STATISTICAL SIGNATURES FOR EARLY DETECTION OF FLOODING DENIAL-OF-SERVICE ATTACKS

John Haggerty[1], Qi Shi[1] and Madjid Merabti[1]
[1]*Liverpool John Moores University, School of Computing & Mathematical Sciences, Byrom Street, Liverpool, L3 3AF. E-mail: {J.Haggerty, Q.Shi, M.Merabti}@livjm.ac.uk*

Abstract: A major threat to the information economy is denial-of-service attacks. Despite the widespread deployment of perimeter model countermeasures these attacks are highly prevalent. Therefore a new approach is posited; early detection. This paper posits an approach that utilises statistical signatures at the router to provide early detection of flooding denial-of-service attacks. The advantages of the approach presented in this paper are threefold: analysing fewer packets reduces computational load on the defence mechanism; no state information is required about the systems under protection; and alerts may span many attack packets. Thus, the defence mechanism may be placed within the routing infrastructure to prevent malicious packets from reaching their intended victim in the first place. This paper presents an overview of the early detection-enabled router algorithm and case study results.

Keywords: network attacks; denial of service; statistical signatures; early detection.

1. INTRODUCTION

The flow of information is the most valuable commodity for organisations and users alike. Information is traded within the networked world and we are becoming ever more reliant on access to data and resources as technologies develop to facilitate this flow. Our reliance on such network technologies has ensured that financially unquantifiable assets, such as

people, reputation, and business relations, are amongst the most important to business [1].

A major threat posed to this information economy paradigm is that of denial-of-service attacks. These attacks present a very real threat as they disrupt or interrupt the flow of data that organisations rely on. Such attacks can be launched in a number of ways, from malicious use of common applications such as e-mail, to subverting Internet protocols. The subversion of Internet protocols leads to flooding attacks, whereby large volumes of data are sent to the victim. Denial-of-service attacks may also be a side-effect of other types of attack, such as Internet worms. Irrespective of the *modus operandi*, denial-of-service attacks are prevalent because the tools required are freely available on the Internet, simple to launch, effective, and difficult to prevent. Thus, large numbers of attacks are continuously being launched [2]. In addition, businesses that rely on their connectivity, such as on-line services, can be blackmailed with the threat of a denial-of-service attack if they were not to pay [3].

Yet, despite the prevalence of these attacks, a cost-effective and efficient countermeasure has yet to be proposed. Current defences rely on the perimeter model of network security, where a boundary is established around the nodes under protection. Inside the perimeter is trusted space, whilst outside is viewed as untrustworthy. Denial-of-service attacks remain a significant problem due to the unsuitability of perimeter devices for two reasons. First, perimeter devices are located on the victim system and are therefore under attack at the point of detection. Second, these devices inspect each and every packet in an attack, which in the case of denial of service places a large computational load on the defence mechanism in addition to the large network load caused by the attack.

This paper demonstrates that the use of statistical signatures for early detection of denial-of-service attacks can greatly reduce the volume of packets that are inspected to determine malicious packets from legitimate. The approach employed by this paper has three novel contributions. First, the computational load is reduced on the defence mechanism as fewer packets are analysed. Second, no state information about the networks under protection needs to be held, again reducing computational load. Third, reports of attacks may relate to several packets rather than 'one packet, one alert' techniques employed by traditional countermeasures utilising non-statistical signatures. The reduction of volume enables detection devices to be placed beyond the perimeter and within the routing infrastructure thus enabling attacks to be thwarted prior to their reaching their intended target. As demonstrated in section 4, the approach posited in this paper remains highly efficient despite a significantly reduced number of packets inspected.

This paper is organised as follows. Section 2 discusses related work. Section 3 presents an overview of statistical signatures and the effects that a

denial-of-service attack has on a network that is used for early detection. Section 4 presents a case study and results. Finally, section 5 presents conclusions and further work.

2. RELATED WORK

Flooding denial-of-service attacks are distinct from other attacks, for example, those that execute malicious code on their victim, in that they require a large volume of traffic, and it is this continuing stream of data that prevents the victim from providing services to legitimate users. It is the mass of all packets directed at the victim that poses the threat, rather than the contents of the packets themselves.

