Requirements for a General Framework for Response to Distributed Denial-of-Service

D.W. Gresty, Q. Shi, M. Merabti

School of Computing and Mathematical Science,
Liverpool John Moores University,
Byrom Street, Liverpool, L3 3AF, UK.

D.Gresty@livjm.ac.uk

Abstract

What is network denial of service (DoS), and why is it such a problem? This research project has sought to investigate these questions and look at the deeper questions such as can denial of service be removed, can it be detected and can network systems adequately respond to denial of service incidents should they become subjected to them?

This paper describes some issues that make network denial of service a difficult security problem, and discusses some solutions that have been provided by the security research community. The paper then provides a classification of denial of service, the Consumer problem and the Producer problem, which forms the bulk of modern network denial of service incidents. Finally the paper proposes requirements for a framework for the management of response to network denial of service incidents, and suggests future directions that are being developed to create the framework.

1. Introduction and Background

In February 2000 [2,16,21] the Internet was subject to a mass distributed co-operative attack incident known as a Distributed Denial of Service. This incident brought a stark reality to the Internet E-Commerce community as small hosts attacked large allegedly well-protected systems. As the scale of the incident was unprecedented within the computer security literature and beyond the experience of the majority of the network entities involved, the response to this situation was ad hoc and demonstrated a clear lack of preparation. For the network entities involved there were very different operating policies or even radically different operating systems to the other entities that were victims during the incident.

In the joint CSI/FBI paper outlining the issues and trends in computer crime [25-26], it is shown that denials of service incidents within computer systems are a real problem. This problem is on a steady increase with 25% of the companies surveyed in 1999 reporting that they had suffered from denial of service and 27% in 2000. Research from experts such as Cohen [5] predicted that distributed co-operative attack (DCA) incidents would become more and more common in the future, and this prediction has to date, held true.

Denial of service is essentially the problem of an entity within a system (e.g. a user), preventing authorised entities (e.g. other users or programs) having access to resources (e.g. data files, programs or network connections) held within the system. Within the conventional model of security, denial of service is considered to be an availability problem. Availability is in fact referred to as "Denial of Service" in [1]. This means that secured assets are considered a service, and another asset or party denies access to the service. However services may also be denied if a party calls for an asset and finds that an unauthorised modification has occurred to the asset. If the party is unable upon subsequent calls to acquire an un-compromised asset then it has also been subject to a denial of service. This is an integrity problem, and therefore the denial of service problem is clearly not just availability of a service, but also the accuracy and integrity of that service.

As could be expected with this problem, there have been several solutions proposed to secure the reliability of computer systems and combat the denial of service problem specifically. Unfortunately there are no solutions that can guarantee 100% service and security as stated by Pfleeger in [23] and Loscocco in [19] as the problem domain is too wide and the system infrastructures used are not able to guarantee service. Even with modern network infrastructures using technology such as Asynchronous Transfer Mode (ATM) it is currently impossible to guarantee service despite the much stronger level of authentication and technological reliability. From this evidence it can be argued that denial of service is an intractable problem.

Unfortunately denial of service is not a static problem, as it has grown and evolved with the growth and development of computer networks. Distributed denial of service (DDoS) is the manifestation of this problem in distributed systems. With the growth of modern

distributed systems impacting upon the lives of millions of people, this kind of network problem needs to be removed. This research project suggests requirements for response to this problem. From these requirements a general framework could be developed to 'combat' the denial of service problem – a problem that currently has no significantly credible solutions.

In section two of this paper, some of the problems that make the denial of service problem so difficult are described. Section three presents a discussion of the different types of denial of service incident and classifies this into two separate problems. Section four proposes the research issues for development of a framework for managing incidents and the components that a general framework for responding to such incidents would require. The paper concludes with a summary and future work to be performed within this research project in section five.

2. Literature Review

This section presents a review of technical and non-technical issues that make the denial of service problem particularly difficult to address.

Distributed Co-ordinated Attack Incidents

Cohen discussed the problems of the Distributed Co-ordinated Attacks (DCAs) in [5] and identified perhaps the most important issue concerning denial of service incidents on the Internet: The primary problem associated with DCAs is trust; The network infrastructures are untrustworthy; and even when the technology can be trusted the human element can never be trusted implicitly. If an attack trace passes through several administrative systems, then the lack of just one of the systems in co-operation can greatly confound the tracing task.

