An intelligent approach to prevent distributed systems attacks

Nasser S. Abouzakhar
The Centre for Mobile Communications Research (C4MCR), University of Sheffield, Sheffield, UK
Gordon A. Manson
The Centre for Mobile Communications Research (C4MCR), University of Sheffield, Sheffield, UK

Keywords
Information technology, Fuzzy logics, Networks, Sabotage, Disaster management, Automated operations

Abstract
In today's society, information and communications technology (ICT) is the force that drives prosperity and provides a higher standard of living. All other e-services and infrastructures tend to play a major role in our daily life and global economy. The growing dependence on such systems, however, has increased their vulnerability to cyber attacks. Any failure to these systems typically would lead to a huge impact, not only on businesses, but also human life, that depends on such interconnected systems. The growing potential for telecommunications network infrastructures problems stems from their nature of openness. A successful attempt for a network attack to a particular organization's network could have devastating effects on the security of the organisation. In this paper we propose an innovative way to counteract distributed protocols attacks such as distributed denial of service (DDoS) attacks using intelligent fuzzy agents.

Introduction

The increase in the number of interconnected networks to the Internet has led to an increase in security threats, such as denial of service (DoS) attacks. The existing network servers, routers' communications protocols, firewalls and intrusion detection systems (IDSs) are not well designed to deal with DoS attacks. Networking protocols such as TCP/IP, which were designed to be used in an open and trusted community, have inherited flaws. In addition, many network operating systems and devices have flaws in their network stacks that weaken their ability to withstand DoS attacks, which are difficult to defend against (Needham, 1994). Currently, firewalls are used to block ICMP packets to avoid Smurf DoS attack. However, external legitimate accesses will still be prevented, because the perimeter firewall will be swamped by the ICMP_ECHOREPLY packets. And it should be taken into consideration that blocking ICMP packets does not represent a proper solution to the problem. ICMP protocol plays a useful role in debugging and maintaining networks. Prevention of DoS is a fundamental objective of protection of resources availability, which is one of three common information security objectives (ISO, 1988). An effective solution to DoS attacks must be based on the control of resource allocation (Gligor, 1983). Needham (1994) suggests that resources availability is the major concern of information systems design, and other security objectives are less important. As violations of availability easily lead to considerably long interrupts of services, they may result in serious disruptions of businesses. Shortening time to detect and respond to DoS attacks would ensure continuous resources availability. The success of an attack is dependent on the time gap between detection and response (Cohen, 1999). This research suggests innovative ways to deploy intelligent fuzzy agents to address different types of distributed denial of service (DDoS) attacks with the ability to respond quickly, and dynamically control the resources allocation and ensure their availability for legitimate users without blocking those useful protocols.

The approach of DDoS attacks

It is often much easier to disrupt the operation of a network or system than to actually gain access. Networking protocols such as TCP/IP, which were designed to be used in an open and trusted community, have inherited flaws. In addition, many network operating systems and devices have flaws in their network stacks that weaken their ability to withstand DoS attacks. DoS attacks go far beyond mere annoyance. DoS attacks could cause major damage to vital systems. An organization that relies on electronic transaction for its livelihood could suffer serious financial loss if its systems were taken off line for even a short duration. DoS attacks could have a life-threatening impact in health care operation. DoS attacks work because computer networks are there to communicate. Some attackers target mail servers. Some attackers target routers. Some attackers target Web servers. The basic idea is the same: flood the target with so much useless traffic that it shuts down. DoS attacks do harm just by the attempt to deliver packets, whether or not the packets would authenticate properly is completely irrelevant. Mandatory authentication would do nothing to prevent these attacks (Schneier, 2000).

While there are many tools available to launch DoS attacks, it is important to identify the main types of DoS attacks, so

Information Management & Computer Security
10/5 [2002] 203–209

[ISSN 0968-5227]
[DOI 10.1108/09685220210447505]

later on to understand how to detect and prevent them. The three most common categories of DoS are:

1 bandwidth consumption;
2 resource starvation; and
3 resource exploitation.