Flooding denial-of-service attacks are problematic due to their subversion of normal network protocols. As such, it is these attacks that pose the greatest problem in today's network infrastructures. Subverting the use of protocols, such as the Transmission Control Protocol (TCP) or User Datagram Protocol (UDP), enables the attacker to disrupt on line services by generating a traffic overload to block links or cause routers near the victim to crash [4]. Because they subvert existing protocols the packets involved in these attacks are high-volume without being conspicuous or easily traceable. For example, TCP SYN flooding specifically targets weaknesses in the TCP protocol to achieve its aim. This attack method, which accounts for 94 per cent of denial-of-service attacks [2], is based on exploiting the three-way handshake in TCP.

A number of approaches have been posited to counter the denial-of-service problem. For example, [5] proposes stronger authentication between communicating parties across a network. Alternatively, [6] suggests that network resources should be divided into classes of service, where higher prices would attract less traffic and ensure that an attacker could not afford to launch an attack. Alternatively, [7] suggests that the routing infrastructure should be more robust by securing servers in the first place. These approaches are not without their problems. For example, authentication, whilst attempting to prevent denial of service, paradoxically leaves itself open to such an attack due to the computational load required for the defence. Payment approaches assume that the consumer is willing to pay for different levels of service, which they are usually not. Finally, it is often poor software development practices due to the pressure of getting a product to market that lead to the release of server applications that are subvertable.

These problems have led to the rise of traffic monitoring approaches and fall into two categories; *statistical monitoring* and *adaptation of congestion algorithms*. Statistical monitoring of networks, such as [8, 9], observes a network and detect upsurges in traffic of a particular type or for system

compromise. An advantage of this approach is that one alert may cover a number of attack packets, thus reducing network load caused by the reporting of events. In addition, a large upsurge of traffic is indicative of a flooding attack, irrespective of the protocol used by the attacker. Alternatively, congestion algorithms are adapted for detection of denial-of-service attacks. Approaches such as [10, 11], use existing congestion techniques, where routers deal with upsurges in traffic to ensure quality of service, to detect denial of service. These approaches have the benefit of being able to detect the attack in the routing infrastructure, thus being able to halt the attack before it reaches its intended victim.

However, even these more sophisticated approaches are not without their problems. Statistical approaches require human intervention to monitor the networks for upsurges, so is both labour intensive and inefficient. The congestion adaptation approaches may only apply simplistic signatures so as to not impede on the throughput of traffic. In addition, approaches such as [10, 11] require that state information is held on the router. This information is computationally too exhaustive to be effective within the routing infrastructure.

Therefore, a new approach is required that provides early detection of denial-of-service attacks; one that can combine and make use of the advantages of both the statistical and adaptation of congestion algorithm approaches. In this way, the benefits as above are achieved.

3. STATISTICAL SIGNATURES

Traditional stateful signature analysis applies statistical methods to collected data within the system over a period of time. This data is then analysed to generate some system-specific values: for example, traffic thresholds or user profiles to define normal or abnormal behaviour [12]. By allowing a system to keep state information of the system, detection signatures can be designed to match a complex series of events which fall outside that normal behaviour. A number of techniques are employed in this area and include:

- *Collection of events.* In any system, a number of events may be observed in conjunction to indicate that an attack is under way.
- *Threshold enforcement.* A certain threshold of acceptable events is determined for the system based on prior experience or threats. Once events in the system surpass this threshold, an alert is generated to indicate that an attack is under way.
- *Frequency threshold.* This is a variation on threshold enforcement and is widely used in authentication. If one or more events are

observed, then an alert is raised or services halted until a time limit is reached.

Other approaches that fall into this category include analysis of mean and standard deviation information, the multivariate model, Markov process model, and clustering analysis [12].

These approaches are widely used in anomaly intrusion detection where misuse against known but ill-defined variables is being matched. Despite the requirement for state information to be held by these approaches, statistical monitoring is effective in detecting large volumes of traffic being directed at a victim host. The way in which this is achieved statelessly is presented in section 4.

These approaches require state information to be held about the systems under protection but this is too computationally exhaustive to be used in the routing infrastructure.