Cohen addresses the point that one person need not perpetrate this form of attack. This paper was written prior to the distributed denial of service 'toolkits'. The author makes much of the painstaking problem of tracing the attack back to the source, but doesn't suggest what kind of response is suitable other than strong filtering, i.e. switching off all possible problem transactions.

Distributed Denial of Service Incidents

In the late 1999's the Distributed Denial of Service (DDoS) was highlighted as being of immediate concern to the network security professionals by organisations such as 'Internet Security Systems' after they had analysed freely available programs that had been gained from the Internet. This attack mitigates the need for complex co-ordination between attackers as would have been needed under Cohen's DCAs, which was arguably the reason that the February 2000 incident [2] did not occur sooner. This is because as few as one attacker can distribute and direct the attack. The attacker accomplishes this by penetrating systems with 'Trojan Horse' programs (see [23] for a description) that are very careful as Geng [24] mentions to do nothing to damage the software or hardware on the penetrated systems. At a predetermined time or after a signal has been sent by the owner of the so-called 'Zombie' machines, the penetrated systems will launch large volumes of traffic at the designated target.

Solutions proposed in the paper by Geng et al. suggest that creating an economic and technical expense for the 'Zombie' systems would provide the necessary incentive for system administrators to toughen up security to identify when the systems are being used to perpetrate hostile incidents and prevent them. The technical suggestions are introducing small problems such as solving a mathematical problem before being permitted to make a connection, or having an egress-limiting bottleneck so only a limited number of transactions can be sent from the system. Non-technical solutions include a pricing structure such that hosts are charged for sending transactions like an electronic postal service. This solution the authors suggest would provide the incentive for system owners and ISPs to monitor for hostile behaviour to reduce the extra unnecessary network costs.

Ingress/Egress filtering

The Internet Society suggested in [11] that network ingress filtering was a valid way to remove spoofed addresses from entering the network. This technique is implemented by an 'upstream' service provider to compromised hosts, and as the compromised hosts send information through the provider, the provider removes transaction packets that are obviously going to be used in an incident through the use of illegal addresses. This approach represented an attempt at self-regulation by the Internet community, rather than the economic restrictions as suggested in [24]. CERT still recommends the advice contained in this request for comment in its advisory on the developments of denial of service [27], however it is very careful to point out that this will reduce spoofing, not eliminate distributed denial of service. This is because the denial of service tools that have been analysed by Dettrich in [6-8] do not need to use spoofed addresses to attack their targets. Therefore this approach although considered effective against the older style attacks, has been superseded by the advances in denial of service 'technology'.

Network security group UC Davis

The work [3,4] that this group has concentrated on is the protection of network routers from denial of service incidents. The basic premise is that existing protocols are not equipped to deal with denial of service attacks. One of the most important aspects to come out of this work is the identification of co-operation within the network environment. This work identifies that attackers can

collude to hide evidence of the attack. Colluding attackers may even be able to discredit innocent network parties by sending fraudulent messages to intrusion detection systems (IDS) on the network. As a response to this, the group has decided to use a co-operative intrusion detection model to effectively co-operate in the detection of what the papers refers to as "misbehaving routers". This work was very important as it highlighted that a stand-alone system could not adequately detect co-ordinated attacks and therefore must co-operate to stand a fair chance of detection.

There are a couple of points within this work which need noting. The WATCHERS protocol developed to implement these ideas follows the US aphorism "It's my way, or the highway". That is to say all participants must adhere strictly to the protocol, or risk 'banishment'. This principle would be perfectly acceptable for a single type of protocol. However this technique is quite obviously not going to be acceptable to every party on a network for general transactions, as such it would be more suitable for a restricted environment (such as the control of routing protocols, as it was originally intended).

Summary

The primary issue for the Internet is trust, or more importantly the basic lack of trust in any human influenced transactions, as the Internet was not designed with the current demands it faces in consideration. Many of the potential solutions that are highlighted by the research community suggest essential changes to the Internet or the protocols to increase the level of trust technically, and to make untrustworthy behavior by network entities undesirable.