The three categories are discussed in the following subsections (Northcutt *et al.*, 2001).

Bandwidth consumption

Bandwidth consumption represents the most insidious form of DoS attacks, where the attacker consumes all available bandwidth either locally or remotely. System performance problems will only be detected in DoS or computation-intensive attacks. The Smurf attack targeted a feature in the IP specification known as direct broadcast addressing. A Smurf attack is one of the most harmful DoS attacks due to the amplification of its effects, which result in consuming the targeted network bandwidth. The amplification effect is a result of sending a directed broadcast ping request to a network of systems that will respond to such requests. Unfortunately, most network sites require the use of the ping facility for network debugging purposes. A stateful firewall can help, by allowing ping replies to ping requests originating from within the internal network only. Although this will allow activity to continue within the internal network, external access will still be prevented, because the perimeter firewall will be swamped by the ICMP_ECHOREPLY Packets (Northcutt *et al.*, 2001).

Resource starvation

Resource starvation is meant to consume system resources, such as CPU utilisation, queuing buffer, or other system processes. Resource starvation DoS attacks (Criscuolo, 2000) generally result in unusable system resources. When a TCP connection is initiated, it is a three-way process. TCP SYN Flood represents a resource starvation DoS attack. When the SYN ACK is destined for an unavailable host, the last part of the "three-way handshake" is never completed, the resource allocated is useless and an entry remains in the connection queue until a timer expires, typically for about one minute. While waiting for the ACK to the SYN ACK, a connection queue of finite size on the destination host keeps track of connections waiting to be completed. This queue typically empties quickly since the ACK is expected to arrive a few milliseconds after the SYN ACK.

