcpdump.org/manpages/tcpdump.1.html.

14. Lee W, Stolfo SJ. Data mining approaches for intrusion detection. In: Proceedings of the 7th USENIX Security Symposium; 1998 Jan; San Antonio, TX.

15. Grinstein G, Laskowski S, Wills G, Rogowitz B. Information exploration shootout project and benchmark data sets (panel): evaluating how visualization does in analyzing real-world data analysis problems. In: Proceedings of the 8th Conference on Visualization '97; 1997 Oct 18-24; Phoenix, AZ. (VIS '97) Los Alamitos (CA): IEEE Computer Society Press; 1997. p. 511–513.

16. Claise B. Cisco systems netflow services export version 9. Fremont (CA): Internet Engineering Task Force (IETF); 2004 Oct. Report No.: RFC 3954.

17. Luo J. Integrating fuzzy logic with data mining methods for intrusion detection [thesis]. [Starkville (MS)]: Mississippi State University; 1999.

18. Bridges SM, Vaughn RB. Fuzzy data mining and genetic algorithms applied to intrusion detection. In: Proceedings of the 12th Annual Canadian Information Technology Security Symposium; 2000 Jun 19–23; Ottawa, Canada. p. 109–122.

19. Dokas P, Ertoz L, Kumar V, Lazarevic A, Srivastava J, Tan PN. Data mining for network intrusion detection. In: Proc. NSF Workshop on Next Generation Data Mining; 2002 Nov 1–2; Baltimore, MD. p. 21–30.

20. Garcia-Teodoro P, Diaz-Verdejo J, Maciá-Fernández G, Vázquez E. Anomaly-based network intrusion detection: techniques, systems and challenges. Computers & Security. 2009;28(1):18–28.

21. Mukherjee B, Heberlein LT, Levitt KN. Network intrusion detection. Network, IEEE. 1994;8(3):26–41.

22. Heberlein LT, Dias GV, Levitt KN, Mukherjee B, Wood J, Wolber D. A network security monitor. In: Proceedings. 1990 IEEE Computer Society Symposium on Research in Security and Privacy; 1990 May 7-9; Oakland, CA. p. 296–304.

23. Heberlein LT, Mukherjee B, Levitt K, Dias G, Mansur D. Towards detecting intrusions in a networked environment [master's thesis]. [Davis (CA)]: U. of Calif., Davis; 1991.

24. Heberlein L, Levitt K, Mukherjee B. A method to detect intrusive activity in a networked environment. In: Proceedings of the 14th National Computer Security Conference; 1991 Oct 1–4; Washington, DC. p. 362–371.

25. Sekar R, Uppuluri P. Synthesizing fast intrusion prevention/detection systems from high-level specifications. In: Proceedings of the 8th USENIX Security Symposium; Vol. 99; 1999 Aug 23–26; Washington, DC. ACM; 1999.

26. Uppuluri P, Sekar R. Experiences with specification-based intrusion detection. In: Recent Advances in Intrusion Detection; 2001 Oct 10–12; Davis, CA. p. 172–189.

27. Sekar R, Gupta A, Frullo J, Shanbhag T, Tiwari A, Yang H, Zhou S. Specification-based anomaly detection: a new approach for detecting network intrusions. In: CCS '02: Proceedings of the 9th ACM Conference on Computer and Communications Security; 2002 Nov 18-22; Washington, DC. New York (NY): ACM; 2002. p. 265–274.

28. Lippmann RP, Fried DJ, Graf I, Haines JW, Kendall KR, McClung D, Weber D, Webster SE, Wyschogrod D, Cunningham RK, Zissman MA. Evaluating intrusion detection systems: the 1998 DARPA off-line intrusion detection evaluation. In: DARPA Information Survivability Conference and Exposition, 2000. DISCEX'00. Proceedings; Vol. 2; 2000 ; Hilton Head, SC. p. 12–26.