This strategic response, by the research community, is opposed to the current commercial attitude which is the introduction of entities into the network that can attempt to assure security by increasing the technological cost of attacking systems, e.g. the introduction of firewall technology, cryptography etc. This is because there is no central authority or responsibility for how the Internet operates other than the issue of cost. With cost as a primary consideration, the attackers are always going to be able to develop attacks faster than it takes to introduce stumbling blocks for the attacks.

The attack technology has developed and to some extent come full circle as denial of service attacks were often originally used to assist intrusion attempts by bringing down systems so that the addresses could be spoofed. Denials of service attacks have become a threat by themselves, unrelated to system intrusions. Now the DDoS toolkits require system penetration and Trojan horse programs, so the system intrusion is now motivated by the denial of service attack. This returns to the trust issue as there is now no longer a need for an attacker to spoof addresses, if the attacker can penetrate just one program onto a system, the question must be asked "can every user be trusted to have a high level of security awareness"?

To prevent denial of service within a distributed environment, what is required is a network that can fairly guarantee delivery of services and requests from a purely technical aspect. However the purely technical solution also needs fair human-computer transactions i.e. the administrators are required to permit the computer systems to behave in a fair manner.

By the above-mentioned criteria the effective operation of the system requires that every party behave fairly in every transaction. Any party therefore misbehaving can cause the distributed system to be exploited to deny service to any other party. Therefore there can be no true technical solution to the Denial of service problem, however many researchers have sought to develop systems that make this possibility undesirable and difficult.

3. Classification of Denial-of-Service

Denial of service is not a difficult problem to detect. Gilgor's work [12] highlighted that using a maximum waiting time it was a simple matter to identify that a service or request had not been delivered within that specified time. That paper identifies two important points: the acceptable time is predetermined as an operational policy requirement; and without an alternative route to request resources or direct requests then there is no way to do anything other than detect the denial of service.

However denial of service is not a single problem. It is in-fact two clearly separate problems, the Producer and Consumer problems as highlighted in [17]:

- Consumer attacks
  A Consumer attack is when a party *C* seeks to consume another party *A*'s request for a resource to a third party *B*, or consume the actual resource that *B* has allocated to *A*.

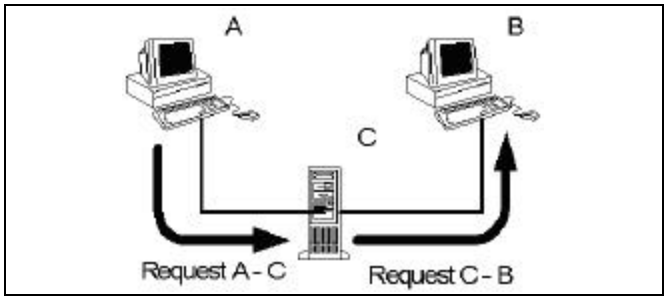

Figure 1. Request passing from A to B

For example, Server A is requesting a resource from Server B, via Server C. Figures 1 and 2 show the normal sequence of events for this request and supply transaction. Figure 1 shows the request passing from A to C, and from

C to B. Figure 2 shows the requested resource then passing from B to C, then from C to A.

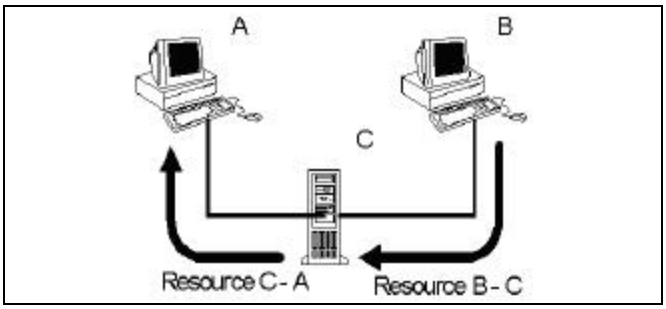

Figure 2. Resource allocation from B to A

Figures 3 and 4 show the consumer problem. In figure 3, no request can reach B as Server C is refusing to pass requests from A to B. Figure 4 shows a request (Fig 1) has been successful but the resource that B is allocating to A is unable to reach A due to C consuming that resource.