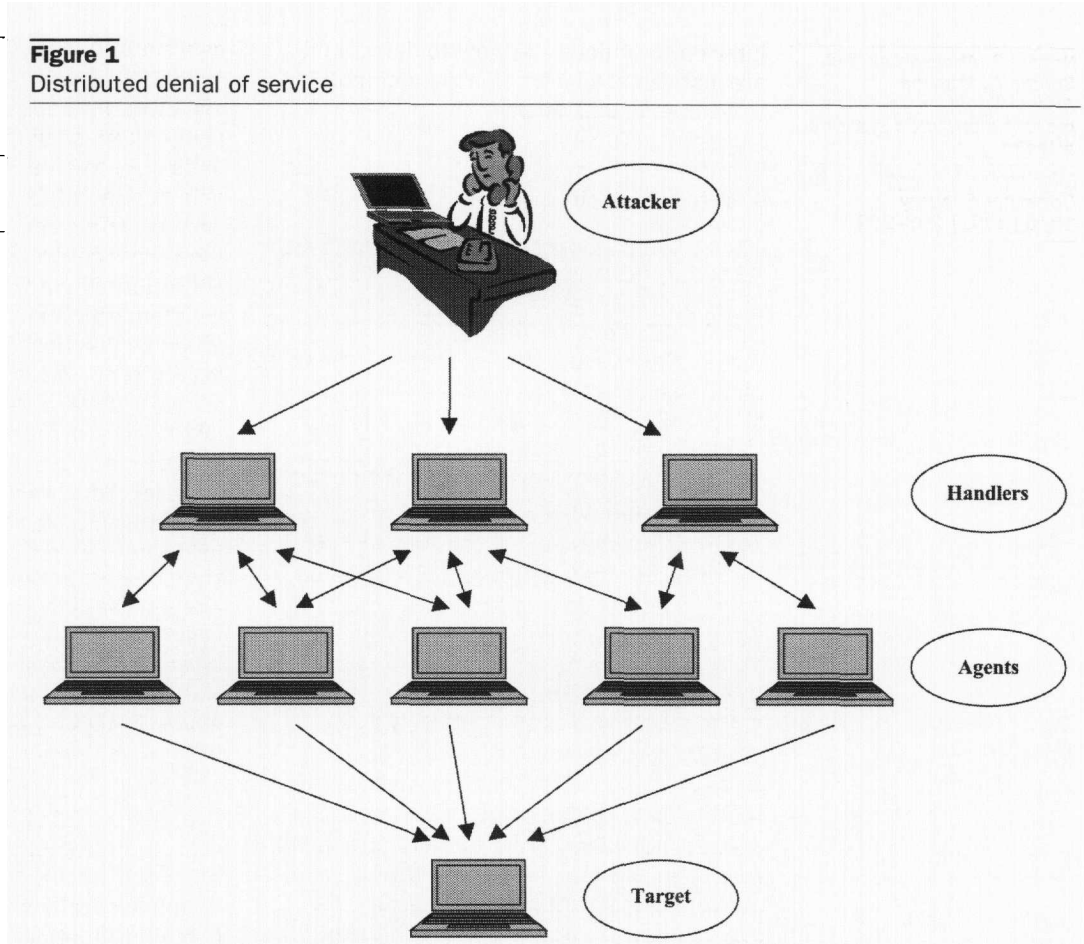
Resource exploitation

Resource exploitation attacks exploit a flaw in the target system's software in order to cause a processing failure or to cause it to exhaust system resources. Flaws exploitations represent failures of an application, operating system, or embedded logic chip to handle exceptional conditions, or by deliberately executing excessive cryptographic computations, for example at the access control server. These exceptional conditions normally result when a user sends unintended data to the vulnerable element or executes privileged commands, for example to kill local processes in a running Web server. Many companies are now doing business over the Internet, selling their products and services to anyone with a Web browser. However, poor e-commerce design can allow attackers to falsify values such as price and make a fundamental flaw in their HTML or XML coding. Firewalls and filters assume that a document's XML metadata tags honestly and accurately describe the code's content. Firewalls and filters thus could admit deceptively coded XML documents that contain malicious code. "It is not practical for a filter to examine the code, it is computationally intensive and expensive to analyse a lot of incoming code all the time, especially from multiple sources" (Angeles *et al.*, 2001).

Buffer overflow is one of the resource exploitation attacks that allows attackers to put a value greater than expected into a program variable, and by doing so, execute arbitrary code with the privilege of the running user. This is what makes it possible to take down a Web server with only a Web browser. By utilising the width value fields in HTML or XML coded page, attackers change this value to a large number like 100,000 and submit a large string of characters to a Web server. Web programmers put functionality before security. The problem of poorly designed Web servers can be more difficult to tackle, especially once the design is in place.

A DDoS attack harnesses the distributed nature of the Internet. DDoS attack uses many computers to launch a coordinated DoS attack against one or more targets, as shown in Figure 1. The attacker is able to multiply the effectiveness of the DoS significantly by harnessing the resources of multiple unwitting accomplice computers, which serve as attack platforms. Typically, a DDoS master program is installed on one computer using a stolen account. The master program, at a designated time, then communicates to any number of "agent" programs installed on computers anywhere on the Internet. The agents, when they receive the command, initiate the attacks. The master program can initiate hundreds or even thousands of agent

Figure 1
Distributed denial of service



programs within seconds. Many other attacks can be mapped into the distributed model. For example, an attacker can set up a group of agents to conduct a more stealthy port scan or network mapping. Stacheldraht (Criscuolo, 2000) is one of the well developed DDoS attacks. Stacheldraht relies on TCP, rather than UDP, for transport. It attacks with ICMP, UDP, SYN, and Smurf-type attacks.

The proposed intelligent fuzzy agents

There are two intelligent fuzzy agents that have been proposed, one is located at the network router (router fuzzy agent (RFA)) to respond to network layer directed DoS attacks, such as Smurf. The other one is located at a network server (server fuzzy agent (SFA)) to respond to transport layer directed attacks, such as TCP SYN Flood. Both fuzzy agents incorporate fuzzy inferencing to check if limited conditions are about to be exceeded by fuzzifying defined DoS process variables. These variables are examined as to their fitness, or degree of belonging to fuzzy sets defined in linguistic variables.