The effects on network dynamics of a denial-of-service attack and the applicability of a statistical-based approach can be clearly demonstrated through the use of a probability plot. A probability plot assesses whether a particular distribution fits the given data. The plot points are calculated using a non-parametric method. The fitted line provides a graphical representation of the percentiles. The fitted line is created by calculating the percentiles for the various percents based on the chosen distribution. The associated probabilities are transformed and used as the y variables. The percentiles may be transformed and then used as the x variables. A goodness of fit measure, such as the Anderson-Darling statistic [13], is then applied to the data. This is a measure of how far the plot points fall from the fitted line in the probability plot. The statistic is a weighted squared distance from the plot points to the fitted line with larger weights in the tails of the distribution. A smaller Anderson-Darling "Goodness of Fit" indicates that the distribution fits the data better.

To demonstrate the effect of a denial-of-service attack on the data, attacks are calculated and plotted according to this technique. Figures 1 to 4 compare the results for control traffic, *nuke* which utilises only a small number of packets during an attack, a SYN flood attack, and a UDP flood.

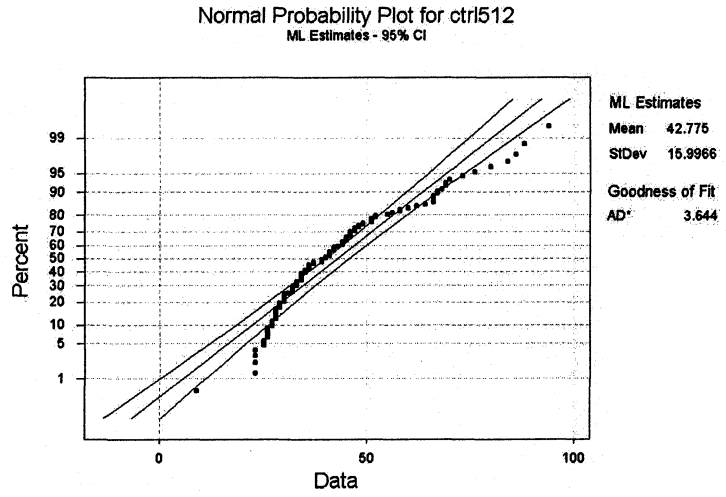

Figure 1. Probability plot for control traffic.

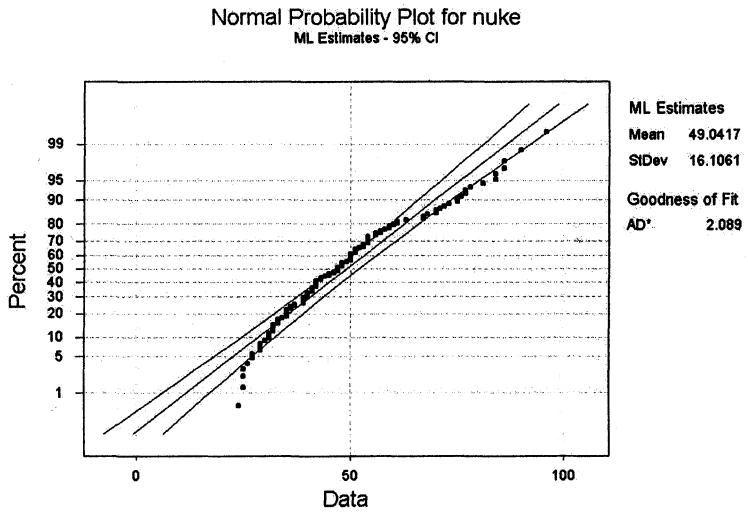

Figure 2. Probability plot for nuke attack traffic.