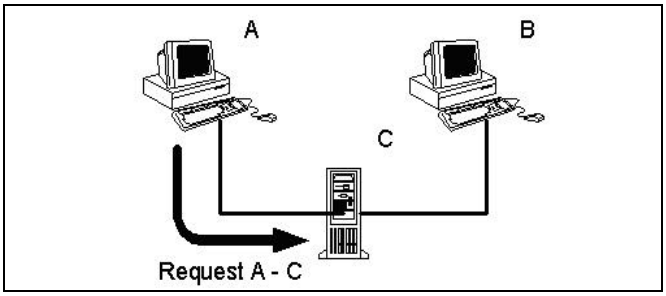

Figure 3. Refusing to pass requests

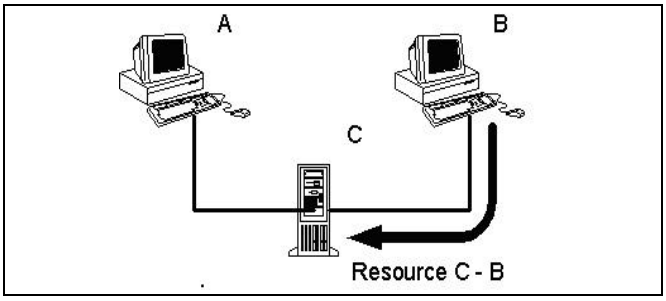

Figure 4. Consuming allocated resources

- Producer attacks
  This is where parties C_1-C_n causes a resource to be made available to another party *A*, which violates *A*'s operating policy.

Figure 5 shows the Producer problem. In this example Server A is connected to a number of other servers $C_{1\text{-}n.}$ A receives a number of resource from the servers it is connected to. In general, if a server increases the number of resources it has access to this cannot normally be considered a problem. The increase in resources is realistically an increase in functionality. The Producer problem is however based upon the resources being made available violating Servers A's operating policy – that is to say that Server A is unable to manage those resources. For Server A to determine that the resources being made available are unsuitable or unmanageable, requires A to use its own internal resources. Server A has only a finite number of internal resources and if it has to manage a large number of 'made available' resources this can force it to use up all the internal resources, therefore Server A can no longer operate – it has had its operating policy violated.

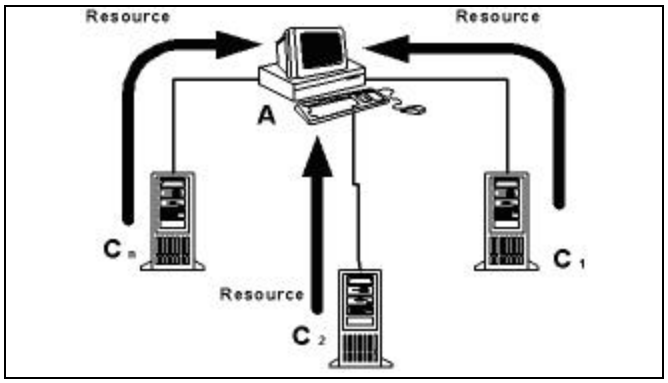

Figure 5. The Producer problem

The Consumer problem (Figures 3 and 4) is the classical Man-in-the-Middle problem, where party C sits between parties A and B. It is a simple matter for party C to perform this attack. The only effective defence against this form of attack is for party A to identify the attack, by using Gilgor's [13] principle of maximum waiting time and route through an alternative channel if available.

The Producer problem can be considered the 'modern' network denial of service or DDoS. The Producer problem clearly highlights the untrustworthy nature of the network entity transactions, as there are a number of policy constraints which if violated cause this type of incident. As noted above, the victim must identify if any policy constraint has been violated when it accepts a resource. This acceptance test causes the victim to use its own internal resources to search through a large number of possible violations. The types of policy constraints that have been identified as capable of violation within this work are:

1. The resource must arrive in a timely manner
2. The resource must contain correct information
3. The resource must be answered (if required)
4. The resource must actually arrive
5. The resource must be well formed (i.e. not corrupted)
6. The resource must be in the correct format

Descriptions:

- Points 1 and 4 are distinct due to the term 'timely'. An operating policy will have the acceptable arrival time specified. If the element fails to arrive within that time, then it is acted upon with regard to the policy, which will most likely mean the element being discarded. Point 4 has an arrival time of infinity, therefore the operating policy is irrelevant

and the element's lack of arrival is dependent purely upon the communication medium.