Once the RFA receives Smurf type of traffic, it follows the rules defined to overcome the router's buffer overloading and smoothly avoid any deliberate excessive bandwidth consumption and traffic congestion. A sample fuzzy rule situation is expressed by the phrase:

IF the Router Spoofed-Rxed-ICMP-ECHOREPLY packets is High
AND Txed-ICMP-ECHOREQUEST is Low
THEN AllocatedBuffer is Low.

The fuzzy buffer allocation is a 2 * 1 (a "two-by-one") process. That means there are two input variables and one solution variable. It is often convenient to think of such a process as a matrix of actions in an M*N array. The fuzzy states of one input variable form the horizontal axis, and the fuzzy states of the other input variable form the vertical axis. At the intersection of a row and column is the fuzzy state of the solution variable. This form of representation, very common in control engineering field, is called a fuzzy associate memory (FAM). Figure 2 shows a FAM for the buffer allocation controller implemented by the RFA, which includes all nine required rules.

The RFA only allows controlled incoming ICMP-ECHOREQUEST packets from a private intranet with known source IP

Figure 2
The FAM for the buffer allocation controller to overcome Smurf DoS attack

Txed-ICMP-ECHOREQ \ Spoofed Rxed-ICMP-ECHO REPLY	Low	Medium	High
Low	Low	Low	Low
Medium	Medium	Medium	Medium
High	High	High	High

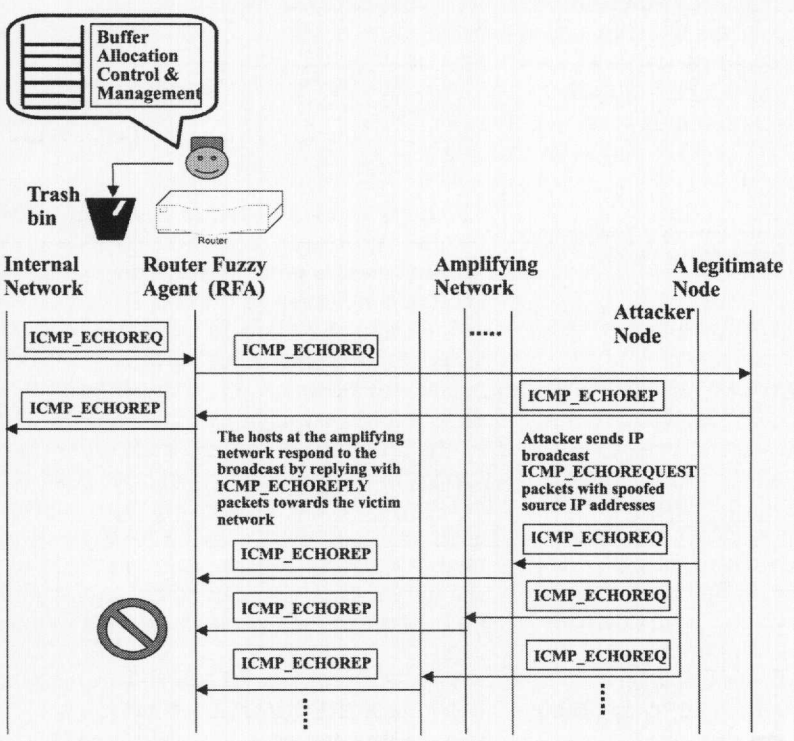
addresses and within the limits specified in the fuzzy rules, as shown in Figure 3. The RFA represents a network IDS sensor, in which specific threats are being sought. By reducing and specifying the number of attacks that agents are looking for, a network IDS will increase its performance.