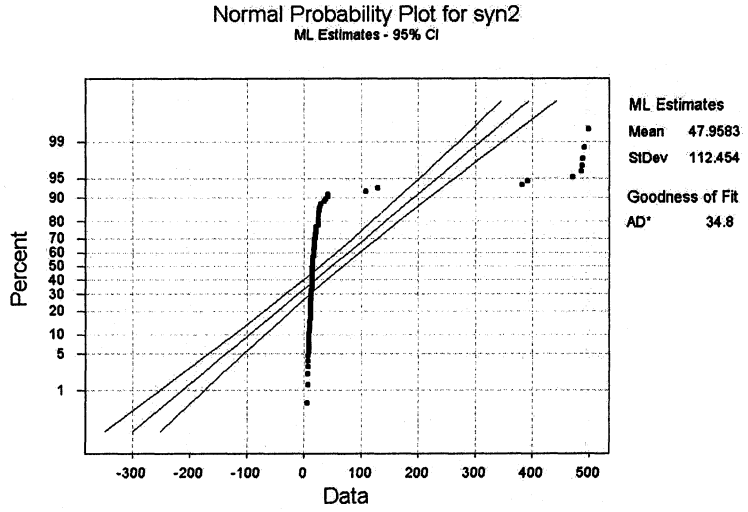

Figure 3. Probability plot for TCP SYN flood attack traffic.

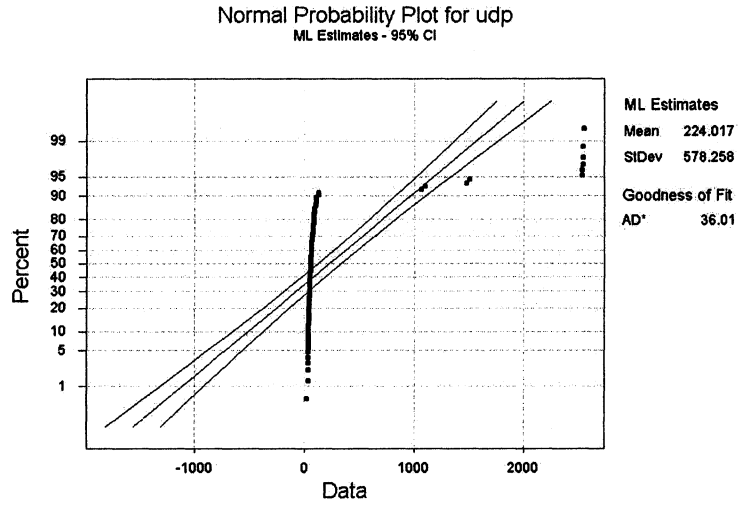

Figure 4. Probability plot for UDP flood attack traffic.

The Anderson-Darling Goodness-of-Fit statistics provides the maximum likelihood and least squares estimation. The control traffic in figure 1 has a low Anderson-Darling Goodness of Fit of 3.64, suggesting that the distribution fits the data well, i.e. an attack is not underway. In

vulnerability attacks, such as nuke in figure 2, again the distribution fits the data well with a Goodness of Fit of 2.09. This is due to the few additional packets placed on the network for this attack to achieve its objective. In total, only nine packets are sent to the victim to exploit the kernel vulnerability on the target machine. A further six packets are sent to ensure that the attack has caused the desired effect. Therefore, during an attack, only 15 malicious packets are required. This accounts for the low Goodness of Fit.

Figures 3 and 4 demonstrate that high-volume attacks have a severe effect on the network. In figure 3, the TCP SYN flood attack generates a much higher than control mean and standard deviation, 47.95 and 112.45 respectively. The Anderson-Darling Goodness of Fit is therefore high, 34.8, suggesting that the distribution fits the data significantly poorer than control. This pattern is repeated in figure 4 with the UDP flood attack on the network. The mean and standard deviation remain high, 224.02 and 578.26 respectively. The Anderson-Darling Goodness of Fit is also high at 36. As we can see, the Anderson-Darling Goodness of Fit statistics are ineffective for the detection of attacks that utilise only a small number of packets, such as nuke, but are effective for flooding denial-of-service attacks such TCP SYN flood or UDP flood.

4. CASE STUDY AND RESULTS

The challenge remains as to how to implement statistical signatures within the routing infrastructure in an environment that is unable to support state information. This is achieved by enhancing the existing congestion algorithms present on routers. Congestion occurs within networks, and so routers employ congestion algorithms, such as RED [14] or CHOKe [15], to ensure that they do not fail when faced with high levels of traffic. These algorithms may be as simple as employing a *first in, first out* (FIFO) queue. Once the queue maximum limit is reached, packets are dropped in accordance with the congestion algorithm to ensure queue space for further incoming packets. In this way, an acceptable level of traffic throughput is maintained.