- Point 2 refers to the element containing correct information. This is important because the receiver is expecting to be given information that is correct, and if it does not then it cannot perform the action that it wanted to, i.e. the correct information has been denied.
- Point 3 covers incidents, where the element can be considered denied until it is acknowledged [22]. If a communications element from party A arrives at party B and is not acknowledged, then clearly party A has no idea if the element has got through to party B, and may well try to re-send the element depending upon A's operating policy.
- Points 5 and 6 are very similar yet distinct. Point 5 states that the information in an element is useless if it has been changed which is an integrity issue, whereas point 6 states that the information is equally useless if it is laid out in a format that the receiving party doesn't understand. Point 5 is distinct also from point 2, as it covers useless corrupted information not information that is correctly formatted, which is worthless in content.

Within this section the denial of service problem is classified into two separate and very distinct problems: the Producer and the Consumer problems. The Producer problem is synonymous with modern DDoS. It has been demonstrated that there are many ways of making a resource available to a victim machine that can possibly violate that victim machines operating policies. This highlights the complexity of the Producer Problem and any framework to respond to DDoS must consider the producer problem in general.

4. Requirements for Responding to Network Denial of Service

There are two clear research directions for managing network denial of service incidents: prevention of incidents prior to occurrence; and response to incidents as they are occurring. There are no general solutions to the general denial of service problem, or any guides or frameworks for dealing with incidents. There are specific problems that can be readily identified such as SYN flood, as these are incidents that have a specific symptom and effect. Solutions have been developed for many of these specific instances, however they do not solve the general capability for certain parties on a network to be able to cause incidents, as the capacity to violate the policy constraints was demonstrated to be a simple matter in the previous section. Gilgor argues in [14] that prevention of network denial of service is virtually intractable.

When policy is considered crucial and it is accepted that denial of service is an intractable problem, then prevention of denial of service becomes an unreasonable goal. This project identified that response to incidents and detection of incidents are important for providing assurance to networks requiring rapid reaction to DDoS incidents. A general framework for responding to DDoS incidents would have several desirable requirements for the framework of operation:

a) Generality – Denial of service may occur across many different platforms within a computer network and development of a framework restricted in its domain of application is not desirable. For example a framework for the detection of UNIX denial of service gives less insight into combating the same denial of service on a Microsoft NT system, than a general theory that could be deduced to the specific systems.

b) Not exploitable – The framework should be secure, but as Pfleeger [23] points out that no system is 100% secure, therefore the framework must not be able to be exploited such that it can cause a denial of service. Other aspects such as fault tolerance [18] and confidentiality must also be considered as operational policy constraints.

c) Policy – An integral part of denial of service is the traditional concept of integrity and availability, but as technology advances, networks are required to perform a variety of different roles. Now the model should be confidentiality, integrity and acceptability. This acceptability as defined by the policy of operation, needs to be a fundamental framework requirement. There are many possible operational policy constraints, with some that have been highlighted from the literature survey and many others which need more thorough research:

    - Timely – The term "real time" as noted in [20] is something of a misnomer, when it comes to intrusion detection. However for the framework to be effective it must be considerate to the speed of operation and the cost/benefit evaluation of time/detection.
    - Survivable/Adaptive – From the work in [10] it can be shown that a system must be able to provide core functions while compromised. This is the concept of system survivability, and it is central to 'policy directed adaptability'.
    - Scalable – The framework needs to be equally applicable to a small-scale network as it does a large distributed internet.
    - Reaction – The victim must be very clear on the policy of normal operation and the policy of dealing with hostile incidents.

From the list of requirements that have been derived from a wide survey, it can be shown that no existing technique or system currently can meet all these requirements and the requirements that will almost certainly be derived from further investigation.

The focus of research for future work is on system response to a large-scale network producer flood incident. For a large distributed network to effectively recover from such an incident requires time and a fault tolerant capacity to move to a stable state. If a network incident is not caused by 'normal' random anomalous behaviour, but is rather a co-ordinated hostile incident, it is conjectured that the only effective response is for the entities to deliberately co-ordinate the stabilising procedure. This co-ordinated stabilisation should ensure that the critical functions within the network could be restored in near optimal time. Research issues that will be of critical importance to develop a framework that can manage these incidents would be:

1) Derive 'Policy Primitives'. These primitives can be used to represent the states of the operational behaviour of systems. The objective is to find the implementation independent constraints that are needed to understand system operation for interaction with other network entities. It is hypothesised that these policy primitives are needed for negotiation between heterogeneous and more importantly 'hetro-policy' systems when trying to reach mutually acceptable states. This level of system co-operation is currently unheard of and therefore the issue of acceptable co-operation through policy is crucial. Research issues such as at what level of abstraction can the operational policies be accurately modelled need to be addressed. The co-operation of different administrative systems indicates a confidentiality problem. There must be detailed investigation into the effective communication of fair and accurate operational policies between different administrative systems.