The SFA responds to TCP SYN Flood attacks by performing several processing steps on every TCP packet received at the server. The problem here is that most systems allocate a finite number of buffering resources and add an entry to the connection queue when setting up a potential connections or connection that has not been fully established. While most systems can sustain hundreds of concurrent connections to a specific port (for example, 80), it may only take a dozen or so potential connection requests to exhaust all resources allocated to setting up the connection. This can lead to filling up the connection queue and denying TCP services, such as access Web server for legitimate users. As the TCP SYN flooding initialises many connections to the victim server without completing the TCP handshake process, each packet is dealt with by the SFA. The SFA generates TCP ACK for each TCP SYN packet to complete the connection. Any duplicated ACK packet received later on will be discarded. If a TCP ACK packet is delayed, the expiry timer triggers the SFA to generate a TCP RST packet, the connection is moved to CLOSED state and the resources are released and each TCP connection is allocated resources according to the defined fuzzy rules. This will block exceeding the resources limits. A sample fuzzy rule situation for preventing deliberate buffer starvation is expressed by the phrase:

IF Sever Rxed-TCP-SYN packets is High
AND TCP-ACK-Delay is High
THEN AllocatedBuffer is High

The FAM for the buffer allocation controller implemented by the SFA is shown in Figure 4. It includes all nine required rules.

Figure 3
The router fuzzy agent blocks (discards) the attacking ICMP_ECHOREPLY packets



Results and discussion

Our results indicate that intelligent fuzzy agents would establish the required distributed and real-time mechanisms, which are essential for any network protections. MATLAB (Jang *et al.*, 1997) simulation environment has been used to provide the fuzzy intelligence behaviour for the agents.

Let us consider modelling the concept of the Smurf DoS attack around the variable SpoofedRxed-ICMP-ECHOREPLY packets. The values of this variable view the number of such packets received by the router from

Figure 4
The FAM for the buffer allocation controller to overcome TCP SYN DoS attack

TCP-ACK-DELAY \ Rxed- TCP-SYN	Low	Medium	High
Low	Low	Low	High
Medium	Low	Medium	High
High	Medium	Medium	High

outside the internal network. The second variable defined is the Txed-ICMP-ECHOREQUEST packets, which represent the legitimate outgoing ICMP ECHOREQUEST packets from within the internal network. This variable has been chosen in order to compare it with any incoming ICMP-ECHOREPLY packets, including the previous attacking packets (i.e. SpoofedRxed-ICMP-ECHOREPLY) traffic. This is to ensure only the legitimate either Txed ICMP-ECHOREQUEST or Rxed ICMP-ECHOREPly packets are allocated at the buffer/queue within the router. The allocated buffer represents the consequent variable of the defined fuzzy rules. The defuzzification process adjusts the fuzzy sets of the buffer queue variable in accordance with the calculated possibilities resulting from the antecedent variables. The deffuzified value or balance point would represent the allocated buffer queue at the router.

The RFA's ability to control the resources allocated for routing network traffic, as shown in Figure 5, indicates that the allocated resources are manageable. A higher spoofed received ICMP_ECHOREPLY packets in conjunction with a lower rate of transmitted ICMP_ECHOREQUEST packets traffic coming in the opposite direction from an internal network yet still enables RFA effectively to prevent the router's buffering resources and network bandwidth from being overloaded. The RFA manages to drop all of the spoofed received ICMP_ECHOREPLY packets allowing most of the resources allocated for legitimate network traffic.

As the transmitted ICMP_ECHOREQUEST packets traffic starts to increase, the RFA continues to monitor and manage the buffer allocation so that it ensures enough storage space for legitimate users traffic, as shown in Figure 6.

The surface view shown in Figure 7 indicates a smooth buffer allocation for legitimate ICMP ECHOREQUEST packets, coming from either inside or outside the internal network. The RFA will not allow any buffer allocation for Spoofed-ICMP-ECHOREPLY packets. This will ensure buffer resources availability for legitimate network traffic and avoid deliberate bandwidth consumption, despite the presence of ICMP DoS traffic within the router.