This paper presents the Distributed Denial-of-Service Defence Mechanism (DiDDeM) architecture [16] for early detection of denial-of-service attacks. DiDDeM is a domain-based system that adapts congestion algorithms within the routing infrastructure. The DiDDeM system comprises a server liaising with a number of DiDDeM-enhanced routers that pre-filter attack traffic outside the traditional perimeter.

Congestion algorithms are adapted for statistical signature matching by detecting large traffic volumes associated with a flooding denial-of-

service attack. Rather than purely dropping packets when the router threshold is met, packets to be dropped from the queue are inspected. This enables inference of stateful information about traffic flows and whether these unusual flows are intended for a particular destination thereby suggesting an attack. It is the random inspection that allows the state inference. If two (or more) sampled dropped packets are heading to one destination, they are checked against other (stateless) signatures to confirm an attack.

To demonstrate the way in which this is achieved in the DiDDeM architecture, a *first-in, first-out* (FIFO) queue is used within a *ns2* prototype. The available space within the DiDDeM-enhanced FIFO queue is divided into two sub-queues to allow comparison of packets. If due to bandwidth restrictions the packet cannot be immediately forwarded to the next router, an incoming packet to the router is placed in a queue. These packets are placed in either the first or second sub-queue at the router based on a first-come, first-served basis.

Packets placed in the queue, and its sub-queues, are dequeued and forwarded to their destination. If the threshold of the total queue limit is exceeded the router begins to drop packets to ensure that packets already in the queue are forwarded and that new incoming packets can be placed in the queue. In this way, no stateful information is held about the queue apart from whether the queue limit has been exceeded, thereby reducing the computational overhead placed on the router.

At periods where congestion occurs, packets are dropped. By meeting the threshold of the particular router which invokes packet dropping, an upsurge in traffic can be inferred. However, this may or may not be due to large amounts of traffic, such as would occur during a flooding denial-of-service attack. Therefore, prior to packets being dropped, the IP header is accessed and the destination address obtained. This IP destination address is compared to the previous packet's IP destination address. If they are the same, then the IP destination address is stored for comparison with the next packet and the packet is passed for stateless signature analysis. Stateless signature analysis verifies attacks by applying techniques used in misuse detection whereby fixed byte sequences within the packet are inspected. If the destination addresses are not the same, the destination IP address is still stored for comparison with the next packet, but the packet is dropped. The algorithm is illustrated in figure 5.

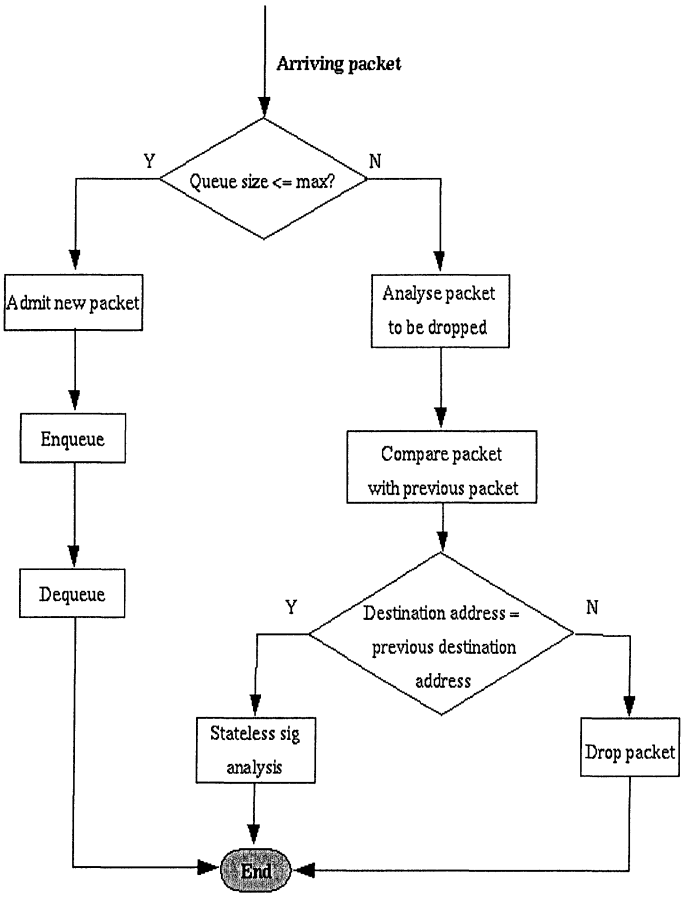

Figure 5. Routing algorithm using a FIFO queue.