2) There is a need for detailed analysis of network behaviour and traffic analysis. This work alone is not new, however there is need for hard empirical data concerning how systems react as mass flood incidents are occurring. This empirical data combined with theoretical work concerning network architecture design will give an important indication to the response time that is available for response. Research issues are the effects of differing network topologies, differing implementations on specific operating systems and the different types of incidents. The speed that networks fail when subjected to incidents, and the timing between the anticipated stages of degradation is critical and currently there is no information in the public domain to give indications concerning this. These timings and bands of degradation will act as guides to determine what is technically feasible for the framework to regard as an adequate response and to assess this against what is realistically and theoretically acceptable as adequate response.

3) Integration of stabilising techniques to develop the framework to respond to incidents. The area of network theory that is concerned with stabilisation of network is well understood and documented such as in [9]. However this is a developing field and this project would have to examine issues such as negotiation of network links, dynamic routing and capacity determination. Work within network theory often considers a link to be operation or failed. However study is needed to evaluate this with respect to denial of service, as there are few occasions when a link is totally failed, but rather the traffic can variably be greater than the receiving capacity. Also the network theory considers failures to be the nodes or link failures. Denial of service needs to be determined to be either a link or node or perhaps a third as yet unknown type of failure before the algorithm developed within network theory can be modified to optimise recovery from such incidents.

Summary

There are many specific problems, exploits and solutions associated with network denial of service. No-one has however presented a general solution that is adequate for managing denial of service incidents. There are two possible directions for managing denial of service incidents, the prevention or the detection and response of incidents. It has been argued that prevention is intractable by Gilgor and as such response is considered a valid research direction.

Operational policy is critical to the examination of network Producer incidents, as they are caused by transactions that violate the policy. From the argument that managing Distributed Co-ordinated Attack incidents will require distributed co-operative response by several systems, it can be seen that those systems would be required to agree to policy directed strategies for managing the incidents. That essentially means that the stabilization protocols would need to be acceptable to all the systems involved in the co-operative response and therefore require those multiple systems to agree on an effective policy or strategy.

The details of the response scenarios need to be examined as does the generic policy requirements of network entities, while these factors are of immediate concern to the research into the development of a framework for response to denial of service incidents.

5. Conclusion

This paper has highlighted that denial of service is not a new problem and there have been many solutions proposed for managing denial of service incidents. There has been a detailed investigation into the nature of denial of service and the Consumer and Producer problems have both been identified. Although both problems are of

concern to network entities, the Producer problem has been identified as a more serious problem as it can be simply invoked by a system providing sufficient resources to another system in violation of normal operating policy. The nature of the Internet does not allow the victim choice about what it receives, and therefore a victim system has very few options in what it can do to respond to such incidents.

Two research directions have become apparent from this investigation: Prevention and Response. It is argued that to prevent this kind of policy violation exploitation would require major changes to the protocols that drive the Internet, which is realistically unfeasible; therefore there must be in-depth analysis of response to such incidents.

Future work seeks to achieve a general framework that can realistically respond to distributed producer denial of service flood incidents. It will do this by developing a policy directed co-operative and adaptable framework. This framework will permit network entities to determine an optimal state that they wish to be at when they are in a degraded condition and this framework will allow them to negotiate steps to achieve that co-operative state. It is presumed a certain amount of load balancing or adaptive routing will be required to achieve this goal. Currently this research project has only highlighted the crucial need for the policy directed adaptability and can only hint at the other research issues to affect this framework.

References

[1] E. Amoroso, "Fundamentals of Computer Security Technology", Prentice Hall International, ISBN: 0131089293, 1994.

[2] L. Arent, D. MuCullagh, "A Frenzy of Hacking Attacks", Wired Online, February 2000. http://www.wired.com/news/business/0,1367,34234,00.html.