29. Eskin E, Arnold A, Prerau M, Portnoy L, Stolfo S. A geometric framework for unsupervised anomaly detection. In: Barbarà S Daniel; Jajodia, editor. Applications of data mining in computer security; New York (NY): Springer; 2002. p. 77–101.

30. Krügel C, Toth T, Kirda E. Service specific anomaly detection for network intrusion detection. In: Proceedings of the 2002 ACM symposium on Applied computing; 2002 Mar 10–14; Madrid, Spain. p. 201–208.

31. Ertoz L, Eilertson E, Lazarevic A, Tan P, Dokas P, Srivastava J, Kumar V. Detection and summarization of novel network attacks using data mining. Minneapolis (MN): Army High Performance Computing Research Center; 2003 May. Report No.: 2003-108.

32. Ertoz L, Eilertson E, Lazarevic A, Tan PN, Kumar V, Srivastava J, Dokas P. Minds-Minnesota intrusion detection system. In: Kargupta H, Joshi A, Sivakumar K, Yesha Y, editors. Next generation data mining. Cambridge (MA): MIT Press; 2004. p. 199–218.

33. Chandola V, Eilertson E, Ertoz L, Simon G, Kumar V. Minds: architecture & design. In: Singhal A, editor. Data warehousing and data mining techniques for cyber security. New York (NY): Springer; 2007. p. 83–108.

34. Münz G, Li S, Carle G. Traffic anomaly detection using k-means clustering. In: GI/ITG Workshop MMBnet; 2007 Sep 13–14; Hamburg, Germany. p. 1–8.

35. Shiravi A, Shiravi H, Tavallaee M, Ghorbani AA. Toward developing a systematic approach to generate benchmark datasets for intrusion detection. Computers & Security. 2012;31(3):357–374.

36. Yassin W, Udzir NI, Muda Z, Sulaiman MN. Anomaly-based intrusion detection through k-means clustering and naives Bayes classification. In: Proceedings of the 4th International Conference on Computing and Informatics (ICOCI); 2013 Jul 7; Sarawak, Malaysia. p. 298–303.

37. Breiman L. Optimal gambling systems for favorable games. In: MacLean LC, Thorp EO, Ziemba WT, editors. The Kelly captial growth investment criterion: theory and practice. New Jersey (NJ): World Scientific; 2012. p. 47–60.

38. Thorp EO. Fortune's formula: the game of blackjack. Notices of the American Mathematical Society. 1960;7(7):935–936.

39. Thorp EO. Beat the dealer, a winning strategy for the game of twenty-one. New York (NY): Random House; 1966.

40. Thorp EO. Understanding the kelly criterion. In: MacLean LC, Thorp EO, Ziemba WT, editors. The Kelly capital growth investment criterion: theory and practice; New Jersey (NJ): World Scientific; 2012. p. 511–525.

41. Thorp EO, Kassouf ST. Beat the market: a scientific stock market system. New York (NY): Random House; 1967.

42. Thorp EO. Optimal gambling systems for favorable games. Revue de l'Institut International de Statistique. 1969;37(3):273–293.

43. Thorp EO. The Kelly criterion in blackjack, sports betting, and the stock market. Finding the Edge: Mathematical Analysis of Casino Games. 1998;1(6).

44. Rotando LM, Thorp EO. The Kelly criterion and the stock market. American Mathematical Monthly. 1992;99(10):922–931.

45. Nekrasov V. Kelly criterion for multivariate portfolios: a model-free approach. Social Science Research Network; 2014 Sep 30 [accessed 15 Aug 2015]. http://dx.doi.org/10.2139/ssrn.2259133.

46. Salah K, Kahtani A. Improving SNORT performance under Linux. IET communications. 2009;3(12):1883–1895.

47. Schaelicke L, Freeland JC. Characterizing sources and remedies for packet loss in network intrusion detection systems. In: Workload Characterization Symposium, 2005. Proceedings of the IEEE International; 2005 6-8 Oct; Austin, TX. IEEE Conference Publications; 2005. p. 188–196.