The SFA is capable of controlling the usage of the queue (backlog) resources allocated for TCP SYN segments, as shown in Figure 8. The server application specifies the limit to this queue.

A highly received TCP SYN packets traffic in conjunction with highly delayed corresponding TCP ACK packets still enables the SFA to effectively prevent the server application queue from being unnecessarily filled. The SFA controls the number of connections already queued for any particular listening point, to see whether to accept a new connection or not. This allows most of the buffering resources to be only allocated for legitimate TCP handshake.

Conclusion

Intelligent agents have been proposed to overcome the risks associated with the current communication protocols, routers, firewalls, and IDSs. The current solutions for preventing DDoS attacks are still not sufficient. The intelligent fuzzy agents are

Figure 5
The RFA manages to control the router's buffer allocation during Smurf attack (rules view)

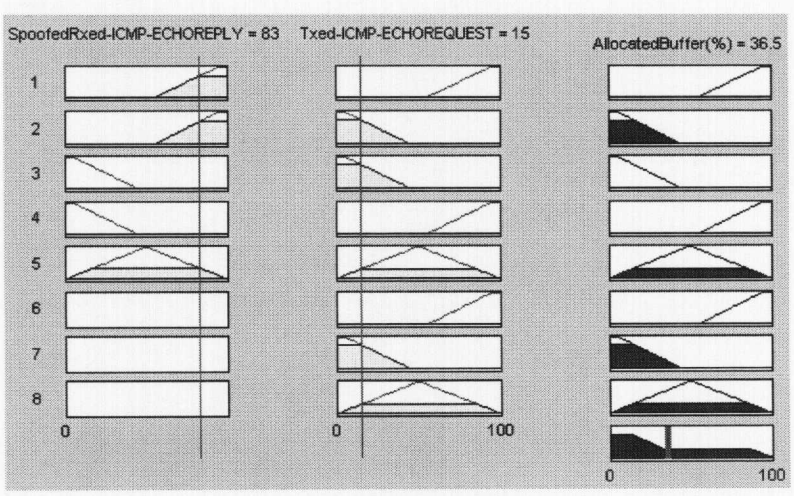


Figure 6
The RFA manages to control the router's resources (buffer) allocation during Smurf DoS attack with higher Txed ICMP_ECHOREQ packets traffic (rules view)

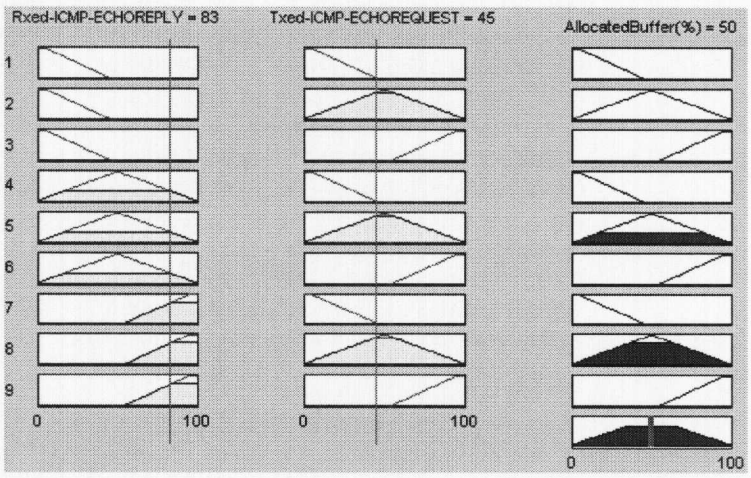


Figure 7
The RFA dynamically manages to control the router's buffer allocation during Smurf attack (surface view)