[3] S. Cheung and K. N. Levitt, "Protecting Routing Infrastructures from Denial of Service Using Co-operative Intrusion Detection," presented at Proceedings New Security Paradigms Workshop, Cumbria UK, 1997.

[4] S. Cheung, K. A. Bradley, N. Puketza, B. Mukherjee, and R. A. Olsson, "Detecting Disruptive Routers: A Distributed Network Monitoring Approach," Proceedings of the 1998 IEEE Symposium on Security and Privacy, Oakland, California, May 3-6, 1998, pp.115-124.

[5] F. Cohen, "Distributed Co-ordinated Attacks (DCA)", Management Analytics, April 1997. http://www.all.net/books/dca/.

[6] D. Dettrich, "The DoS Projects's 'trinoo' distributed denial of service attack tool", University of Washington, 1999. http://staff.washington.edu/dittrich/misc/trinoo.analysis

[7] D. Dettrich, "The 'stacheldraht' distributed denial of service attack tool", University of Washington, 1999. http://staff.washington.edu/dittrich/misc/stacheldraht.analysis

[8] D. Dettrich, "The 'Tribe Flood Network' distributed denial of service attack tool", University of Washington, 1999. http://staff.washington.edu/dittrich/misc/tfn.analysis

[9] S. Dolev, T. Herman, "Superstabilizing Protocols for Dynamic Distributed Systems", Chicago Journal of Theoretical Computer Science, MIT Press, December 1997.

[10] R. J. Ellison, et al. "Survivable Network Systems: An Emerging Discipline", Software Engineering Institute at Carnegie Mellon University, CMU/SEI-97-TR-013, November 1997.

[11] P. Ferguson, "RFC 2267, Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing", The Internet Society, January 1998.

[12] V.D. Gilgor, "A Note on Denial of Service", Transactions on Software Engineering, Vol. SE-10, pp. 320-324, 1984.

[13] V.D. Gilgor, "On Denial of Service in Computer Networks", pp. 608-617, 1986.

[14] V. D. Gilgor and C. F. Yu, "A Formal Specification and Verification Method for the Prevention of Denial of Service," presented at IEEE Symposium on Security & Privacy, 1988.

[15] V.D. Gilgor and C. F. Yu, "A Specification and Verification Method for Preventing Denial of Service," IEEE Transactions on Software Engineering, vol. 16, pp. 581-592, 1990.

[16] T. C. Greene, "The Mother of all DDoS attacks looms", 24th February 2000. http://www.theregister.co.uk/000224-000001.html.

[17] D.W. Gresty, Q. Shi, E.P. Moynihan, "Survivable Systems Concept To Protect Core E-Business Functions from Denial-Of-Service", presented at BIT 2000, November 2000.

[18] K. N. Levitt, S. Cheung, "Common Techniques in Fault-Tolerance and Security", Proc. of the Dependable Computing for Critical Applications 4, San Diego, California, 4-6 Jan. 1994, pp. 373-377

[19] P.A. Loscocco et al, "The Inevitability of Failure: The Flawed Assumption of Security in Modern Computing Environments", presented at NIST'98, 1998. http://csrc.nist.gov/nissc/1998/proceedings/paperF1.pdf

[20] S. Northcutt, "Network Intrusion Detection: An Analyst's Handbook". New Riders, ISBN: 0-7357-0868-1, June 1999.

[21] C. Oakes, "DoS: Defence Is the Best Offence", WIRED online, February 10th 2000. http://www.wired.com/news/technology/0,1282,34230,00.html.

[22] R. Perlman, "Network Layer Protocols With Byzantine Robustness," in Electrical Engineering and Computer Science: MIT, 1988, pp. 121.

[23] C. P. Pfleeger, "Security in Computing", ISBN: 0131857940, Prentice Hall, 1997.

[24] X. Geng, A.B. Whinston, “Defeating Distributed Denial of Service Attacks”, IT Professional, Vol. 2, No. 4, pp17-22, July-August 2000.

[25] “Issues and Trends: 1999 CSI/FBI Computer Crime and Security Survey”, Computer Security Institute, 1999.

[26] “Issues and Trends: 2000 CSI/FBI Computer Crime and Security Survey”, Computer Security Institute, 2000

[27] “Denial-of-Service Developments”, CERT® Advisory CA-2000-01, Carnegie Mellon University, January 2000. http://www.cert.org/advisories/CA-2000-01.html