A simulation was conducted in which approximately 19,500 attack packets were directed at the victim node by two attacking nodes. This represented an attack consisting of approximately 1,000 packets per second. Once the congestion algorithm was invoked by the router, 798 attack and legitimate packets were to be dropped. Of this number, 742 packets were actual attack packets whilst the remainder were legitimate traffic. Therefore, out of a total of 19,500 attack packets, only 4 per cent of this volume was inspected. DiDDeM detected 697 packets using the algorithm in figure 5. The 697 packets detected out of the 742 inspected by the DiDDeM-enhanced router ensure a 94 per cent detection rate. This system is therefore extremely efficient.

The prototype is tested on two systems to measure the impact of DiDDeM detection on the router. The methodology used is to measure the impact of the simulation on the processor by using the *top* program. This program measures the load by applications on the processor in the UNIX/Linux operating system. To provide a comparison with current network standards, the memory and processor usage were tested for DiDDeM and two routing algorithms; DropTail, and RED. This comparison allows us to see the impact on router efficiency in implementing DiDDeM and is presented in table 1.

Table 1. Impact on memory and processor of DiDDeM versus RED and DropTail algorithms.

Algorithm	PII 400 MHz 128 Mb RAM		AMD 2 GHz 256 Mb RAM	
	Memory usage	Processor load	Memory usage	Processor load
DiDDeM	*4.70%*	*95.50%*	*2.30%*	*41.10%*
RED	*4.70%*	*96.70%*	*2.30%*	*60.00%*
DropTail	*4.60%*	*96.20%*	*2.20%*	*51.80%*

As demonstrated by table 1, the impact of the simulation routing algorithm affected the memory usage of both computers. The DropTail routing algorithm required less memory usage, an improvement of 2.13% (PII processor) and 4.34% (AMD Athlon) compared to both DiDDeM and RED. However, DiDDeM was actively detecting denial-of-service attacks whilst the RED algorithm merely detected congestion at the router. In terms of processor load, DiDDeM proved to be more efficient.

One key measure for DiDDeM is its performance within the network environment. In particular, the DiDDeM algorithm should not have an adverse affect on the network, which would require a trade off between usability and effectiveness. In order to measure the performance of the DiDDeM in the network it is compared to the routing algorithms above; RED and DropTail. Unlike RED, DiDDeM and DropTail do not require any information about the state of the queue. However, to test the impact of DiDDeM on the queue and the network, the number of datagrams and packets passed from a router within the attack domain to the second router is measured. This impact is illustrated in figures 6 to 8 showing DiDDeM, DropTail and RED.

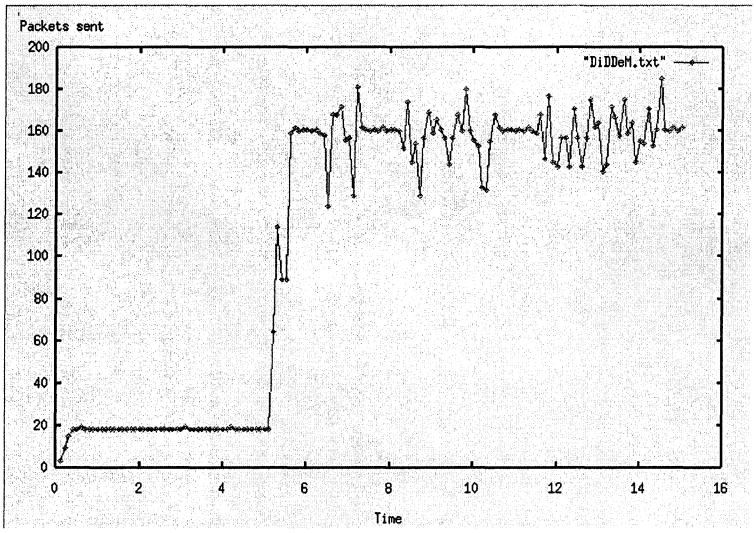

Figure 6. Packets sent by the DiDDeM-enabled router.