48. Kim NU, Park MW, Park SH, Jung SM, Eom JH, Chung TM. A study on effective hash-based load balancing scheme for parallel nids. In: Advanced Communication Technology (ICACT), 2011 13th International Conference on; 2011 13-16 Feb; Gangwon-Do, Korea (South). p. 886–890.

49. Song B, Yang W, Chen M, Zhao X, Fan J. Achieving flow-level controllability in network intrusion detection system. In: SNPD '10 Proceedings of the 2010 11th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing; 2010 9-11 Jun; Washington DC. IEEE Computer Society; 2010. p. 55–60.

50. Wei C, Fang Z, Li W, Liu X, Yang H. The IDS model adapt to load characteristic under IPv6/4 environment. In: Wireless Communications, Networking and Mobile Computing, 2008. WiCOM'08. 4th International Conference; 2008 19-21 Apr; Dalian, China. p. 1–4.

51. Charter for the joint information environment management construct. DoD, Chief Information Officer; 2012 Nov 9 [accessed 2015 Mar]. http://dodcio.defense.gov/Portals/0/Documents/JIE/DoD_CIO_JIE_Management_Construct_Charter_(9Nov2012).pdf.

52. Joint information environment white paper. Joint Chief of Staff; 2013 Jan [accessed 2015 Mar]. http://www.jcs.mil/Portals/36/Documents/Publications/environmentalwhitepaper.pdf.

53. The department of defense strategy for implementing the joint information environment. Department of Defense; 2013 Sep [accessed 2015 Mar]. http://dodcio.defense.gov/Portals/0/Documents/JIE/2013-09-13z_DoD_Strategy_for_Implmenting_JIE_(NDAA_931)_Final_Document.pdf.

54. Guidance for implementing the joint information environment. DoD, Chief Information Officer; 2013 Sep [accessed 2015 Mar]. http://dodcio.defense.gov/Portals/0/Documents/JIE/20130926_JointInformationEnvironmentImplementationGuidance_DoDCIO_Final_Document.pdf.

55. Enabling the joint information environment (JIE). Defense Information Systems Agency; 2014 May [accessed 2015 Mar]. http://www.disa.mil/~/media/Files/DISA/About/JIE101_000.pdf.

56. Jacobson V, Leres C, McCanne S. PCAP – packet capture library. 2015 Mar 8 [accessed 2016 Feb 3]. http://www.tcpdump.org/manpages/pcap.3pcap.1.html.

57. Ptacek TH, Newsham TN. Insertion, evasion, and denial of service: Eluding network intrusion detection. Falls Church (VA): Information Assurance Technology Analysis Center; 1998 Jan. DTIC Document No.: ADA391565.

58. Smoczynski P, Tomkins D. An explicit solution to the problem of optimizing the allocations of a bettor's wealth when wagering on horse races. Mathematical Scientist. 2010;35(1):10-17.

List of Symbols, Abbreviations, and Acronyms

ACRONYMS:

ARL	Army Research Laboratory
CAS	central analysis servers
CNDSP	computer network defense service provider
CPU	central processing unit
IP	internet protocol
MINDS	Minnesota INtrusion Detection System
NIDS	network intrusion detection system
NSM	Network Security Monitor
PCAP	packet capture

MATHEMATICAL SYMBOLS:

l	the amount to bet from the Kelly criterion
G	total wealth from the Kelly criterion
p	the probability of winning
b	the net odds of the wager
S_k	the stochastic return
r_k	return of a riskless bond with return r
u_k	a fraction u_k of capital
$\vec{r}$	the vector of the means T
$\hat{\Sigma}$	the matrix of second mixed noncentral moments of the excess returns

1 (PDF)	DEFENSE TECHNICAL INFORMATION CTR DTIC OCA
2 (PDF)	DIRECTOR US ARMY RESEARCH LAB RDRL CIO LL IMAL HRA MAIL & RECORDS MGMT
1 (PDF)	GOVT PRINTG OFC A MALHOTRA
1 (PDF)	DIR USARL RDRL CIN S S SMITH

INTENTIONALLY LEFT BLANK.