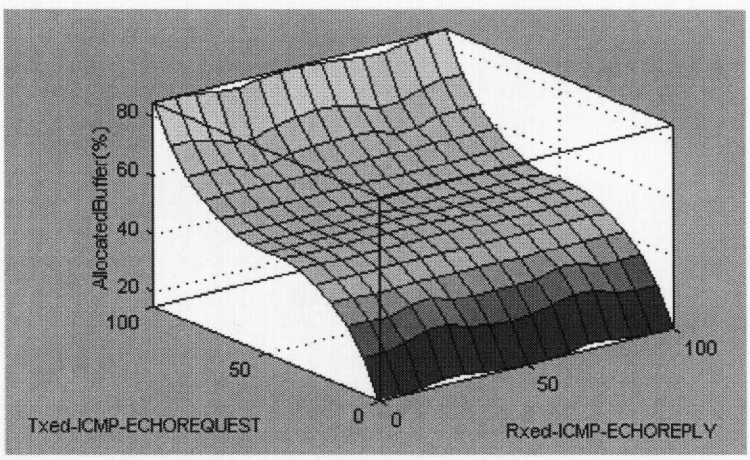
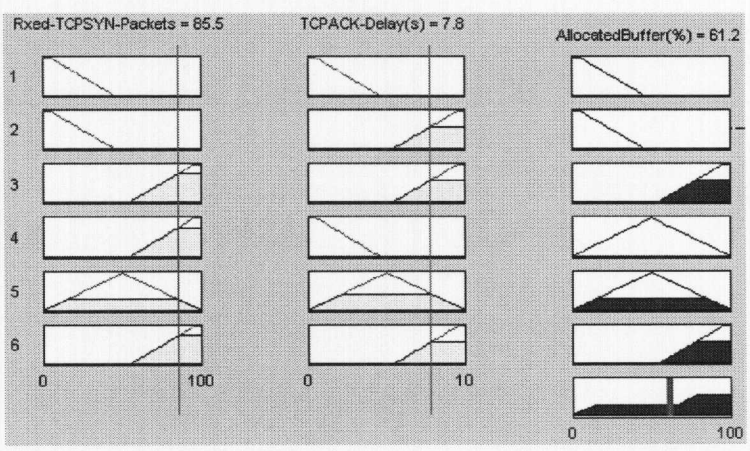


Figure 8
The SFA dynamically controls the allocated buffer by TCP protocol in response to different conditions of TCP SYN flooding attacks (delayed TCP ACK packets)



Information Management & Computer Security
10/5 [2002] 203–209

considered to be a useful tool to provide dynamically automated response actions in a form of managing resources availability once a DDoS takes a place. Fuzzy agents are capable of providing a solution that can be implemented in a cost-effective manner without any TCP/IP protocol modification and special hardware requirements.

References

Angeles, M., Caffaro, G. and Caffaro, M.G. (2001), "The challenges that XML faces", *IEEE Computer*, Vol. 34 No. 10, October, pp. 15-18.

Cohen, F.B. (1999), "Simulating cyber attacks, defences, and consequences", Fred Cohen and Associates, Livermore, CA, March, available at: http://all.net/

Criscuolo, P.J. (2000), "Distributed denial of service", CIAC-2319, 14 February, available at: www.securitytechnet.com/security/hacking.html

Gligor, V. (1983), "A note on the denial-of-service problem", IEEE Symposium on Research in Security and Privacy.

International Standards Organisation (ISO) (1988), "Information processing systems – open systems interconnection (OSI) – basic reference model – part 2: security architecture", ISO 7498-2, ISO, Geneva.

Jang, J.S.R., Sun, C.T. and Mizutani, E. (1997), *Neruro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*, Prentice-Hall, Englewood Cliffs, NJ.

Needham, R. (1994), "Denial of service", *Proceedings of the 1st ACM Conference on Computer and Communications Security*.

Northcutt, S., Cooper, M., Fearnow, M. and Frederick, K. (2001), *Intrusion Signatures and Analysis*, New Rider, Indianapolis, IN.

Schneier, B. (2000), Secrets & *Lies, Digital Security in a Networked World*, John Wiley, Toronto.