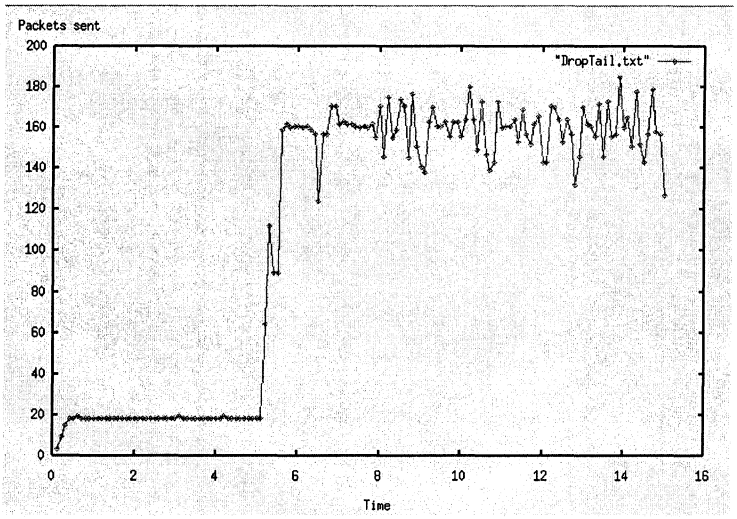

Figure 7. Packets sent by the DropTail-enabled router.

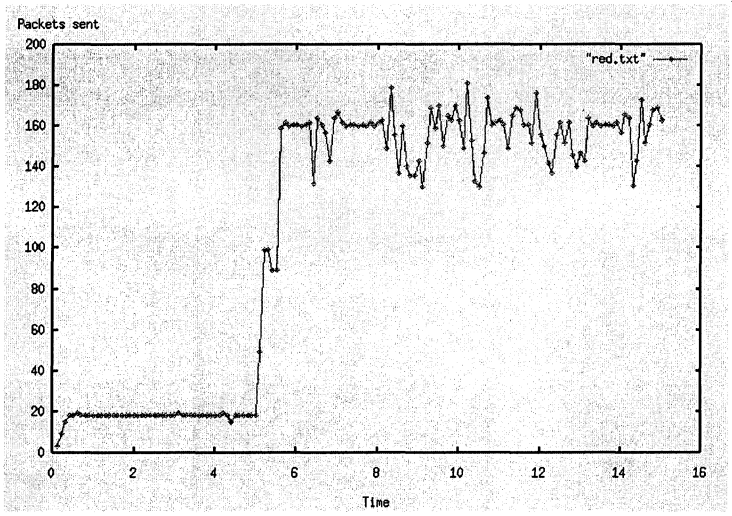

Figure 8. Packets sent by the RED-enabled router.

Figures 6 to 8 show the number of packets sent by router 1 at 0.1 second intervals. The attack is launched just before 5 seconds, as indicated by the sharp increase of traffic. Prior to the attack, all three approaches steadily send the packets comparably, although one slight dip in the number of packets sent is seen in RED at just after 4 seconds. Although all three approaches maintain a queue, they are also able to comparably send packets. In fact, there is very little difference in the number of packets sent between the two routers used in the case study. However, whilst in DropTail and RED, those packets not sent are dropped, DiDDeM is comparing these packets for early detection of denial-of-service attacks. As these figures demonstrate, DiDDeM has little effect on the number of packets sent, and therefore network performance.

5. CONCLUSIONS AND FURTHER WORK

The flow of information is the most valuable commodity for organisations and users alike, and denial-of-service attacks pose a great threat to this flow. These attacks are highly prevalent despite the widespread deployment of network security countermeasures such as firewalls and intrusion detection systems. These countermeasures find denial of service extremely problematic, therefore a number of other approaches have previously been proposed to counter the problem. However, these

approaches are not without their own problems, thus a new approach of early detection of denial-of-service attacks is required.

The paper demonstrates that statistical measures may be used to provide signatures of denial-of-service attacks. However, traditional uses of these techniques are labour intensive and require state information, which is computationally exhaustive within the routing infrastructure. Therefore, this paper has demonstrated the way in which a routing congestion algorithm may be adapted to provide statistical signatures statelessly. The results from our case study in *ns2* demonstrate the applicability of this approach in that it is highly effective in the detection of attacks without impeding on network traffic. This approach provides three benefits: computational load is reduced on the defence mechanism, even during an attack; no state information is required, again reducing computational load; and alerts to attacks may span several packets, thus reducing network load during an attack. In this way countermeasures can be placed beyond the perimeter and within the routing infrastructure to prevent attacks from reaching their intended victim.

Future work concentrates on the application of the system to other attacks that require or generate a high volume of traffic such as worms and malicious mobile code.

REFERENCES

1. Department of Trade and Industry / Price Waterhouse Coopers, "Information Security Breaches Survey 2004," Technical Report, http://www.dti.gov/industries/information-security (2003).
2. Moore, D., Voelker, G.M. & Savage, S., "Inferring Internet Denial-of-Service Activity," in *Proceedings of 10th Usenix Security Symposium*, Washington, DC (2001).
3. Anonymous, "Russian blackmailers arrested: Peace at last for online bookmakers," Computer Fraud and Security, vol. 2004, no. 8, p.1 (2003).
4. Gil, T.M. & Poletto, M., "MULTOPS: a data-structure for bandwidth attack detection," in *Proceedings of USENIX Security Symposium*, Washington, DC, USA (2001).
5. Meadows, C., "A cost-based framework for analysis of denial of service in networks," *Journal of Computer Security*, vol. 9, pp. 143-164 (2001).
6. Brustoloni, J.C., "Protecting Electronic Commerce from Distributed Denial-of-Service Attacks," in *Proceedings of WWW2002*, Honolulu, Hawaii, USA (2001).
7. Papadimitratos, P. & Haas, Z.J., "Securing the Internet Routing Infrastructure," *IEEE Communications Magazine*, vol. 40, pp. 76-82 (2002).
8. Shan, Z., Chen, P., Xu, Y. & Xu, K., "A Network State Based Intrusion Detection Model," in *Proceedings of Computer Networks and Mobile Computing 2001 (ICCNMC)*, Beijing, China (2001).

9. Sterne, D., Djahandari, K., Balupari, R., La Cholter, W., Babson, B., Wilson, B., Narasimhan, P. & Purtell, A., "Active Network Based DDoS Defense," in *Proceedings of DARPA Active Networks Conference and Exposition (DANCE '02)*, San Fransisco, CA, USA (2002).
10. Ioannidis, J. & Bellovin, S.M., "Implementing Pushback: Router-based Defense Against DDoS Attacks," in *Proceedings of Network and Distributed Systems Security Symposium*, San Diego, CA, USA (2002).
11. Kuzmanovic, A. & Knightly, E.W., "Low-Rate TCP-Targeted Denial of Service Attacks," in *Proceedings of Symposium on Communications Architecture and Protocols (SIGCOMM)*, Karlesruhe, Germany (2003).
12. Verwoerd, T. & Hunt, R., "Intrusion detection techniques and approaches," *Computer Communications*, vol. 25, pp. 1356-1365 (2002).
13. Stephens, M.A., "EDF Statistics for Goodness of Fit and Some Comparisons" in the *Journal of American Statistical Association*, vol. 69, pp. 730–737 (1974).
14. Floyd, S. & Jacobson, V., "Random Early Detection Gateways for Congestion Avoidance," *IEEE/ACM Transactions on Networking*, vol. 3, pp. 365-386 (1993).
15. Pan, R., Prabhakar, B. & Psounis, K., "CHOKe: A stateless active queue management scheme for approximating fair bandwidth allocation," in *Proceedings of IEEE INFOCOMM 2000*, Tel Aviv, Israel (2000).
16. Haggerty, J., Shi, Q. & Merabti, M., "DiDDeM: A System for Early Detection of TCP SYN Flood Attacks", in *Proceedings of Globecom 04*, Dallas, TX, USA (2004).